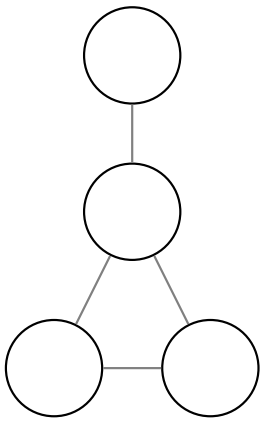
Modelling and Optimisation with Graphs

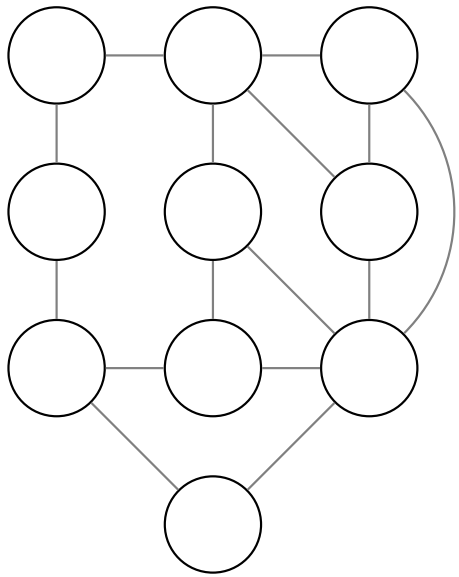
Ciaran McCreesh

University of Glasgow

Finding Subgraphs

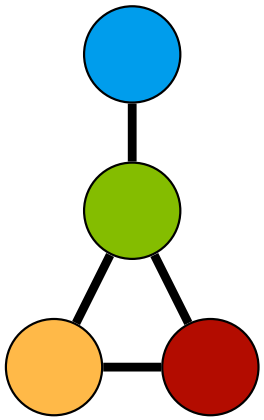
- Given a little *pattern* graph and a large *target* graph, find a copy of the pattern inside the target, or show that no copy exists.

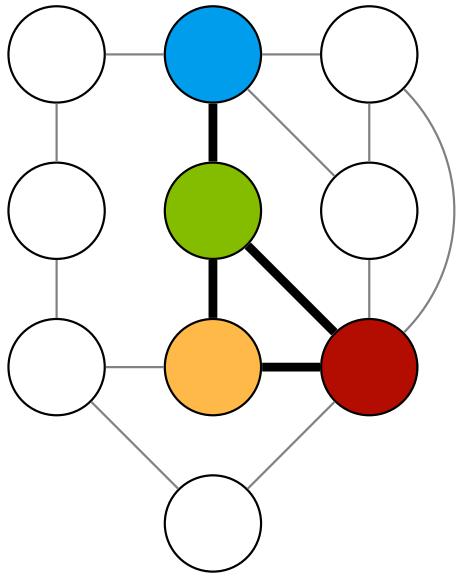




Finding Subgraphs

- Given a little *pattern* graph and a large *target* graph, find a copy of the pattern inside the target, or show that no copy exists.





Who Cares?

- Chemistry and biology (molecules and proteins are graphs)
- Social network analysis (relationships are graphs)
- Fraud detection (financial transactions are graphs)
- Computer vision (images can be turned into graphs)
- Kidney exchange (patient / donor compatibility is a graph)
- Big data queries (dynamic structures are graphs)
- Disease transmission (transmission routes are graphs)
- Algebra (correspondence between semigroups and graphs)
- Verification of computer programs (turned into graphs)
- Design of communication protocols (via lots of maths)

Side Constraints

- Actually, the question is often "subgraphs plus something else".
- Standard optimisation techniques (constraint programming, boolean satisfiability, integer programming, ...) scale very poorly with graphs.
- Adapting dedicated graph algorithms requires an expensive specialist and a lot of work.

Modelling and Optimisation with Graphs

- Develop an open source library for a flexible subgraph algorithm, so scientific advances do not remain unused.
- Enable communication between graph algorithms and general purpose solvers.
  - "Subgraph maps modulo theories".
  - Needs learning subgraph algorithms with external failure.
- Integration with a high level optimisation modelling language.

Modelling and Optimisation with Graphs

- Develop an open source library for a flexible subgraph algorithm, so scientific advances do not remain unused.
  - Primarily about high quality **algorithm engineering**.
- Enable communication between graph algorithms and general purpose solvers.
  - "Subgraph maps modulo theories".
  - Needs learning subgraph algorithms with external failure.
  - Substantial **scientific and mathematical** challenge, but also requires careful engineering to ensure practicality.
- Integration with a high level optimisation modelling language.
  - Via case studies: **collaborative** (academic and industrial) work.

Why I Can Do This

- Technical skills:
  - Maths.
  - Programming / algorithm engineering.
  - Empirical algorithmics.
- Collaborations:
  - Research trips to U. of Melbourne (Jim Gatheral Scholarship) and U. of Toronto.
  - Co-authors at INSA Lyon, U. of British Columbia, U. of Edinburgh; work underway with U. of St Andrews.
- Writing and publication:
  - Top Artificial Intelligence (IJCAI) and Optimisation (CP, CPAIOR) conferences (peer-review and published proceedings).
  - ACM Transactions on Parallel Computing.

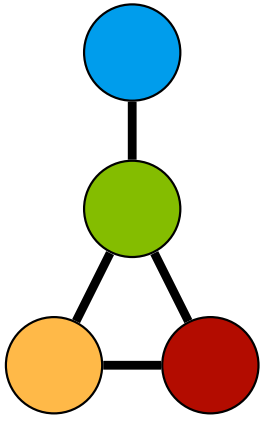
Other Things I've Done

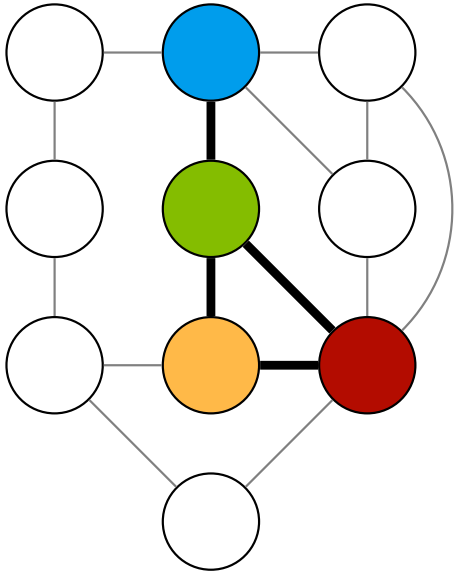
- Tutoring, Lecturing and Project Supervision:
  - Five lectures (three on new material of my own) to a 4/M class.
  - Tutoring for 2, 3 and M classes.
  - Co-supervised two final year and one masters project.
- Research students class representative.
- Organisation:
  - First Workshop on Teaching Constraint Programming.
  - MATCH UP.
- Talks:
  - Many conference talks.
  - Invited talks at U. of Dundee, U. of St Andrews and U. of Toronto.
  - Three Minute Thesis.
  - Best Poster, SICSAConf 2015; runner up, CP 2014.

Modelling and Optimisation with Graphs

Ciaran McCreesh

University of Glasgow





http://www.dcs.gla.ac.uk/~ciaran

c.mccreesh.1@research.gla.ac.uk