MAXIME GEAY – DOMINIQUE JEANCLER – JULIEN MAKSOUD – CHLOE RICHARD – ***MS BGD 20-21***

Analyse en Graphes du Réseau de Transport Francilien

L'idée du projet était de faire une étude en graphe des réseaux de transports ferrés et routiers de la région parisienne. Nous avons récupéré un jeu de données de https://github.com/ComplexNetTSP/MultilayerParis et avons tenté de les analyser d'abord séparément puis par groupes. A partir du calcul de métrique propre aux graphes, nous nous sommes intéressés à la typologie de réseaux. Par la suite nous avons exploré plusieurs algorithmes nous paraissant pertinents autour de problématiques liées aux transports.

TRAITEMENT DES DONNEES

Les données récupérées étaient des CSV. Afin de les rendre compatibles à l'étude de graphes, nous les avons chargés en Pandas DataFrames puis utilisé **NetworkX** afin de créer un graph contenant les nœuds et edges.

- Les nœuds contenaient NodeID, Latitude, Longitude, Layer (Métro/Tram/RER)
- Les edges contenaient EdgeID, SourceNodeID, TargetNodeID, Direction, Layer (Métro/Tram/RER)

Afin d'enrichir les données Edge, nous avons calculé la distance euclidienne des edges en se basant sur les Lat/Long. **Nous nous sommes par la suite rendus compte que cela n'était peut-être pas nécessaire, car les librairies savent à priori calculer directement les distances. Nous n'avons donc pas utilisé ces prétraitements.**

Pour les transports en commun, l'immense majorité du graph était non-dirigé (les lignes vont dans les 2 sens à quelques exceptions près où l'on a une boucle) Le graph obtenu était donc dirigé.

Afin de rendre les données compatibles avec **scikit-network**, nous avons exporté le graph en fichier graphml puis l'avons réimporté au format **scikit-network**.

ANALYSE DE CENTRALITE DU RESEAU DE METRO – CRITERE DE KATZ

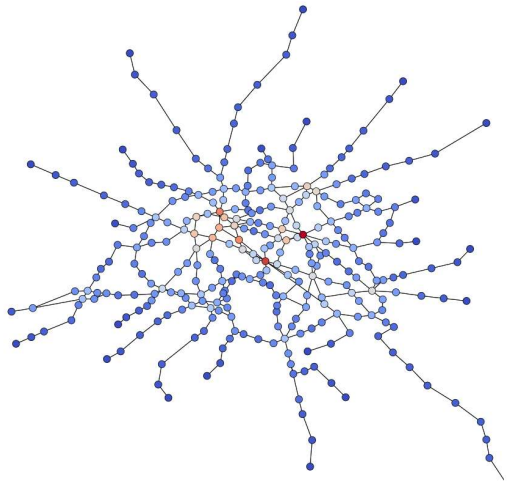


Fig : Distribution des scores de Katz

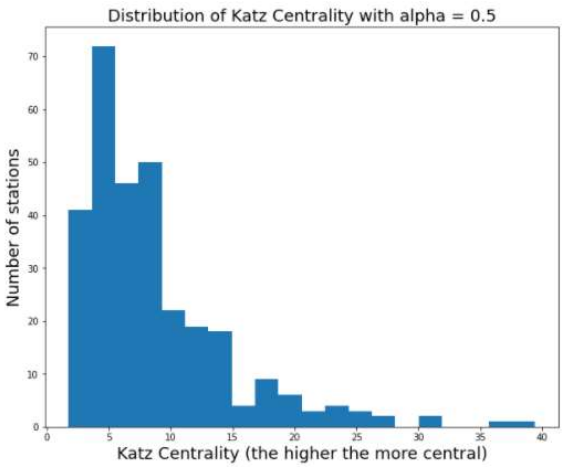


Fig : Histogramme des scores

Avec les paramètres par défaut, on repère sans difficulté la centralité de Chatelet et République. Dans une moindre mesure, Bastille, Gare de l'Est, Stalingrad et une polarisation dans le quartier de Saint Lazare

On voit que la centralité est calculée à l'aide du second voisin. On constate la centralité de République et Châtelet quelque soit le paramètre alpha. Quand alpha tend vers 0 (par exemple alpha = 0.1, Saint Lazare reste centrale

mais de manière moins prononcée. On peut donc dire que Chatelet et République sont "centrales" du fait de leurs nombreux premiers voisins. Ce sont les stations qui ont le plus de connections.

