



Université
de Lille

UNIVERSITÉ DE LILLE

SAÉ S2.02

THÉORIE DES GRAPHS

RAPPORT

Graphes - Version finale

Élèves :

Baptiste LAVOGIEZ

Mark ZAVADSKIYI

Angèl ZHENG

Enseignant :

Iovka BONEVA

16 mai 2025

Table des matières

1	Contexte	2
2	Version 1	3
2.1	Introduction	3
2.2	Choix pour la modélisation	3
2.3	Exemple complet	4
2.4	Score d'affinité	4
2.5	Retour sur l'exemple	6
2.6	Conclusion	7
3	Version 2	8
3.1	Introduction	8
3.2	Choix pour la modélisation	8
3.3	Exemple avec appariement total	9
3.4	Exemple sans appariement total	9
3.5	Score d'affinité	10
3.6	Retour sur l'exemple	12
3.7	Pour aller plus loin	14
3.8	Robustesse de la modélisation (question difficile)	15
3.9	Conclusion	16
4	Version 3	17
4.1	Introduction	17
4.2	Équilibrage entre affinité incompatibilité	18
4.3	Score d'affinité	19
4.4	Retour sur l'exemple	21
4.5	Conclusion	25

1 Contexte

Ce **rapport** présentera les trois versions de la partie *Graphes* de la SAÉ S2.02. Il traitera de modélisation théorique d'appariement, puis de la réalisation d'un pseudo-code. Le problème traité ici est de quantifier l'affinité entre un hôte et un visiteur donné présentant des caractéristiques propres (hobbies, préférence de genre, âge...) pour ensuite l'implémenter sous forme de graphe biparti en utilisant la matrice d'adjacence. Nous devons juger de leur compatibilité afin de présenter les appariements les plus justes dans le cadre d'un échange scolaire. Tous les fichiers ou documents utilisés (scripts, archive...) sont disponibles dans l'archive de ce fichier.

Les trois versions traiteront chacune d'une modélisation différente, en posant un autre regard sur les critères importants qui composeront les paires hôte-visiteur jugées comme les plus adaptées.

Précision

La notation se fait suivant un plan avec des sections précises et nécessaires. Ce document comporte toutes ces sections, mais inclut également des sections supplémentaires souhaitées à l'écriture pour plus de clarté (Introduction, Conclusion, Pour aller plus loin...).

2 Version 1

2.1 Introduction

Cette version sert à poser les premières briques de notre modèle et de notre façon de fonctionner. Elle utilise seulement les composantes de base, telles que la différence d'âge, les hobbies en commun et les préférences de genre. Il n'y a pas de contrainte à proprement parler.

2.2 Choix pour la modélisation

Personnes / Critères	HOBBIES	GENDER	PAIR_GENDER	BIRTH_DATE
H1	foot	male		19/06/11
H2	basket	female	female	29/06/12
H3	running	other	other	15/09/13
H4	swimming	other	female	04/07/11
H5	volley	male		28/08/12
V1	foot	male	male	19/07/11
V2	handball	male	male	08/06/11
V3	foot	other	female	15/11/12
V4	running	female		29/05/11
V5	volley	female	female	11/08/13

FIGURE 1 – Tableau fictif d'hôtes et de visiteurs

Forte affinité

La paire $(H1, V1)$ présente une **forte affinité** selon nos critères car :

- Ils aiment tous deux le **football**.
- Ils ont seulement **un mois d'écart d'âge**.
- $H1$ préfère avoir un visiteur **male** et $V1$ est un **male**..

Faible affinité

La paire $(H2, V2)$ présente une **faible affinité** selon nos critères car :

- Ils n'ont **pas de hobbies en commun**.
- Ils ont **un an d'écart**.
- Leurs **préférences de genre** ne sont pas satisfaites.

Arbitrage entre les critères d'affinité

Les trois autres paires $(H3, V3)$, $(H4, V4)$, $(H5, V5)$ présentent une affinité relativement équivalente. Ces étudiants sont plus ou moins compatibles, à un niveau tel qu'on ne

peut distinguer une faible ou forte affinité claire.

- $(H3, V3)$ ont pour seul critère satisfait **la préférence de genre** de $H3$ envers $V3$. La préférence de genre de $V3$ n'est, elle, pas satisfaite.
- $(H4, V4)$ ont pour seul critère satisfait **la préférence de genre** de $H4$ envers $V4$. $V4$ n'a pas de préférence de genre définie. Néanmoins, leur âge est très proche.
- $(H5, V5)$ **partagent un hobby : le volley**. Si ce n'est ce point, ils ne partagent pas beaucoup d'affinité, n'ayant pas de préférence de genre satisfaite, et ayant un an d'écart.

2.3 Exemple complet

Après une introduction théorique à notre vision de ce problème, nous pouvons déterminer un **exemple complet et pratique**, ayant 4 hôtes et 4 visiteurs sobrement nommés par ordre alphabétique.

HOTES	HOBBIES	GENDER	PAIR_GENDER	BIRTH_DATE
Alix	football, trees, cats, tea	male		19/06/11
Brune	basket, music, kpop, dogs	female	female	20/09/12
Charlie	running, computer, dogs	other	other	05/12/10
Diego	swimming, running, hiking	other	male	24/04/11
VISITEURS				
William	football, handball, computer	male	male	26/09/13
Xavier	handball, running, cats	male	male	27/09/12
Yvan	football, trees, dogs, coffee	other	other	12/09/13
Zorro	running, hiking, music	female		10/12/12

FIGURE 2 – Tableau complet d'hôtes et de visiteurs

L'appariement que nous pensons être le meilleur est :

- Alix et Xavier (cats, préférence de genre, âge proche)
- Brune et Zorro (music, préférence de genre, âge très proche)
- Charlie et Yvan (dogs, préférence de genre 2x, malgré un âge éloigné)
- Diego et William (par élimination)

2.4 Score d'affinité

Après avoir réfléchi sur un appariement entre les hôtes et visiteurs, il est temps de mettre en place une logique d'importance sur chaque critère. Pour ce faire, un algorithme en pseudo-code nous permet de définir un score d'affinité entre chaque hôte et visiteur. Ce score varie en fonction des caractéristiques similaires constatées afin d'appareiller des hôtes et visiteurs les plus compatibles possible. Plus ce score est bas, plus les personnes sont jugées compatibles. Ces valeurs seront utilisées par la suite afin de construire les poids des arêtes d'un **graphe biparti**.

