

Détection de communautés à long terme dans les graphes dynamiques

Thomas Aynaud et Jean-Loup Guillaume

LIP6 – CNRS – Université Pierre et Marie Curie
4 place Jussieu
75005 Paris, France

Résumé. La plupart des graphes de terrain peuvent être décomposés en sous graphes denses appelés communautés. Habituellement, dans des graphes dynamiques, les communautés sont détectées pour chaque instant indépendamment ce qui pose de nombreux problèmes tels que la stabilité ou le suivi de des communautés entre deux décompositions successives. Nous proposons ici une méthode pour trouver une partition unique, de qualité, couvrant une longue période. Cette décomposition peut être trouvée efficacement via une adaptation de la méthode de Louvain et la perte de qualité à chaque instant due à la contrainte de détecter des communautés globales s'avère assez faible.

1 Introduction

L'étude des graphes de terrain a montré qu'ils partageaient un certains nombre de propriétés. En particulier, bien que leur densité soit en général très faible, leur coefficient de clustering est souvent élevé et ils se comportent donc localement presque comme des cliques. Cette particularité peut être expliquée par l'existence de communautés. Il s'agit de groupes de nœuds très densément connectés à l'intérieur mais avec peu de liens vers l'extérieur.

Une formalisation¹ du concept de communautés a été proposée dans Newman et Girvan (2004). Une décomposition en communautés est ainsi une partition de l'ensemble des nœuds qui maximise une fonction de qualité appelée la modularité. Trouver cette partition optimale est NP-complet et de nombreuses heuristiques ont été proposées (voir Fortunato (2009) pour une étude plus exhaustive). Néanmoins, cette définition et ces heuristiques ont été proposées pour le cas de graphes statiques. Or la majorité des graphes de terrain sont dynamiques. Par exemple, des pages sont ajoutées, modifiées ou supprimées constamment sur le web, et donc ne considérer qu'une capture à un instant donné revient à négliger beaucoup d'informations.

C'est pourquoi plusieurs études ont été faites pour intégrer la dynamique dans la détection de communautés. On peut par exemple chercher à détecter une décomposition différente à plusieurs instants et essayer de suivre les communautés entre ces multiples décompositions (Hopcroft et al. (2004); Palla et al. (2007)). L'instabilité des algorithmes de détection de communautés, qui ont tendance à donner des résultats très différents entre deux instants proches, a conduit à proposer d'autres fonctions de qualité. Ainsi, dans Kumar et al. (2006) les auteurs

1. Ce n'est pas la seule, mais elle est largement employée

proposent d'ajouter à la fonction de qualité un terme de stabilité, représentant la proximité entre la partition à t et celle à $t - 1$ et dans Song et al. (2007) est proposé d'ajouter un terme imposant que la partition à t soit également bonne à $t - 1$.

Nous étendons ici cette idée en montrant que pour plusieurs exemples, il est possible de définir et détecter des communautés qui soient bonnes à pratiquement tout instant. Nous allons dans une première partie les définir et proposer un méthode d'optimisation pour les détecter, puis nous verrons sur plusieurs exemples que l'on trouve des communautés effectivement bonnes globalement.

2 Communautés globales

Considérons un graphe dynamique comme une succession de graphes statiques, chacun représentant l'état du réseau à un instant donné et que nous appellerons graphes instantanés. Notons $G_t = (V_t, E_t)$ l'état du réseau à l'instant t et soit V l'union de tous les V_t . Les pas de temps seront dans l'ensemble $0, 1, \dots, T_m$. Enfin, $Q(G_t, \pi)$ représentera la modularité de la partition π restreinte aux nœuds de G_t pour le graphe G_t et nous l'appellerons modularité instantanée. La détection de communautés cherche habituellement à maximiser $Q(G_t, \pi)$. Nous allons chercher à maximiser la modularité globale :

$$Q_{glob}(G, \pi) = \sum_{t=0}^{t=T_m} Q(G_t, \pi)$$

Afin de trouver des partitions de bonne modularité globale nous allons modifier légèrement la méthode de Louvain qui est un algorithme initialement prévu pour maximiser la modularité instantanée (voir Blondel et al. (2008)).

Cette approche est composée de deux phases qui sont répétées de manière itérative jusqu'à obtenir un maximum local de la modularité. Initialement, les sommets sont tous seuls dans leur communauté. La première phase est constituée de plusieurs itérations. Une itération consiste à considérer tous les sommets un par un et à déplacer le sommet considéré dans la communauté voisine maximisant le gain pour la fonction de qualité utilisée si ce gain est positif. Ces itérations sont répétées jusqu'à ce qu'aucun sommet ne soit bougé. Commence alors la deuxième phase où le graphe est modifié pour représenter le graphe entre les communautés trouvées : on construit un nouveau graphe avec pour ensemble des nœuds les communautés et un lien entre deux communautés de poids la somme des poids des liens entre les sommets qui les composent. Enfin, on recommence à la première phase sur ce nouveau graphe et on alterne ces deux phases jusqu'à ce que l'on n'obtienne plus aucun gain de qualité.

Cet algorithme est particulièrement efficace dans le cas de l'optimisation de la modularité instantanée car le gain obtenu en bougeant un sommet dans une communauté voisine peut être calculé très rapidement avec des informations locales au nœud. Cette particularité se retrouve pour la modularité globale : le gain pour le graphe dynamique est égal à la somme des gains sur chaque instantané. On peut donc voir l'algorithme comme une application en parallèle de la méthode de Louvain sur chacun des graphes instantanés composant le graphe dynamique, le choix du mouvement effectué étant effectué en maximisant la somme des gains. Le temps de calcul est de l'ordre de la somme des temps de calculs de Louvain sur chaque graphe instantané.