Par contre, la centralité de Katz observée dans le quartier de Saint-Lazare semble être liée au fait de nombreuses connections réparties sur différentes stations alentours.

ANALYSE DU RESEAU ROUTIER

Il est impossible d'afficher le réseau routier car celui-ci comporte trop de nœuds et d'edges. Par contre, travailler avec des clusters permet de mettre en évidence les caractéristiques du réseau routier.

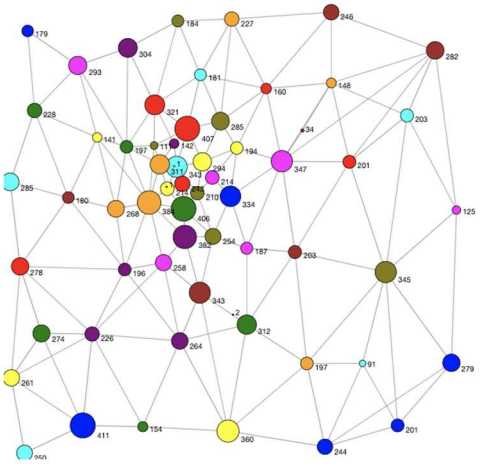


FIg : Aggrégation en Clusters du Réseau Routier avec l'algorithme de Louvain

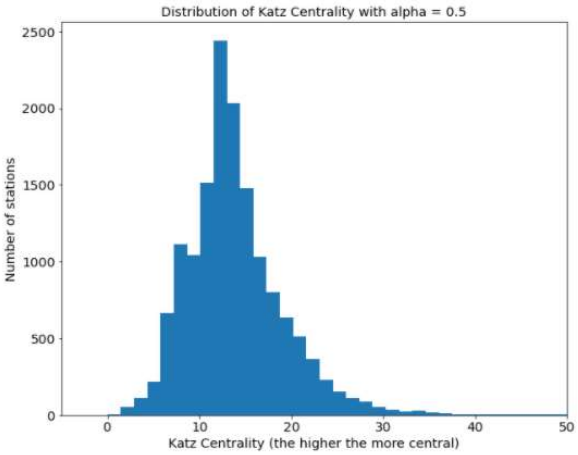


Fig : Distribution des scores de Katz sur le réseau routier

Le réseau comporte 14807 noeuds et 44559 edges. Nous utilisons donc un clustering de Louvain afin de représenter le réseau. On constate que la densité en nombre de routes est plus élevée sur Paris et proche banlieue.

On constate également la flexibilité du réseau routier. Sur une région telle que l'IDF, le réseau de routes est dense et permet d'aller d'un cluster à un autre de manière assez directe. Toutefois, au fur et à mesure que l'on s'éloigne de Paris, la densité de routes diminue. Dans le sud-est du graph, il manque un nœud entre les clusters 91 et 279.

Cette diminution est plus marquée à l'est (départements 94) qu'à l'ouest (92). A l'est de Paris, le réseau est moins dense qu'à l'ouest. On peut penser à une corrélation avec le développement économique historique et la concentration des patrimoines plus importants dans l'ouest parisien qu'à l'est.