Algorithme

```

1  score_affinité_1(hôte , visiteur):
2      score = 10
3      // valeur initiale du score
4
5      hôte.age=dateaujourd'hui-hôte.naiss
6      visiteur.age=dateaujourd'hui-visiteur.naiss
7
8      diff = valeurabsolue(hôte.naiss - visiteur.naiss)
9      // la différence se compte à la précision du mois
10     age_moyen = (hôte.age + visiteur.age) / 2
11
12     si diff <= 18 mois
13         score += (diff / 12) * 0.9**age_moyen
14         score = score * 0.9 // diminution de 10% car proches
15                               en âge
16     sinon
17         score += ((diff / 12) * 0.9**age_moyen)
18         score = score * 1.5 // la différence est plus
19                               prononcée avec un facteur de 1.5 car ils sont
20                               assez éloignés en âge (+ de 18 mois)
21
22     // l'objectif est de calculer une différence qui est
23     // proportionnelle à l'âge des personnes.
24     // quand on a 10 ans, et l'autre 11 ans, l'autre est 10%
25     // plus âgé ;
26     // quand on a 20 ans, et l'autre 21 ans, l'autre est 5%
27     // plus âgé ;
28     // le but est de mettre la différence, et donc l'impact
29     // sur l'affinité, à l'échelle de l'âge de la personne.
30
31     si (visiteur.pair_gender existe) // préférence exprimée
32         si hôte.gender != visiteur.pair_gender
33             score = score * 1.5
34     // la séquence contraire de cet événement ne change pas
35     // le score, donc on cherche la négativité initiale de
36     // la comparaison
37
38     si (hôte.pair_gender existe) // préférence exprimée
39         si visiteur.gender != hôte.pair_gender
40             score = score * 1.5
41
42     N = nombre_hobbies_en_commun(hôte , visiteur)
43     score = score * 0.85**N
44     // le score est diminué d'un facteur proportionnel au
45     // nombre de hobbies (*0.85 si un en commun, *0.72 si
46     // deux...)
47     retourner arrondir(score , 2)

```

Cet algorithme a été testé plusieurs fois, puis rééquilibré afin qu'un critère satisfait ne soit pas trop important par rapport aux autres. Nous avons toutefois décidé d'apporter une bonne importance aux préférences de genre et à l'âge.

2.5 Retour sur l'exemple

En appliquant la fonction définie ci-dessus entre chaque hôte et visiteur, nous pouvons en déduire une matrice d'adjacence du graphe biparti entre les hôtes et visiteurs, les poids des arêtes étant les scores d'affinité.

Hôte / Visiteur	William	Xavier	Yvan	Zorro
Alix	13.56	7.91	17.29	9.36
Brune	20.85	20.25	17.72	7.7
Charlie	30.8	29.96	13.69	20.1
Diego	23.96	11.89	23.96	16.94

FIGURE 3 – Matrice d'adjacence avec poids

Représenté au *format CSV*, on obtient le fichier suivant :

```
;William;Xavier;Yvan;Zorro
Alix;13.56;7.91;17.29;9.36
Brune;20.85;20.25;17.72;7.7
Charlie;30.8;29.96;13.69;20.1
Diego;23.96;11.89;23.96;16.94
```

FIGURE 4 – Fichier CSV généré

Nous remarquons que les scores obtenus sont cohérents et qu'il n'y a pas d'énorme écart de valeur.

Ensuite, nous n'avons plus qu'à calculer l'affectation de coût minimal pour ce graphe en utilisant l'archive *jar* "**calcul-affectation**" prenant en entrée un fichier CSV. Nous obtenons alors l'appariement suivant :

```
$ java -jar calcul-affectation.jar scores_affinite.csv
Matrice du graphe :
      William  Xavier    Yvan    Zorro
Alix    13.560    7.910    17.290    9.360
Brune   20.850    20.250    17.720    7.700
Charlie 30.800    29.960    13.690    20.100
Diego   23.960    11.890    23.960    16.940

Affectation de cout minimal 46.840000 :
(Diego, Xavier, 11.890000)
(Alix, William, 13.560000)
(Brune, Zorro, 7.700000)
(Charlie, Yvan, 13.690000)
```

FIGURE 5 – Exécution de l'archive sur le graphe

L'appariement obtenu n'est pas totalement celui que nous avons identifié comme le meilleur, car il y a plusieurs possibilités et nous pouvons remarquer que Alix a du "sacrifier" sa bonne entente avec Xavier au profit de celle entre Diego et Xavier, car le "second choix" de Alix est moins coûteux que celui de Diego.

Ce genre de remarque est difficile à établir sans programme, ou alors elle est affirmée avec peu de certitudes. Le programme nous aide alors à voir plus loin que les simples affinités en réglant des conflits d'appariement.

2.6 Conclusion

Cette première version nous a permis d'établir les bases de notre solution d'appariement en prenant en main les outils utiles, qu'ils soient théoriques (pseudo-code) ou pratiques (affectation, génération CSV...).

3 Version 2

3.1 Introduction

Cette deuxième version du rapport de la partie *Graphes* de la SAÉ S2.02 traitera de la modélisation d'un graphe biparti entre les hôtes et les visiteurs, ainsi que de l'implémentation d'un algorithme d'appariement. Cette partie traitera de l'implémentation de contraintes rédhibitoires, de la recherche d'appariement total et de l'optimisation de l'appariement.

