Graph Modular Decomposition as a Means of Compression

Advisors: Fabien de Montgolfier and Stefano Zacchiroli

Place: IRIF, Université de Paris

PhD funding duration: three years

Many science fields are dealing with *relations*, such as the ones between software files and versions, social interactions, computer networks, the Web, and so on. These relations are usually modeled as graphs. In the "Big Data" era, those graphs can become so huge that reducing their size while keeping them usable is a common need. The general field of this thesis is the design of *compressed* graph representations (i.e., fewer bits per edge or node) that preserve the ability to answer specific queries (like adjacency and distance) without requiring graph decompression. The specific goal of this thesis is to conceive and implement *modular decomposition* algorithms, and to build on them to obtain practically usable compressed graph representations.

A module is a subset M of vertices such that any vertex $x \notin M$ is either a neighbor of every vertex of M, or of none of them. Reducing a module to one vertex yields a lossless compression of the graph. A definition of *modular decomposition* suitable for many objects has been proposed: graphs may be directed, or edge-labeled, or bipartite... It has been defined for hypergraphs also.

During this PhD thesis, the candidate shall:

- *define* new modular decomposition variants, to deal with new objects, or to decompose them further,
- design efficient *algorithms* for computing these decompositions,
- *implement* those algorithms into existing graph manipulation frameworks.

The use case we will focus on is Software Heritage—the largest, freely accessible archive of software source code in the world, together with its development history as captured by Version Control Systems (VCS) like Git, Mercurial, etc.—whose data model is a typed Merkle directed acyclic graph (DAG), enriched with properties on nodes and edges. The Software Heritage graph contains ≈20 billion nodes and ≈200 billion edges, and is handled in a compressed format, using state-of-the-art graph compression frameworks. This allows to perform *scale-up* graph processing and exploitation. Given the size of the graph though, it is worth trying to *distribute* it across several machines while retaining its compressed representation.

While the current approach guarantees lossless compression, it would also be interesting to investigate *lossy compression*, using approximate modules (there may exist a few extra or missing edges between them and the rest of the graph).

This PhD thesis is supported by the French ANR project CORÉGRAPHIE.

Desired skills to pursue this thesis:

- algorithmics: good knowledge of general-purpose algorithms and graph algorithms in particular
- development: Linux system programming, and development experience in one or more programming languages among C, Java, and Python
- experience in coding efficient algorithms on large graphs, or in using large-scale graphs frameworks would be a plus

How to apply: Send curriculum vitae, cover letter, statement of university grades, and optional recommendation letters, to Fabien de Montgolfier `<fm@irif.fr>` and Stefano Zacchiroli `<zack@irif.fr>`.

This thesis offer will be available until the ideal candidate is found. The next preferential deadline for application is **29 June 2021**.