COMPARAISONS ENTRE LES DIFFERENTS TRANSPORTS ET RESEAU ROUTIER

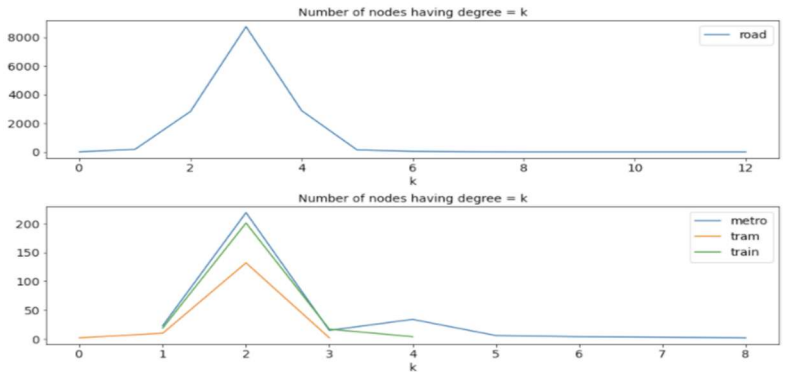


Fig : Nombre de nœuds en fonction du nombre de degrés.

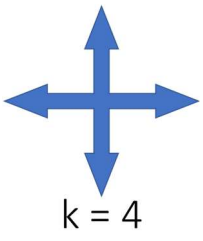
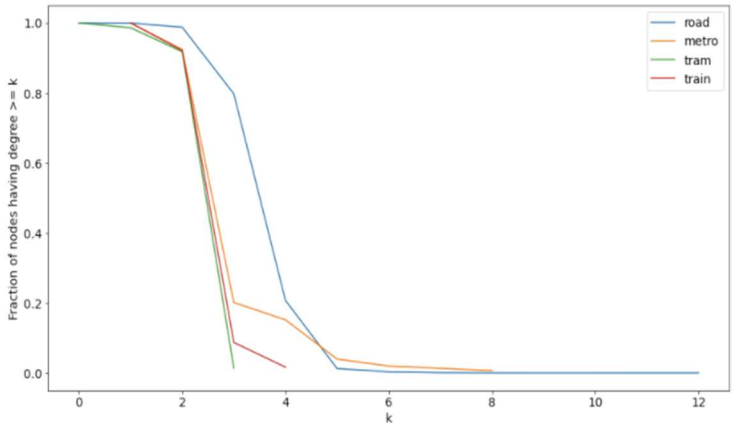


Fig : Schéma de nœud routier correspondant au cas k = 4

On constate que pour les routes, il est beaucoup plus fréquent d'avoir des degrés élevés sur chaque nœud. En effet, un simple croisement de rue correspond à k = 4. Tandis que les croisements de métro sont déjà beaucoup plus rares car beaucoup plus complexes à mettre en œuvre.



Le réseau routier comporte beaucoup plus de nœuds que les transports en commun.
Le cas de k = 4 correspond à un croisement entre deux routes, d'où le fait que 20% des nœuds (croisements) ont 4 edges. Ce qui signifie que sur ce réseau, 20% des croisements donnent 4 chemins possibles.
Typiquement, on peut – même dans les zones les plus lâches du graph, trouver des routes qui se croisent.

Fig : Proportion de nœuds du réseau routier VS Degrés

LIMITATIONS : Parmi les métriques que nous avons calculées, il y a pour les routes le closeness, la betweenness ainsi que la degree centrality. La limitation de ce graph est qu'il ne prend pas en compte le fait qu'une route soit « principale » (dans le centre grand axe très emprunté) ou secondaire. Par conséquent, les meilleurs scores de closeness ou de degree centrality étaient trustés par des départementales telles la D50 ou la D910. Mais ces routes ne sont pas considérées comme « principales » en termes de trafic ou stratégique dans le réseau routier francilien.

CLUSTERING DE RESEAU : RESULTATS ET EXEMPLES D'APPLICATIONS

L'utilisation de graph permet de se poser des questions clefs sur :

- L'identification des nœuds importants
- La structuration du réseau
- L'évaluation de besoin de nouveaux nœuds

Nous présentons ci-dessous le résultat de plusieurs algorithmes de classification obtenu via l'outil SciKit Network qui nous paraissent pertinents pour des problématiques familières à un opérateur.