3.2 Choix pour la modélisation

Nous nous appuierons sur l'exemple suivant :

FORENAME;	NAME;	COUNTRY;	BIRTH_	GUEST_	HOST_	GUEST_	HOST_	HOBBIES;	GEN	PAIR_	HIS
DATE;			ANIMAL_	HAS_	FOOD_	FOOD;			DER;	GEN	TO
			ALLERGY;	ANIMAL;	CONS					DER;	RY
					TRAIINT;						
Adonia;	A;	FRANCE;	;	no;	;	;	;	sports,technology;	;	;	;
Bellatrix;	B;	FRANCE;	;	yes;	;	;	;	culture,science;	;	;	;
Callista;	C;	FRANCE;	;	no;	;	;	;	science,reading;	;	;	;
Xolag;	X;	ITALY;	;	;	no;	;	;	culture,technology;	;	;	;
Yak;	Y;	ITALY;	;	;	yes;	;	;	science,reading;	;	;	;
Zander;	Z;	ITALY;	;	;	no;	;	;	technology;	;	;	;

FIGURE 6 – Tableau exemple

Note

Nous partons du principe que les visiteurs n'amènent pas de nourriture ni d'animaux avec eux (donc les allergies et les préférences alimentaires des hôtes ne sont pas à prendre en compte).

Explication

Nous reprenons nos 8 personnes préférées mais cette fois nous les connaissons mieux. En effet, nous les avons interrogées sur leurs préférences alimentaires, leurs allergies, leurs animaux de compagnie.

Le fait de connaître plus d'informations sur les hôtes et les visiteurs nous permet de mieux les appairer et de faire une comparaison avec la première version par la suite. Les paires définies comme les meilleures ne seront pas forcément les mêmes que celles de la première version, car nous avons plus d'informations sur les personnes.

Nous remarquons que malgré son allergie aux chats, Xavier les apprécie beaucoup. Il doit juste en rester éloigné. Brune et Charlie adorent les chiens, mais leurs parents leur ont toujours interdit d'en avoir un... Diego a des animaux mais il n'y éprouve pas beaucoup d'intérêt, préférant manger du melon au citron.

Ainsi, les hobbies et la possession d'animaux sont des critères n'étant pas forcément reliés !

En plus de ces 8 personnes, d'autres se sont également manifestées pour participer à l'échange scolaire (car au total, la suite des questions nous demande 16 personnes).

L'historique

En plus de ça, nous avons implémenté la notion d'historique. Dans la représentation d'un tableau, cela est considéré comme :

- soit il souhaite *other*, un autre correspondant que le précédent
- soit il n'a pas renseigné, donc il peut aussi bien être avec son précédent correspondant qu'un nouveau, cela dépendra du score.

Ainsi, si *other* est renseigné, cela sera traité comme une contrainte.

Pour ce qui est de qui est le précédent correspondant, cela sera stocké dans le code et non dans le tableau. L'attribut *past_people* présente les anciens correspondants de la personne. Il est *caché* dans le pseudo-code.

3.3 Exemple avec appariement total

Pour cet exemple, nous devons appareiller les hôtes et les visiteurs de manière à respecter les contraintes rédhibitoires. Il existe des incompatibilités dans cet ensemble, mais il est possible de trouver un appariement qui respecte les contraintes.

Nous avons les incompatibilités suivantes :

- Alix et Xavier | Zorro (allergie aux chats)
- Alix et Yvan (allergie aux cacahuètes)
- Brune et Zorro (Brune adore le lait, mais Zorro y est allergique)
- Charlie et William (allergie au poisson)
- Diego et Zorro (allergie aux chiens)

Ce qui nous donne l'appariement qui apparaît comme le meilleur :

- Alix et William (par élimination, et ils n'ont pas d'incompatibilité)
- Brune et Yvan (dans la version 1, Brune était avec Zorro, mais ils sont maintenant incompatibles. La deuxième meilleure affinité est avec Yvan).
- Charlie et Zorro (car Zorro est incompatible avec tout le monde sauf Charlie)
- Diego et Xavier (ils s'entendaient très bien dans la première version et n'ont pas d'incompatibilité)

Le calcul se fait à la main, en vérifiant hôte par hôte les visiteurs compatibles. Par la suite, un programme pourrait être mis en place pour faire ce calcul de manière automatique car des bases beaucoup plus grandes pourraient être utilisées.

3.4 Exemple sans appariement total

Nous avons ici nos 4 nouveaux hôtes et visiteurs.

La réunion impossible

Toutefois, il n'est pas possible de former quatre paires hôte-visiteur à cause d'incompatibilités.

La raison est simple :

- Yaoundé est incompatible avec tous les hôtes
- Par conséquent, il n'est pas possible de former quatre paires hôte-visiteur sans incompatibilité.
- Un hôte au moins devra alors rester seul.

En ce qui concerne les autres, aucune autre personne n'est incompatible avec tout le monde. Nous pouvons alors former 3 paires hôte-visiteur au maximum (voir exemple ci-dessous).

Le meilleur appariement

Le meilleur appariement que nous avons trouvé est :

- L'hôte restant et Yaoundé (ici, Carmen)
- Bosphore et Williamelle (pas d'incompatibilité)
- Douala et Xanthane (beaucoup de points en commun, et pas d'incompatibilité)
- Ardente et Zoroark (points en commun, et pas d'incompatibilité)

3.5 Score d'affinité

Implémentation

Dans la logique énoncée qu'un score le plus bas possible signifie une forte affinité, nous devons rendre un score très élevé en cas d'incompatibilité afin de le rendre impossible à emprunter dans le cadre d'un parcours de graphe. Dans cette version, pour la sécurité des personnes, il est normalement impossible de faire un appariement entre deux personnes incompatibles et même la pire des affinités au monde doit primer sur une incroyable amitié mais incompatible.

Il est à noter que la troisième version regardera cela d'un autre angle en cherchant à compenser des incompatibilités par des affinités fortes. Notre logique sera donc revue d'une autre manière. Il n'existe pas de solution parfaite, mais juste des points de vue et solutions différentes et elles seront explorées.

