since Oct 2012 doctoral researcher with the Parallel Computing Group, Institute of Theoretical Informatics, Karlsruhe Institute of Technology

2012 Diplom in computer science, Karlsruhe Institute of Technology
Content | Overview

- graph algorithms for **large-scale network analysis**
- original research on the subproblems of
  - **community detection**: fast parallel heuristics
  - **complex network sparsification**: experimental comparison of structure-preserving methods
  - **generative models**: realistic generator using multiscale methods
- assembling effective solutions into the open-source software package **NetworKit**
Introduction | Network Analysis

phenomenon → abstraction → network concept → representation → network data

network model

[Brandes 2012]

complex system

entities and relations

graph data
Methods, e.g.

- measure node centrality → rank nodes by structural importance
  - e.g. PageRank (C)
- identify modular subgraphs
  - e.g. community detection methods
- develop realistic models for networks
Introduction | Network Analysis

**goal**

- exploratory data analysis
- link network structure to system behavior and function

**challenges**

- many methods pioneered on small (social) network
- how to make them scalable for large data sets?
Methods | Overview

- **algorithm engineering:**
  - iterative approach
  - real inputs
  - typical-case performance

- achieving **scalability** through
  - parallelism
  - approximations and heuristics

- **transfer** algorithm research to software product
  - → **correctness** and **practicality**
  - apply good software engineering practices
  - build a community of users and developers
Methods | Overview

- **algorithm engineering:**
  - iterative approach
  - real inputs
  - typical-case performance

- **achieving scalability** through
  - **parallelism**
  - **approximations and heuristics**

- **transfer** algorithm research to software product
  - → **correctness** and **practicality**
  - apply good software engineering practices
  - build a community of users and developers
Methodology | Example: Betweenness Centrality

- betweenness centrality: nodes are as important the fraction of shortest paths passing through them

\[ b(v) = \sum_{s \neq v \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}} \]

- best algorithm for exact computation: \( O(nm) \) time [Brandes 2001] → does not scale!
Example: Betweenness Centrality

- **but**: exact solution not required to appreciate network structure and identify high-ranking nodes

- **idea**: sample only a fraction of shortest paths and extrapolate [Sanders, Geisberger 2008] [Riondato, Kornaropoulos 2015]
Methods | Overview

- **algorithm engineering:**
  - iterative approach
  - real inputs
  - typical-case performance

- achieving **scalability** through
  - **parallelism**
  - approximations and heuristics

- **transfer** algorithm research to software product
  - → **correctness** and **practicality**
  - apply good software engineering practices
  - build a community of users and developers
Task | Community Detection

- Identify internally dense, externally sparse subgraphs → modules of the complex system
- Ill-defined problem, often formalized through objective function modularity:

\[
mod(\zeta, G) := \sum_{C \in \zeta} \left( \frac{\omega(C)}{\omega(E)} - \frac{\text{vol}(C)^2}{4\omega(E)^2} \right)
\]

- Optimizing modularity is NP-hard → efficient heuristics needed
Results | Parallel Community Detection Heuristics

- two effective parallel heuristics
  - PLP: parallel label propagation (based on [Raghavan 07])
  - PLM: parallelization of Louvain method (based on [Blondel 08])
- outperform state-of-the-art implementations
- e.g. 3 billion edge graph can be processed in < 3 minutes on parallel workstation

[Staudt, Meyerhenke 2015 in Transactions on Parallel and Distributed Systems]
Task | Sparsification

- Can we reduce amount of data but retain qualitative aspects?
- → sparsification methods filter edges while preserving some network properties

Jazz musicians collaboration network sparsified → highlights community structure
Results | Sparsification

- experimental study comparing existing and novel sparsification methods
- approach: edge ranking and filtering
- \( \rightarrow \) classes of methods
- \( \rightarrow \) new linear-time method LD performs well across a wide range of properties

LD method preserves edges to local hubs

positive correlation of edge scores:
- \# triangles
- modularity
- methods favoring edges within dense subgraphs
- LD favors edges between dense regions

[Lindner, Staudt, Hamann, Meyerhenke, Wagner 2015 at ASONAM]
Task | Generative Network Models

- design of algorithms that generate graphs similar to observed networks
- network scientists: understand generative processes
- algorithm engineers: obtain realistic synthetic data

simple generative models: Barabasi-Albert, Dorogovtsev-Mendes, Watts-Strogatz
**Generative Network Models**

- **multiscale generator** [Gutfraind, Meyers, Safro 2015] applies multi-level coarsening and editing strategy
  - input: real network
  - output: randomized/scaled replica with similar properties
- current work: scalable parallel implementation and algorithm improvements

network from medical data used in disease modelling: original (left) and replica (right)

[TBA: Staudt, Gutfraind, Safro 2016]
NetworKit | Architecture

- Python interface
- C++/OpenMP core
- → usability and performance
NetworKit Performance

- extensive set of analytics algorithms
- implementations among the fastest available

[Staudt, Meyerhenke, Sazonovs 2016, under review for Network Science]
NetworKit | Performance

- extensive set of analytics algorithms
- implementations among the fastest available

[Staudt, Meyerhenke, Sazonovs 2016, under review for *Network Science*]
NetworKit | Exploratory Network Analysis

- NetworKit generates reports with network structure overview
- → hint at hypotheses

here: internet router topology network
to be released... next week
NetworKit | More Info

- **current paper (preprint):** http://bit.ly/1LSh1Up
- **project website:** https://networkit.iti.kit.edu/
- **email:** christian.staudt@kit.edu