1. Clustering du réseau – Application de Louvain et agrégation par classes

a. Sur le Métro uniquement

Louvain a été appliqué avec les 3 différentes modularités possibles (Dugue, Potts et Newman). Les résultats obtenus sont très similaires en termes de modularité (qualité de clustering). **Le paramètre qui fait vraiment la différence au niveau de la qualité du clustering est clairement tol_aggregation. Quand il est élevé (*tol_aggregation = 0.5*), les clusters sont complètement mélangés. Une bonne valeur serait celle par défaut *tol_aggregation = 0.001***

- Pour Dugue et Newman : Modularité = 0.799
- Pour Potts : Modularité = 0.796

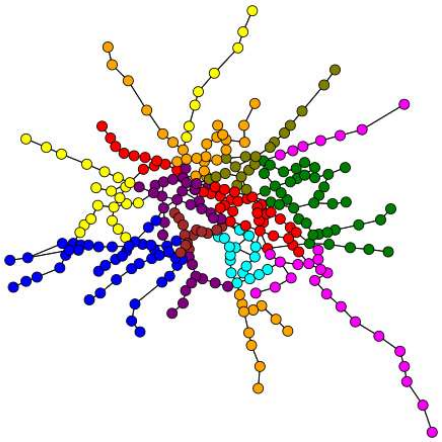


Fig : Clustering de Louvain

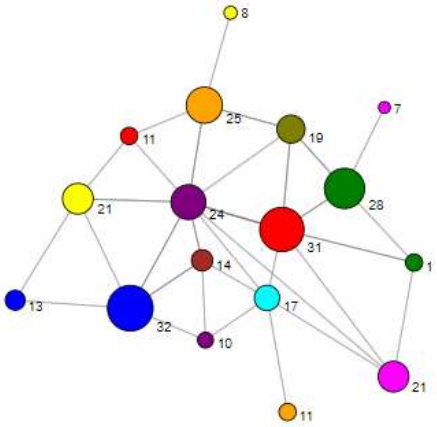


Fig : Clustering de Louvain agrégé

Le clustering de Louvain agrégé est intéressant car il permet de mettre en exergue les liens présents mais surtout **l'absence de certains liens** et la taille de certains clusters.

On remarque également que pour aller du cluster 28 au cluster 32 (du nord est au sud-ouest de Paris), le graph n'est clairement pas optimal **car il manque un edge entre le cluster 31 et le cluster 14**. Cela se ressent quand on essaie d'aller par exemple du 19ème à Porte de Versailles. Le trajet n'est pas optimal. Par contre, du cluster 21 au Cluster 11, le lien est beaucoup plus direct (peut-être inspiré par les lignes 1 et 14 qui traversent la ville).

On remarque également un autre manquement entre les clusters 10, 11 et 21 qui ne sont pas reliés. Également les clusters 7 et 8. Pour ces derniers, la Ligne 16 est en construction et devrait permettre de générer ce lien.

b. Sur le Métro + RER + Tram

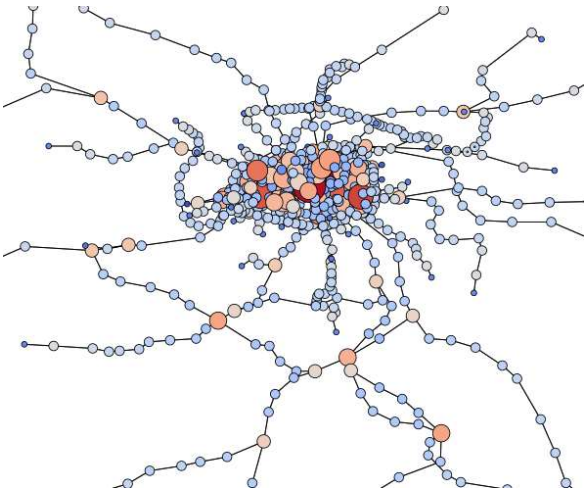


Fig : PageRank appliqué au Métro + RER + Tram

Quand on regarde le PageRank sur le réseau de RER + Métro + Tram, on constate que les scores les plus élevés sont majoritairement concentrés sur Paris. Par contre, on voit au sud et à l'ouest quelques scores intéressants, ce qui prouve que les trajets de banlieue à banlieue restent compliqués car non optimaux. Mais ils restent possibles dans certains cas.
En regardant les plans du Grand Paris, on s'aperçoit que le but est d'améliorer la connectivité entre différentes banlieues qui est trop faible pour le moment. L'ouest parisien reste mieux développé que l'est avec le tram allant de Cachan à la Défense et la construction de la ligne 15 en cours