Algorithme

```

1  score_affinité_2(hôte, visiteur):
2      score = 10
3
4      hôte.age = dateaujourd'hui - hôte.naiss
5      visiteur.age = dateaujourd'hui - visiteur.naiss
6
7      diff = valeurabsolue(hôte.naiss - visiteur.naiss)
8      age_moyen = (hôte.age + visiteur.age) / 2
9
10     si diff <= 18 mois
11         score += (diff / 12) * 0.9**age_moyen
12         score = score * 0.9
13     sinon
14         score += ((diff / 12) * 0.9**age_moyen)
15         score = score * 1.5
16
17     si ((visiteur.pair_gender existe) et (si hôte.gender !=
18         visiteur.pair_gender))
19         score = score * 1.5
20     si ((hôte.pair_gender existe) et (si visiteur.gender !=
21         hôte.pair_gender))
22         score = score * 1.5
23
24     N = nombre_hobbies_en_commun(hôte, visiteur)
25     score = score * 0.85**N
26
27     // Gestion de l'incompatibilité
28     // On vérifie si l'hôte a un animal et si le visiteur
29     // est allergique ou si l'hôte a un aliment et si le
30     // visiteur a une contrainte alimentaire
31     // Si c'est le cas, on ajoute 1000 au score, car on veut
32     // juste savoir si c'est incompatible ou pas.
33
34     si (un animal dans hôte.has_animals est dans visiteur.
35         animal_allergy)
36     ou ((un aliment dans hôte.host_food est dans visiteur.
37         food_constraint))
38     // on regarde l'historique
39     ou (hôte.history == "other" et visiteur.nom est dans
40         hôte.past_people)
41     ou (visiteur.history == "other" et hôte.nom est dans
42         visiteur.past_people)
43     score += 1000
44
45     retourner arrondi(score, 2)
  
```

Remarque

Nous ajoutons 1000 au score à la toute fin en cas d'incompatibilité. Pourquoi procéder ainsi ?

Par exemple, l'hôte Xoxo et le visiteur Yoyo ont une affinité de 12, mais ils sont incompatibles car Xoxo possède un dragon et Yoyo est allergique aux dragons. Leur score sera donc de 1012.

- Si nous ajoutons 1000 au score à la fin, cela ne changera pas le score d'affinité entre deux personnes compatibles.
- Nous pourrions ainsi retrouver le score d'affinité entre deux personnes incompatibles en enlevant 1000 si la valeur est supérieure à 1000.
- Nous pouvons ainsi analyser les scores d'affinité par la suite !

Deuxième remarque

Le pseudo-code utilisé se base sur la première version et il a été optimisé en termes d'écriture et de lisibilité. Par conséquent, les commentaires du pseudo-code de la première version ont été supprimés mais sont toujours présents dans la première version.

D'autres indications seront apportées par la suite (notamment sur pourquoi la valeur 1000 a été choisie).

3.6 Retour sur l'exemple

Données utilisées

Les personnes utilisées lors du calcul seront celles de l'ensemble 1 et 2, d'abord séparément, puis ensuite réunifiées (partie supplémentaire à la notation).

Ensemble 1

Nous obtenons les matrices d'adjacence suivantes :

	William	Xavier	Yvan	Zorro
Alix	13.56	1007.91	1017.29	1009.36
Brune	20.85	20.25	17.72	1007.7
Charlie	1030.8	29.96	13.69	20.1
Diego	23.96	11.89	23.96	1016.94

FIGURE 7 – Matrice d'adjacence de l'ensemble 1

Ensemble 1, le meilleur appariement

Nous avons trouvé le meilleur appariement entre les hôtes et les visiteurs.

```
Matrice du graphe :
      William  Xavier    Yvan   Zorro
Alix  13.560 1007.910 1017.290 1009.360
Brune  20.850   20.250   17.720 1007.700
Charlie 1030.800   29.960   13.690   20.100
Diego   23.960   11.890   23.960 1016.940

Affectation de cout minimal 63.270000 :
(Diego, Xavier, 11.890000)
(Brune, Yvan, 17.720000)
(Charlie, Zorro, 20.100000)
(Alix, William, 13.560000)
```

FIGURE 8 – Meilleur appariement pour l'ensemble 1

Ensemble 1, Remarque

Nous remarquons que Alix et Xavier avaient la meilleure affinité si l'on enlève l'incompatibilité ($1007.91 - 1000 = 7.91$). C'est donc l'utilité d'avoir ajouté une valeur fixe.

Ensemble 2

Nous obtenons les matrices d'adjacence suivantes :

```
+-----+-----+-----+-----+
|           | Williamelle | Xanthane | Yaoundé | Zoroark |
+-----+-----+-----+-----+
| Ardente   | 1015.88   | 13.7    | 1030.11 | 13.8    |
| Bosphore  | 20.03     | 1017.29 | 1021.01 | 17.4    |
| Carmen    | 31.31     | 1035.31 | 1013.8  | 35.61   |
| Douala    | 35.58     | 11.6    | 1035.26 | 1009.93 |
+-----+-----+-----+-----+
```

FIGURE 9 – Matrice d'adjacence de l'ensemble 2

Ensemble 2, le meilleur appariement

Nous avons trouvé le meilleur appariement entre les hôtes et les visiteurs.

```

Matrice du graphe :
      Willame Xanthane Yaoundé Zoroark
Ardente 1015.880 13.700 1030.110 13.800
Bosphore 20.030 1017.290 1021.010 17.400
Carmen 31.310 1035.310 1013.800 35.610
Douala 35.580 11.600 1035.260 1009.930

Affectation de cout minimal 1059.230000 :
(Douala, Xanthane, 11.600000)
(Ardente, Zoroark, 13.800000)
(Bosphore, Williamelle, 20.030000)
(Carmen, Yaoundé, 1013.800000)
  
```

FIGURE 10 – Meilleur appariement pour l'ensemble 2

Nous pouvons remarquer qu'effectivement, il y a 3 paires possibles au maximum dans cet ensemble. Carmen a du être appareillé avec Yaoundé. Cette arête pesant plus de 1000, et représentant donc une incompatibilité, n'est pas à prendre en compte.

Ainsi, Carmen et Yaoundé, sont, dans cette version, seuls.

3.7 Pour aller plus loin

Cette partie est supplémentaire à la notation et sert d'argument pour la suite.