3 Résultats

Afin d'évaluer les résultats de notre modification de l'algorithme de Louvain nous l'avons appliquée sur deux graphes de terrain dynamiques :

- Le premier est un réseau de blogs monitorés pendant quatre mois. Initialement, le réseau est vide et on lui ajoute chaque jour les nœuds et les liens vus.
- Le second, tiré de Pansiot et al. (2009), représente les connexions entre les routeurs multicast. L'outil `mrinfo` permet d'interroger un tel routeur et de lui demander ses voisins. Tous les jours, un réseau entre ces routeurs a été construit en les interrogeant de proche en proche à partir d'une source donnée.

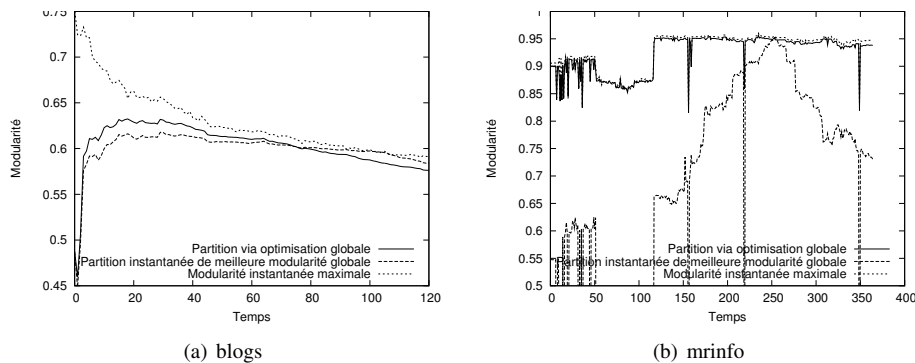


FIG. 1 – Modularité instantanée de certaines partitions au cours du temps

La figure 1 présente la modularité instantanée à chaque instant de la meilleure partition obtenue via la méthode de Louvain classique², de la partition parmi les partitions trouvées par l'algorithme de Louvain sur tous les instantanés qui a la meilleure modularité globale³ et de la partition obtenue en optimisant la modularité globale.

Sur chacun des réseaux, on constate que la partition obtenue en maximisant la modularité moyenne est souvent très proche en qualité instantanée de celle obtenue en utilisant un algorithme statique, qui n'a pas toutes les contraintes liées au fait d'être bonne sur d'autres réseaux. Cela montre que la partition trouvée reflète bien la structure du réseau. Si l'on regarde par contre la partition instantanée ayant la meilleure qualité globale, on constate un comportement différent suivant les réseaux. Avec le réseau de blog, cette partition a une qualité globale très proche de celle de la partition globale. C'est lié au fait que l'on agrège les données et que donc le dernier réseau contient en partie tous les précédents. La structure du réseau évoluant peu, une bonne partition à un instant s'avère être une bonne partition à chaque instant. En revanche, quand on considère les réseaux `mrinfo`, on voit que ce résultat n'est plus vrai. Maximiser la somme des modularités apporte un vrai gain par rapport à choisir la partition instantanée de meilleure qualité globale.

2. C'est en quelque sorte la modularité instantanée atteignable maximale

3. Les nœuds n'existant pas à l'instant considéré sont tous regroupés

4 Conclusions

Nous avons donc montré qu'il est possible et pertinent de ne pas chercher des communautés bonnes à un instant donné mais d'en chercher des bonnes tout le temps ou sur une longue période. L'algorithme de Louvain peut être modifié dans ce sens.

Ces partitions posent de nombreuses questions. Certaines communautés peuvent regrouper par exemple des nœuds existant à des instants différents mais fortement liés à un même groupe. La durée sur laquelle porte la sommation a aussi certainement une influence importante. Enfin, un des défauts importants de la sommation est de ne pas tenir compte de l'ordre : il n'y a aucune causalité entre les graphes et par exemple l'enchaînement G_1, G_2, G_3 devrait avoir la même décomposition que G_3, G_1, G_2 .

Références

- Blondel, V. D., J.-L. Guillaume, R. Lambiotte, et E. Lefebvre (2008). Fast unfolding of communities in large networks. *J. Stat. Mech* 10008, 1–12.
- Fortunato, S. (2009). Community detection in graphs. *Physics Reports*.
- Hopcroft, J., O. Khan, B. Kulis, et B. Selman (2004). Tracking evolving communities in large linked networks. In *National Academy of Sciences of the United States of America*, Volume 101, pp. 5249. National Acad Sciences.
- Kumar, R., A. Tomkins, et D. Chakrabarti (2006). Evolutionary clustering. In *In Proc. of the 12th ACM SIGKDD Conference*.
- Newman, M. E. J. et M. Girvan (2004). Finding and evaluating community structure in networks. *Physical Review E* 69(2), 26113.
- Palla, G., A.-L. Barabasi, et T. Vicsek (2007). Quantifying social group evolution. *Nature* 446, 664–667.
- Pansiot, J., P. Mérindol, B. Donnet, et O. Bonaventure (2009). Extracting Intra-Domain Topology from mriinfo Probing. In *Passive and Active Measurement*.
- Song, X., Y. Chi, B. L. Tseng, D. Zhou, et K. Hino (2007). Evolutionary spectral clustering by incorporating temporal smoothness. In *Proceedings of the 13th ACM SIGKDD conference*.

Summary

Complex networks can usually be divided in dense subnetworks called communities. In dynamic networks, communities are often detected independantly at several timesteps and this causes trouble regarding the stability and the tracking of communities across timesteps. We propose here a new method to detect communities that are good for (almost) every timesteps. The decomposition is good on average and not just for one particular time. It can be detected with a modification of the Louvain method and we show that the loss of instantaneous modularity can be low despite the constraint of maximizing globally.