2. Clustering du réseau – Classification Ascendante Hiérarchique

Après avoir vu l'utilisation de l'algorithme de Louvain et sa forme agrégée débouchant sur une vue à large échelle des régions et de leurs connexions, nous utilisons maintenant un outil de classification hiérarchique. Cet outil va créer des clusters de manière itérative en agrégeant les nœuds ayant le maximum de similarité.

Dans notre cas (x,y comme attributs des nœuds), on va regrouper les nœuds les plus proches et obtenir à la fin du calcul des cluster dont chaque nœud sera atteignable sans sortir du cluster. Les résultats nous semblent pertinents pour des problématiques de gestion de services et matériels partagés comme par exemple :

- Disposition de générateur de courant de secours ne pouvant alimenter plus de n stations
- Localisation d'équipe d'intervention et de proximité (contrainte temps pour se rendre sur site …)

Dans notre cas nous avons utilisé l'algorithme « Paris » pour notre classification hiérarchique et pris en compte deux conditions arbitraires à savoir un maximum de 60 stations par cluster (soit un cinquième de notre réseau) et un ratio de max 2 entre nombre maximum et minimum de station par cluster (à savoir que si le plus petit cluster a k stations, le plus grand cluster n'aura pas plus de 2k stations). Nous obtenons en sortie un découpage homogène de 9 clusters regroupant entre 23 à 45 stations.

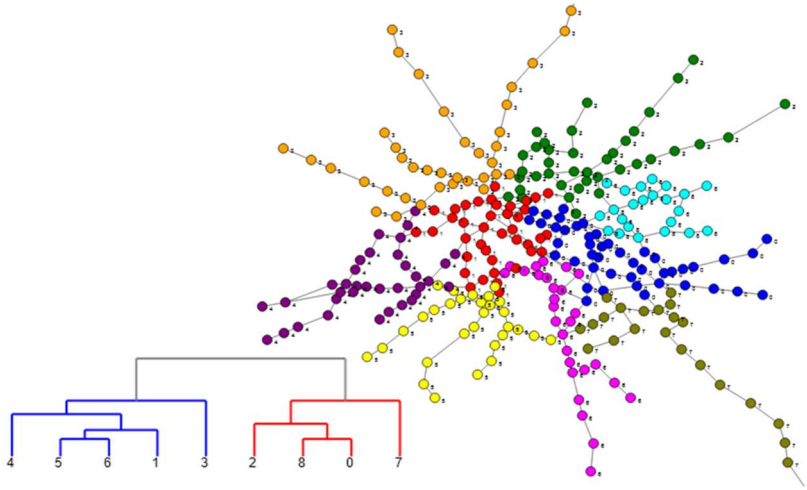


Fig : Classification hiérarchique « Paris » et dendrogramme

On peut aussi imaginer utiliser ce type de calcul pour planifier des opérations lourdes en maintenance nécessitant une fermeture partielle du réseau. On note sur le dendrogramme que les derniers clusters formés (3 ,4 ,7) correspondent aux zones périphériques ayant une plus faible densité de couverture.