Nous pouvons également chercher un appariement entre les 16 personnes !

	William	Xavier	Yvan	Zorro	Williamelle	Xanthane	Yaoundé	Zoroark
Alix	13.56	1007.91	1017.29	1009.36	20.43	1011.74	1020.25	1013.92
Brune	20.85	20.25	17.72	1007.7	9.31	1013.62	1017.64	13.53
Charlie	1030.8	29.96	13.69	20.1	30.93	20.94	16.04	1035.18
Diego	23.96	11.89	23.96	1016.94	1030.67	1013.85	20.28	1011.86
Ardente	35.58	20.75	30.24	1007.89	1015.88	13.7	1030.11	13.8
Bosphore	14.07	1011.63	21.11	1009.18	20.03	1017.29	1021.01	17.4
Carmen	36.69	35.68	1016.31	1023.93	31.31	1035.31	1013.8	35.61
Douala	1015.74	1009.18	35.42	1011.78	35.58	11.6	1035.26	1009.93

FIGURE 11 – Matrice d'adjacence de l'ensemble 1 et 2

```

Matrice du graphe :
      William Xavier Yvan Zorro Willame Xanthane Yaoundé Zoroark
Alix 13.560 1007.910 1017.290 1009.360 20.430 1011.740 1020.250 1013.920
Brune 20.850 20.250 17.720 1007.700 9.310 1013.620 1017.640 13.530
Charlie 1030.800 29.960 13.690 20.100 30.930 20.940 16.040 1035.180
Diego 23.960 11.890 23.960 1016.940 1030.670 1013.850 20.280 1011.860
Ardente 35.580 20.750 30.240 1007.890 1015.880 13.700 1030.110 13.800
Bosphore 14.070 1011.630 21.110 1009.180 20.030 1017.290 1021.010 17.400
Carmen 36.690 35.680 1016.310 1023.930 31.310 1035.310 1013.800 35.610
Douala 1015.740 1009.180 35.420 1011.780 35.580 11.600 1035.260 1009.930

Affectation de cout minimal 145.440000 :
(Douala, Xanthane, 11.600000)
(Bosphore, Yvan, 21.110000)
(Ardente, Zoroark, 13.800000)
(Alix, William, 13.560000)
(Brune, Williamelle, 9.310000)
(Carmen, Xavier, 35.680000)
(Charlie, Zorro, 20.100000)
(Diego, Yaoundé, 20.280000)
  
```

FIGURE 12 – Meilleur appariement pour l'ensemble 1 et 2

3.8 Robustesse de la modélisation (question difficile)

Cette partie consiste à analyser la capacité de notre modèle à faire respecter les contraintes avec des paramètres flexibles.

La question est la suivante :

Notre modélisation respecte-elle les contraintes, peu importe l'ensemble défini ?

La réponse est :

- Oui, dans le sens où une relation incompatible a un score qu'il est impossible de dépasser avec la pire affinité non incompatible possible
- Concrètement, comme dans le meilleur appariement pour l'ensemble 2, si le meilleur appariement comprend une incompatibilité, au vu du grand écart entre une mauvaise affinité (environ un score de 30) et une affinité quelconque incompatible (1000+).

Dans le cas d'un grand exemple, tel que l'ensemble 1 et 2 réuni (16 personnes, donc 64 affinités calculées).

Rappel

```
Matrice du graphe :
      William  Xavier   Yvan   Zorro William Xanthane Yaoundé Zoroark
Alix   13.560 1007.910 1017.290 1009.360   20.430 1011.740 1020.250 1013.920
Brune  20.850  20.250  17.720 1007.700    9.310 1013.620 1017.640   13.530
Charlie 1030.800 29.960  13.690  20.100   30.930  20.940  16.040 1035.180
Diego  23.960  11.890  23.960 1016.940 1030.670 1013.850  20.280 1011.860
Ardente 35.580  20.750  30.240 1007.890 1015.880  13.700 1030.110   13.800
Bosphore 14.070 1011.630 21.110 1009.180   20.030 1017.290 1021.010   17.400
Carmen 36.690  35.680 1016.310 1023.930   31.310 1035.310 1013.800   35.610
Douala 1015.740 1009.180  35.420 1011.780   35.580  11.600 1035.260 1009.930

Affectation de cout minimal 145.440000 :
(Douala, Xanthane, 11.600000)
(Bosphore, Yvan, 21.110000)
(Ardente, Zoroark, 13.800000)
(Alix, William, 13.560000)
(Brune, Williamelle, 9.310000)
(Carmen, Xavier, 35.680000)
(Charlie, Zorro, 20.100000)
(Diego, Yaoundé, 20.280000)
```

FIGURE 13 – Meilleur appariement pour l'ensemble 1 et 2

Nous voyons qu'il y a beaucoup de relations incompatibles (score à plus de 1000), or, lors de l'appariement elles sont toujours respectées.

Justification

Nous disons que notre modèle garantit que les contraintes soient respectées, c'est à dire qu'il est impossible de dépasser un score de 1000 (le minimum pour une relation incompatible quelconque) avec une relation n'ayant pas d'incompatibilité.

Ce qui revient à dire que n'importe quelle relation non incompatible sera toujours préférée à n'importe quelle relation incompatible, peu importe les affinités dans les deux cas.

Comment le prouver ? Prenons l'exemple de la pire affinité possible sans incompatibilité selon le pseudo-code.

- On démarre avec un score de 10.
- On a une différence d'âge de 10 ans (ce qui est déjà quasiment impossible en pratique) et un âge moyen de 10 pour exemple (5 et 15 ans).
- Soit $\text{score} += (120 \text{ mois de différence} * 0.9 * 10) \rightarrow$ on ajoute 41.84 au score et on le multiplie par 1.5. Soit 77.76
- Les deux préférences de genre ne sont pas respectées. On augmente donc le score de 1.5 deux fois, soit par 2.25
- $77.76 * 2.25 = 174.96$
- Ensuite, il n'y a pas de hobbies en commun donc le score est multiplié par 0.85^0 , soit 1. Le score ne change donc pas.
- Enfin, il n'y a pas d'incompatibilité, donc le score ne change pas.

