

# Proposition de stage M2 – STL

## PALS: Patterns of Link Streams

(Motifs dans les Flots de Liens)

### Encadrants

— **Frédéric Peschanski**

Laboratoire d'informatique de Paris 6 – Equipe APR  
email : frederic.peschanski@lip6.fr

— **Matthieu Latapy**

Laboratoire d'informatique de Paris – Equipe ComplexNetworks  
email : matthieu.latapy@lip6.fr

### Contexte et objectifs

Les flots de liens (*link streams*) sont des séquences de triplets permettant d'observer l'évolution d'un graphe dynamique. Chaque triplet de la forme  $(t, a, b)$  signifie qu'un lien entre deux entités  $a$  et  $b$  a été observé au temps  $t$ . Le principal intérêt de ce formalisme est de combiner les aspects structurels et les aspects temporels des phénomènes dynamiques. Il s'agit, en quelque sorte, de combler un vide entre d'une part la théorie des graphes – focalisés sur la structure – et d'autre part le traitement du signal, en particulier les séries temporelles.

Des travaux récents de l'équipe ComplexNetworks du LIP6 ont permis d'identifier des propriétés importantes sur les flots de liens, notamment la *densité de contacts* [3] ainsi que les *cliques maximales* [6].

Dans le cadre de ce stage nous nous intéressons à la problématique de **détection de motifs dans les flots de liens**. Par exemple, un motif peut consister en une forme de triangle soit la détection d'un lien établi entre deux noeuds  $a$  et  $b$  à l'instant  $t_1$  donc un triplet  $(a, b, t_1)$ . Plus tard on observe l'établissement d'un lien  $(a, c, t_2)$  et finalement un lien  $(b, c, t_3)$ . Bien sûr de nombreux autres événements ont pu se produire entre  $t_1$  et  $t_3$ , la détection de ce simple triangle peut donc en fait être assez complexe. Dans un premier temps, l'objectif est d'élaborer un langage de spécification des motifs de flots de liens, dans l'esprit des expressions rationnelles. Ce langage doit être expressif, donc

permettre la spécification simple de motifs complexes, tout en restant raisonnable d'un point de vue algorithmique. Se baser directement sur les grammaires de graphes [5] nous semble par exemple impossible : la détection de motif correspond à l'isomorphisme de sous-graphe, un problème intrinsèquement difficile (NP-complet). D'un point de vue structurel, on s'intéressera donc à des motifs reconnus de façon incrémentale et possédant des spécificités permettant leur détection rapide, soit de façon complémentaire à des motifs intrinsèquement probabilistes (en acceptant donc que les algorithmes de détection produisent des faux négatifs).

Les motifs qui nous intéressent sont également – et avant tout – de nature dynamique : un exemple simple est celui d'un lien « clignotant », apparaissant et disparaissant à intervalles réguliers. Pour ces aspects dynamiques, notre point de départ est une variante des automates à mémoire fini (FMA, cf. [4]) que nous avons défini pour raisonner sur les ressources dynamiques dans les systèmes concurrents [2]. Ces automates permettent, contrairement au automates finis classiques, une forme d'apprentissage par stockage temporaire d'informations. Ceci augmente grandement l'expressivité du modèle au prix d'une algorithmique plus complexe (mais toujours efficace). Pour intégrer l'aspect temporel des flots de lien, notre principale piste concerne les expressions et automates temporisés [1]. Ceci suggère la fusion entre les modèles temporisés et les modèles à mémoire, qui conduirait ainsi à une classe nouvelle et selon nous très intéressante d'automate hybride.

## Tâches à réaliser

Dans un premier temps, une étude bibliographique concernant la détection de motif dans les structures graphiques sera réalisée. Cette étude abordera également les différents modèles d'automates envisagés pour le projet : automate à mémoire finie et automates temporisés. Il sera important d'intégrer les deux points de vue : expressivité du langage reconnu d'un côté, et efficacité algorithmique de l'autre côté. Dans la deuxième partie du stage, on s'attachera à concevoir un langage (ou une famille de langages) de spécification pour les motifs de flots de liens, associé à une algorithmique efficace de reconnaissance. Un prototype d'outil de détection de motifs sera finalement réalisé et expérimenté sur des flots de liens extraits de données réelles. Dans l'hypothèse d'obtention de bons résultats, il est fort probable qu'un article de recherche sur le sujet sera rédigé par le stagiaire et ses encadrants, et soumis à une conférence internationale de haut niveau.

## Intérêt et apports du stage

Ce stage représente une étape importante dans l'étude du formalisme des flots de liens, qui représente une orientation technique stratégique pour l'équipe *ComplexNetworks*. Le potentiel de publication sur le sujet est donc très bon. Ce

stage offre ainsi un panorama complet des activités de recherche : études bibliographique et théorique en amont, conception algorithmique puis implémentation prototype des solutions retenues. Plusieurs laboratoires, en complément du LIP6, s'intéressent aux flots de liens, le potentiel de poursuite en thèse de doctorat sur le sujet est donc fort.

## Références

- [1] Rajeev Alur and David L. Dill. A theory of timed automata. *Theor. Comput. Sci.*, 126(2) :183–235, 1994.
- [2] Aurelien Deharbe and Frédéric Peschanski. The omniscient garbage collector : A resource analysis framework. In *ACSD 2014*, pages 102–111. IEEE Computer Society, 2014.
- [3] Noé Gaumont, Clémence Magnien, and Matthieu Latapy. Finding remarkably dense sequences of contacts in link streams. *Social Netw. Analys. Mining*, 6(1) :87 :1–87 :14, 2016.
- [4] Michael Kaminski and Nissim Francez. Finite-memory automata. *Theor. Comput. Sci.*, 134(2) :329–363, 1994.
- [5] Grzegorz Rozenberg, editor. *Handbook of Graph Grammars and Computing by Graph Transformations, Volume 1 : Foundations*. World Scientific, 1997.
- [6] Tiphaine Viard, Matthieu Latapy, and Clémence Magnien. Computing maximal cliques in link streams. *Theor. Comput. Sci.*, 609 :245–252, 2016.