3. PageRank – Identification et accessibilité des « Hubs »

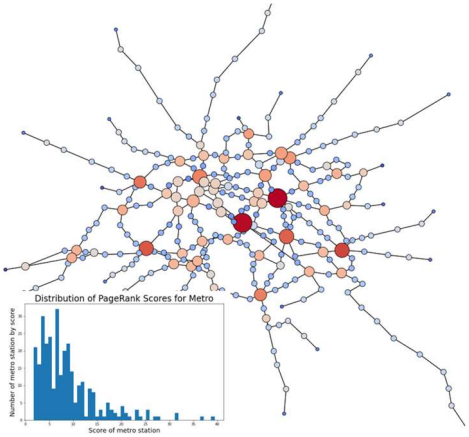


Fig: PageRank et histogramme des scores

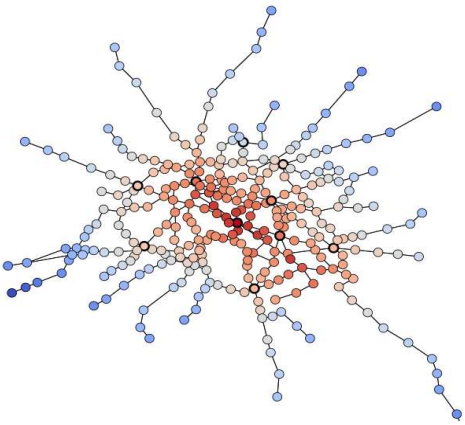


Fig: Distance aux 10 stations ayant score le plus élevé

Sur le graph de droite, les 10 stations aux plus gros scores sont entourées en noir. Il est intéressant de constater que les plus gros scores sont relativement bien distribués (dans Paris intra-muros, les hubs sont bien répartis).

Le fait qu'il n'y ait aucun « hub » en banlieue prouve que les déplacements de banlieue vers banlieue est difficile, et l'absence de liens implique qu'un passage par Paris est obligatoire quand on est en métro.

Des lignes supplémentaires sont en construction (par exemple Ligne 15 de Cachan à La Défense et Ligne 16 de Saint-Denis à Noisy). Ces nouvelles lignes suivent la logique de l'A86 qui est un « périphérique autour du périphérique ». A titre de comparaison, Pékin est dotée de 6 périphériques concentriques.

PLUS COURT CHEMIN

Une application évidente des graphs est le calcul du plus court chemin. A l'image des fluides qui s'écoulent selon le parcours présentant la plus grande pente, l'usager cherchera naturellement à prendre le trajet le plus rapide. La manipulation des graphs permet à l'utilisateur de vérifier par le calcul son intuition et sa connaissance du réseau. Après avoir analysé les réseaux à l'aide des graphs, nous avons donc développé un outil s'appuyant sur le calcul des plus courts chemins sur le réseau métro.

Cet outil retourne le trajet le plus rapide d'un utilisateur et l'affiche sur une carte interactive. On prend en entrée les informations de départ et d'arrivée soit par coordonnées géographique (latitude longitude GPS), soit par adresse postale. Dans notre cas, l 'algorithme retourne le trajet minimisant le nombre de nœuds empruntés. Le lecteur est invité à tester cette fonction sur le notebook joint *(le résultat n'étant pas sauvable en html mais disponible dans un html séparé)*

```
Entrée [207]: #l'utilisateur doit entrer 4 arguments correspodnant aux coordonnées GPS du lieu de départ et arrivée
              #dans l'ordre : latitude_depart,longitude_depart,latitude_arrivee,longitude_arrivee
              stationproche_trajet(48.833399, 2.3261858,48.8751177,2.3276115)

Vous vous rendez à pied la station numéro noeud 300 et sortez à la station numéro noeud 163
Le trajet comporte 13 stations identifiées par noeuds :
[300, 299, 301, 200, 257, 26, 25, 24, 93, 61, 86, 237, 163]
Distance parcoure: 5.32 km pour une durée estimée de 15 minutes
```