En résumé, on parle d'une situation déjà très improbable en terme de différence d'âge et qui a la pire affinité possible. On obtient 175 de score d'affinité, soit le maximum de toute relation sans incompatibilité, ce qui reste assez éloigné de 1000, soit le minimum de toute relation avec incompatibilité.

Nous pouvons donc affirmer que n'importe quelle relation non incompatible sera toujours préférée à n'importe quelle relation incompatible, peu importe les affinités dans les deux cas.

3.9 Conclusion

Cette deuxième version nous a permis d'étendre les capacités de notre solution d'appariement en implémentant la gestion de contraintes rédhitoires, en utilisant un algorithme donnant des scores élevés résultant en arêtes impossibles à emprunter dans le graphe biparti.

4 Version 3

4.1 Introduction

Cette version traitera des éléments de la deuxième version sous un autre angle, en suivant ce paradigme :

Est-ce qu'une incompatibilité peut être surpassée par une énorme affinité ?

Nous nous demandons si, des personnes avec une forte affinité, partageant beaucoup de critères satisfaits, peuvent "passer au dessus" de certaines incompatibilités. La deuxième version voyait ces contraintes comme rédhibitoires et comme impossibles à surpasser.

Une allergie insurmontable ?

On peut traiter les critères alimentaires autant comme des allergies (deuxième version) que par de simples préférences très souhaitables (troisième version).

On peut aussi traiter les allergies animales comme regrettables, mais aussi évitables.

Voici l'exemple suivant de comment pourrait-on faire en sorte d'appareiller deux personnes avec une forte affinité mais ayant des incompatibilités :

- Toto et Zozo sont de véritables âmes soeur. Ils adorent le football, le paintball, l'accrobranche, la plage, la science-fiction. En plus, ils sont nés le même jour et la même année !
- heureusement, Toto a un chat chez lui car son frère, Dodo, adore les chats.
- Zozo est allergique aux chats. Si il est dans la même pièce qu'un chat pendant plus de cinq minutes, il est très probable qu'il doive aller aux urgences.
- Toto et Zozo sont très embêtés. Ils s'apprêtent à avoir l'échange parfait mais à cause d'un chat, Zozo serait mis en danger.
- Toto décide, avec ses parents Nono et Vovo, de faire garder le chat de Dodo, Wowo, chez son cousin Gogo, qui adore les chats également.
- Dodo est assez embêté et il demande, en compensation, que Toto fasse la vaisselle pendant un mois. Les deux acceptent.

Nous retenons de cet exemple qu'il est possible, dans la vraie vie, de s'arranger pour que des incompatibilités ne soient plus un obstacle, si l'affinité au bout du compte vaut le coup.

Par conséquent, de tels sacrifices nécessitent une forte affinité. On ne prend pas des dispositions spéciales pour n'importe quelle raison.

Lors de la modélisation, nous ferons alors attention à bien définir les scores d'affinité des relations parfaites mais incompatibles afin qu'elles soient parfois préférées à des relations moyennes.

En somme, cela dépend de la vision que l'on donne à notre modélisation, et en *Graphes*,

c'est toute l'essence de la modélisation ; appliquer plusieurs points de vue, sans forcément en chercher un seul unique et parfait. Il est donc important pour nous d'explorer plusieurs versions et possibilités d'appariement !

4.2 Équilibrage entre affinité | incompatibilité

Nous reprenons notre ensemble 1 et 2 précédemment énoncé car il répond à tous nos besoins pour cette version. L'objectif de cet exemple est de démontrer **combien et quel type d'affinité permet de compenser combien et quel type d'incompatibilité**.

ENSEMBLE 1								
HOTES								
NAME	HOBBIES	GENDER	PAIR_GENDER	BIRTH_DATE	HOST_HAS_ANIMAL	HOST_FOOD	HISTORY	PAST_PEOPLE
Alix	football, trees, cats, tea	male	female	19/06/11	cats, rabbits	peanuts	other	Xavier
Brune	basket, music, kpop, dogs	female	other	20/09/12	no	milk		
Charlie	running, computer, dogs	other	other	05/12/10	no	fish		
Diego	swimming, running, hiking	other	male	24/04/11	horse, dogs	watermelon, citrus		
VISITEURS								
NAME	HOBBIES	GENDER	PAIR_GENDER	BIRTH_DATE	GUEST_ANIMAL_ALLERGY	GUEST_FOOD_CONSTRAINT	HISTORY	
William	football, handball, computer	male	male	26/09/13	no	fish		
Xavier	handball, running, cats	male	male	27/09/12	cats, lions		other	Alix
Yvain	football, trees, dogs, coffee	other	other	12/09/13	no	peanuts, grapefruit		Mogadishu
Zorro	running, hiking, music	female		10/12/12	cats, dogs	milk		Carmen
ENSEMBLE 2								
HOTES								
NAME	HOBBIES	GENDER	PAIR_GENDER	BIRTH_DATE	HOST_HAS_ANIMAL	HOST_FOOD	HISTORY	PAST_PEOPLE
Ardente	cycling, music, dogs, tea	female	female	10/10/11	dogs	kiwi		Yoann
Bosphore	reading, chess, cats, piano	male	female	25/03/12	cats	milk		
Carmen	swimming, drawing, rabbits, cinema	other	other	18/05/10	rabbits	peanuts	other	Zorro
Douala	basketball, hiking, reptiles, science	male	male	01/12/11	cats, snakes	fish		
VISITEURS								
NAME	HOBBIES	GENDER	PAIR_GENDER	BIRTH_DATE	GUEST_ANIMAL_ALLERGY	GUEST_FOOD_CONSTRAINT	HISTORY	
Williamelle	cinema, running, cats, tech	female	female	30/11/13	dogs	kiwi, watermelon		Kalpan
Xanthane	reading, reptiles, football	male	female	09/05/12	rabbits, horse	peanuts, milk		
Yaoundé	swimming, music, birds, trees	other	other	15/07/13	cats, rabbits	milk, chocolate, kiwi		Alix
Zoroark	hiking, science, computers, piano	male	female	02/08/12	snakes, bears	fish, citrus	other	Alix