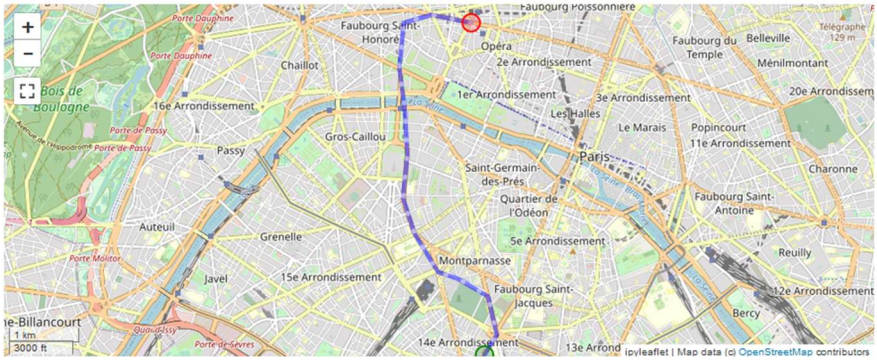


Fig : Plus court chemin métro GPS

```
Entrée [206]: #l'utilisation entre 2 adresses définies en chaine de caractère
              trajet_adresse("37 rue de charonne paris","1 rue du commerce paris")

Vous vous rendez à pied la station numéro noeud 98 et sortez à la station numéro noeud 91
Le trajet comporte 12 stations identifiées par noeuds :
[98, 121, 190, 191, 192, 180, 9, 205, 93, 107, 71, 91]
Distance parcoure: 7.12 km pour une durée estimée de 21 minutes
```

Fig: Plus court chemin métro adresse

Des voies d'amélioration étaient en réflexion sur l'utilisation du plus court chemin à la finalisation du projet :

- Pour une meilleure ergonomie, il aurait été souhaitable d'obtenir le nom des stations. Ce problème provient de la donnée d'entrée et nous avons choisi de concentrer nos efforts sur d'autres sujets.
- Une contrainte portant sur le nombre de changement de ligne aurait été pertinente à intégrer dans le calcul. La difficulté ici est que l'information de ligne n'est pas dans les nœuds mais dans les liens.
- Une plus-value du calcul automatique de trajet aurait été d'intégrer des contraintes sur l'évitement de certain nœud. Dans des circonstances inattendues, telles qu'une panne ou un incident, l'usager peut ainsi avoir une indication immédiate d'un plus court détour qui est pour lui un calcul beaucoup moins intuitif pour lui. Le réseau de transports ferrés ne comptant que quelques centaines de nœuds, l'option envisagée était de faire une copie de l'objet graph en ayant au préalable retiré les nœuds à éviter. Puis faire tourner notre plus court chemin sur cette copie (option peu couteuse)
- D'un point de vue exploitant, il aurait été intéressant d'utiliser des données externes sur les déplacements quotidiens domicile/travail (représentant le facteur prépondérant du trafic en pointe) et simuler les distributions de trajet aux heures de pointes. L'agrégation de ces résultats donnerait une indication des lieux en tension pour penser le développement du réseau et/ou proposer un réseau de complément à moindre coût (bus).
- Enfin, d'un point de vue décideur public (dans un contexte d'encouragement à l'usage des transports collectif) : nous aurions voulu croiser ces simulations sur le réseau des transports en commun avec celui du réseau routier. Ce comparatif aurait pu mettre en évidence des zones prioritaires dans l'investissement de transport collectif.

CONCLUSION

Nous avons pu observer certaines limitations des réseaux de transports parisiens. Les développements mis en place dans le cadre du projet du Grand Paris sont cohérents au vu desdites limitations, et va aider à la création de nouveaux « hubs » en dehors de Paris intra-muros.

Il y aura plus d'options que le sempiternel changement RER B / RER A de Chatelet afin de se rendre de banlieue à banlieue, notamment pour les gens se rendant du sud de Paris vers La Défense.

Le tramway est plus vu comme une solution de désengorgement du réseau existant que comme une réelle extension de celui-ci. En effet, il ne créé pas de « Hub » car circulaire autour de Paris.

Il serait intéressant de faire cette étude sur d'autres villes afin de tester l'efficacité de leurs réseaux de transports et voir s'ils ont été conçus de manière similaire à Paris.

Nous avons eu l'occasion de tester un certain nombre de méthodes (soft clustering par exemple) mais la plupart n'ont pas été concluantes.

Les objets graphes et les métriques associées permettent d'explorer de manière originale les liaisons au sein d'un jeu de données. Il constitue un outil performant dans l'analyse des réseaux.