FIGURE 14 – Ensemble 1 et 2

Exemple d'appariement

- Brune et Zorro : ils ont une contrainte mais ont toutefois démontré une très grande affinité. Avec ce modèle, elles devraient être appareillables sous de bonnes conditions.
- Carmen et Yaoundé : leurs préférences de genre sont respectées et ils partagent un hobby. Ils devraient bien s'attendre malgré l'allergie aux lapins.
- Diego et Zorro : ils partagent un hobby et une seule préférence de genre est respectée. Avec l'allergie aux chiens, leur score est considéré comme moyen ; d'autres paires seront normalement préférées. Ce n'est pas assez pour compenser.
- Carmen et Xanthane : ils n'ont pas de hobby en commun, les préférences de genre ne sont pas respectées, et ils ont deux allergies (cacahuètes, lapins). Leur affinité sera parmi les pires car ici aucun critère ne peut rattraper ces deux allergies.
- Globalement, nous pourrions observer la matrice d'adjacence des deux ensembles dans la deuxième version en comparaison à la troisième version pour donner à voir les différences concrètes.

En somme, et toujours selon l'algorithme, le score peut surtout baisser grâce aux hobbies en commun, qui peuvent donc compenser une incompatibilité. De plus, la préférence de genre, si satisfaite, n'engendre pas la multiplication du score par 1.5.

Pour ce qui est de combien, la réponse est mathématique car le partage se fait surtout dans l'algorithme et traduit au mieux notre pensée.

4.3 Score d'affinité

Implémentation

Ici, la contrainte est gérée comme une préférence. Ainsi, le but est que le score d'affinité ne soit pas extrêmement élevé dans ces cas (d'une façon opposée à la version 2) afin que le chemin puisse être emprunté dans un graphe, en cas de fortes affinités.

Algorithme

```

1  score_affinité_3(hôte, visiteur):
2
3      score = 10
4      si (un animal dans hôte.has_animals est dans visiteur.
        animal_allergy) score+= 10
5      si (un aliment dans hôte.host_food est dans visiteur.
        food_constraint) score+= 10
6      si (hôte.history == "other" et visiteur.nom est dans
        hôte.past_people) score+= 10
7      si (visiteur.history == "other" et hôte.nom est dans
        visiteur.past_people) score+= 10
8
9      hôte.age=dateaujourd'hui-hôte.naiss
10     visiteur.age=dateaujourd'hui-visiteur.naiss
11
12     diff = valeurabsolue(hôte.naiss - visiteur.naiss)
13     age_moyen = (hôte.age + visiteur.age) / 2
14
15     si diff <= 18 mois
16         score += (diff / 12) * 0.9**age_moyen
17         score = score * 0.9
18     sinon
19         score += ((diff / 12) * 0.9**age_moyen)
20         score = score * 1.5
21
22     si ((visiteur.pair_gender existe) et (si hôte.gender !=
        visiteur.pair_gender))
23         score = score * 1.5
24     si ((hôte.pair_gender existe) et (si visiteur.gender !=
        hôte.pair_gender))
25         score = score * 1.5
26
27     N = nombre_hobbies_en_commun(hôte, visiteur)
28     score = score * 0.85**N
29
30     retourner arrondi(score, 2)

```

Remarque

Nous choisissons d'incrémenter le score au début, car il peut par la suite être baissé par le nombre de hobbies partagé, mais aussi réaugmenté si les contraintes ne suffisent pas. Nous trouvons que cette logique est une des plus équilibrées afin de réutiliser notre algorithme pour le plus de clarté. La valeur de 10, choisie comme incrémentation du score pour chaque contrainte, a été choisie après plusieurs essais. Après avoir essayé 15,

la contrainte était trop difficile à surmonter dans l'algorithme.

En utilisant la valeur 10 à chaque contrainte, nous ouvrons la porte, pas de beaucoup, mais d'assez pour que les paires les plus courageuses puissent s'y faufiler grâce à leur affinité dite compensatoire.

4.4 Retour sur l'exemple

Données utilisées

Afin de permettre une comparaison avec les résultats obtenus lors de la deuxième version, les mêmes ensembles seront utilisés. Cela permettra de voir quelles paires peuvent être nouvellement formés selon ce nouveau paradigme !

Ensemble 1

Nous obtenons les matrices d'adjacence suivantes :

	William	Xavier	Yvan	Zorro
Alix	13.56	15.56	33.55	18.36
Brune	20.85	20.25	17.72	15.35
Charlie	59.49	29.96	13.69	20.1
Diego	23.96	11.89	23.96	33.2

FIGURE 15 – Matrice d'adjacence de l'ensemble 1

Ensemble 1, le meilleur appariement

Nous avons trouvé le meilleur appariement entre les hôtes et les visiteurs.

```

Matrice du graphe :
      William  Xavier    Yvan   Zorro
Alix   13.560   15.560   33.550  18.360
Brune  20.850   20.250   17.720  15.350
Charlie 59.490   29.960   13.690  20.100
Diego  23.960   11.890   23.960  33.200

Affectation de cout minimal 54.490000 :
(Diego, Xavier, 11.890000)
(Alix, William, 13.560000)
(Brune, Zorro, 15.350000)
(Charlie, Yvan, 13.690000)
    
```

FIGURE 16 – Meilleur appariement pour l'ensemble 1

Ensemble 1, Remarque

Nous remarquons que Brune et Zorro figurent comme paire dans l'affectation de coût minimal malgré leur contrainte. L'algorithme a permis de passer au dessus de cette contrainte car nous avons observé une forte affinité entre les deux. La contrainte a augmenté le score, certes, mais le reste a permis de rattraper. C'est là l'utilité de cette troisième version.

Ensemble 2

Nous obtenons les matrices d'adjacence suivantes :

	Williamelle	Xanthane	Yaoundé	Zoroark
Ardente	45.88	13.7	58.79	13.8
Bosphore	20.03	34.5	61.51	17.4
Carmen	31.31	102.81	26.55	35.61
Douala	35.58	11.6	69.01	29.44

FIGURE 17 – Matrice d'adjacence de l'ensemble 2

Ensemble 2, le meilleur appariement

Nous avons trouvé le meilleur appariement entre les hôtes et les visiteurs.

```

Matrice du graphe :
      willame Xanthane Yaoundé Zoroark
Ardente 45.880 13.700 58.790 13.800
Bosphore 20.030 34.500 61.510 17.400
Carmen 31.310 102.810 26.550 35.610
Douala 35.580 11.600 69.010 29.440

Affectation de cout minimal 71.980000 :
(Douala, Xanthane, 11.600000)
(Ardente, Zoroark, 13.800000)
(Bosphore, willamelle, 20.030000)
(Carmen, Yaoundé, 26.550000)
  
```

FIGURE 18 – Meilleur appariement pour l'ensemble 2

Nous pouvons remarquer que le coût total de l'affectation de cet ensemble 2 est plus élevé que celui de l'ensemble 1 ($72 > 54.5$) car cet ensemble a été créé pour que des contraintes y figurent. La différence entre les deux ensembles reste à notre sens assez équilibrée compte tenu d'une création distincte.

Ainsi, Carmen et Yaoundé, sont, dans cette version, appareillés malgré une contrainte car ils ont une affinité que l'on peut qualifier de moyennement bonne.

Ensemble 1 et 2

Ces ensembles sont réunifiés afin d'obtenir un exemple aux données plus larges. Nous pouvons également chercher un appariement entre les 16 personnes !

	William	Xavier	Yvan	Zorro	Williamelle	Xanthane	Yaoundé	Zoroark
Alix	13.56	23.21	33.55	18.36	20.43	34.69	39.38	27.42
Brune	20.85	20.25	17.72	15.35	9.31	27.12	34.85	13.53
Charlie	59.49	29.96	13.69	20.1	30.93	20.94	16.04	68.93
Diego	23.96	11.89	23.96	33.2	88.05	27.35	20.28	23.34
Ardente	35.58	20.75	30.24	15.54	45.88	13.7	58.79	13.8
Bosphore	14.07	23.1	21.11	27.18	20.03	34.5	61.51	17.4
Carmen	36.69	35.68	31.31	46.43	31.31	102.81	26.55	35.61
Douala	30.74	18.18	35.42	23.26	35.58	11.6	69.01	29.44

FIGURE 19 – Matrice d'adjacence de l'ensemble 1 et 2

Matrice du graphe :

	William	Xavier	Yvan	Zorro	William	Xanthane	Yaoundé	Zoroark
Alix	13.560	23.210	33.550	18.360	20.430	34.690	39.380	27.420
Brune	20.850	20.250	17.720	15.350	9.310	27.120	34.850	13.530
Charlie	59.490	29.960	13.690	20.100	30.930	20.940	16.040	68.930
Diego	23.960	11.890	23.960	33.200	88.050	27.350	20.280	23.340
Ardente	35.580	20.750	30.240	15.540	45.880	13.700	58.790	13.800
Bosphore	14.070	23.100	21.110	27.180	20.030	34.500	61.510	17.400
Carmen	36.690	35.680	31.310	46.430	31.310	102.810	26.550	35.610
Douala	30.740	18.180	35.420	23.260	35.580	11.600	69.010	29.440

Affectation de cout minimal 119.270000 :

(Douala, Xanthane, 11.600000)
 (Charlie, Yvan, 13.690000)
 (Ardente, Zoroark, 13.800000)
 (Alix, Zorro, 18.360000)
 (Carmen, Yaoundé, 26.550000)
 (Brune, William, 9.310000)
 (Bosphore, William, 14.070000)
 (Diego, Xavier, 11.890000)

FIGURE 20 – Meilleur appariement pour l'ensemble 1 et 2

Avons-nous obtenu des scores proches ?

En somme, en analysant les résultats (matrice d'adjacence CSV pour l'ensemble 1 et 2), nous pouvons déterminer à quel point les scores obtenus sont proches ou non.

Appréciation directe

En jetant un oeil rapidement aux résultats, nous voyons que les résultats sont assez proches, s'étendant de 11 à 102 au plus.

Variance et écart-type

Nom	Moyenne	Variance	Écart-type
Alix	26.32	80.66	8.98
Brune	19.87	64.98	8.06
Charlie	32.51	425.24	20.62
Diego	31.50	558.39	23.63
Ardente	29.29	275.85	16.61
Bosphore	27.36	229.14	15.14
Carmen	43.30	611.46	24.73
Douala	31.65	297.85	17.26

FIGURE 21 – Tableau de variance et d'écart-type

Il n'y a pas d'écart trop impressionnant compte tenu du fait que l'ensemble 2 a été

créé pour qu'il y ait plus de contraintes, cela expliquant l'écart. Néanmoins il reste normal au vu de l'objectif de l'algorithme d'implémenter les contraintes comme des préférences. Certaines paires n'ont donc pas pu compenser avec l'affinité qu'elles avaient, les points communs étant trop peu nombreux.

Moyenne et médiane

```
Statistiques globales sur toutes les valeurs :
Moyenne globale : 30.23
Médiane globale : 26.84
```

FIGURE 22 – Moyenne et médiane

La moyenne et la médiane sont assez proches, donc on ne peut pas dire que certaines affinités "tirent" vers le bas ou le haut les autres, car les scores sont assez équilibrés.

Avec ces éléments, nous pouvons affirmer que **nous avons obtenu des scores proches avec cet algorithme flexible et équilibré.**

4.5 Conclusion

Cette troisième version nous a permis de voir sous un autre angle la gestion de contraintes rédhitoires en les interprétant comme des préférences, en utilisant un algorithme donnant des scores équilibrés aux contraintes et affinités, résultant en arêtes possibles à emprunter dans le graphe biparti si une affinité suffisante a permis de compenser les contraintes, à l'image des paires énoncées en exemple.

Merci d'avoir consulté cette présentation