



ISMAGS: Speeding Up Subgraph Enumeration using Dynamic Node Ordering and Symmetry Analysis

Maarten Houbraken, Pieter Audenaert, Mario Pickavet

ISMAGS

What is it?

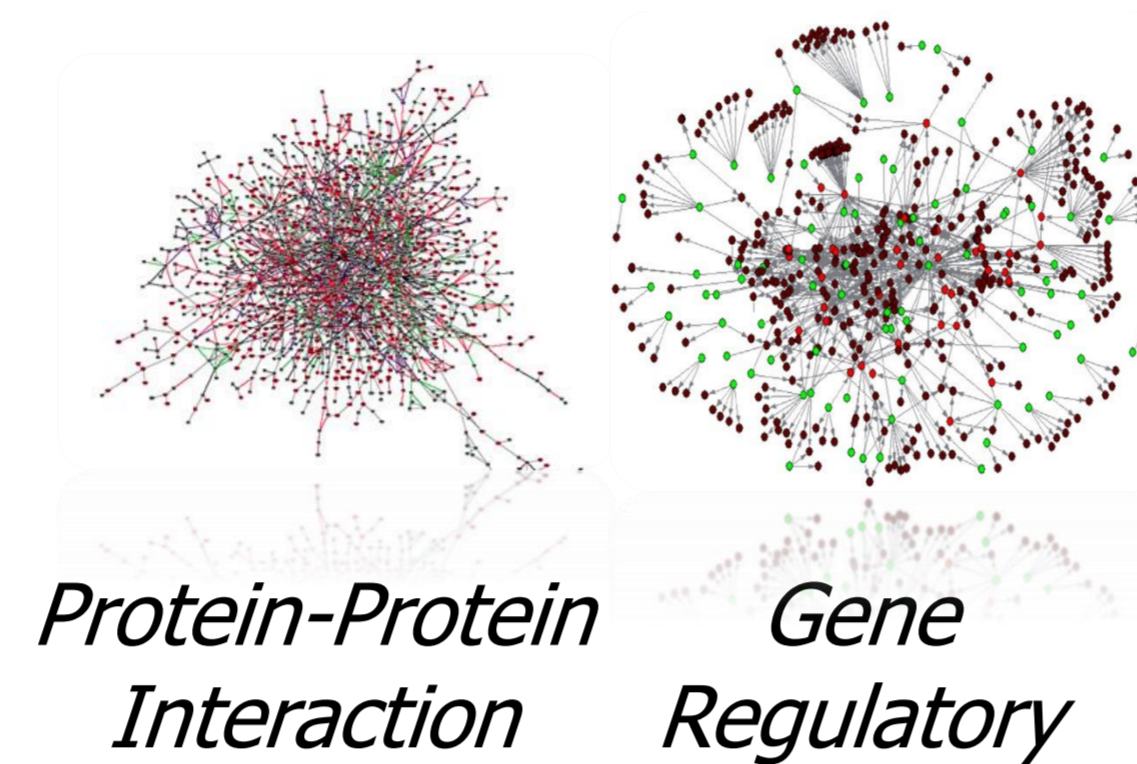
The **I**ndex-based **S**ubgraph **M**atching **A**lgorithm with **G**eneral **S**ymmetries

Subgraph enumeration algorithm

- Finding all instances of a subgraph in a larger graph
- Motif detection
- Graphlet analysis
- ...

Where can we use it?

Biological networks



General networks



How does it work?

ISMAGS

Exhaustive recursive search
Map each node of graph to node of subgraph & check for possible instance

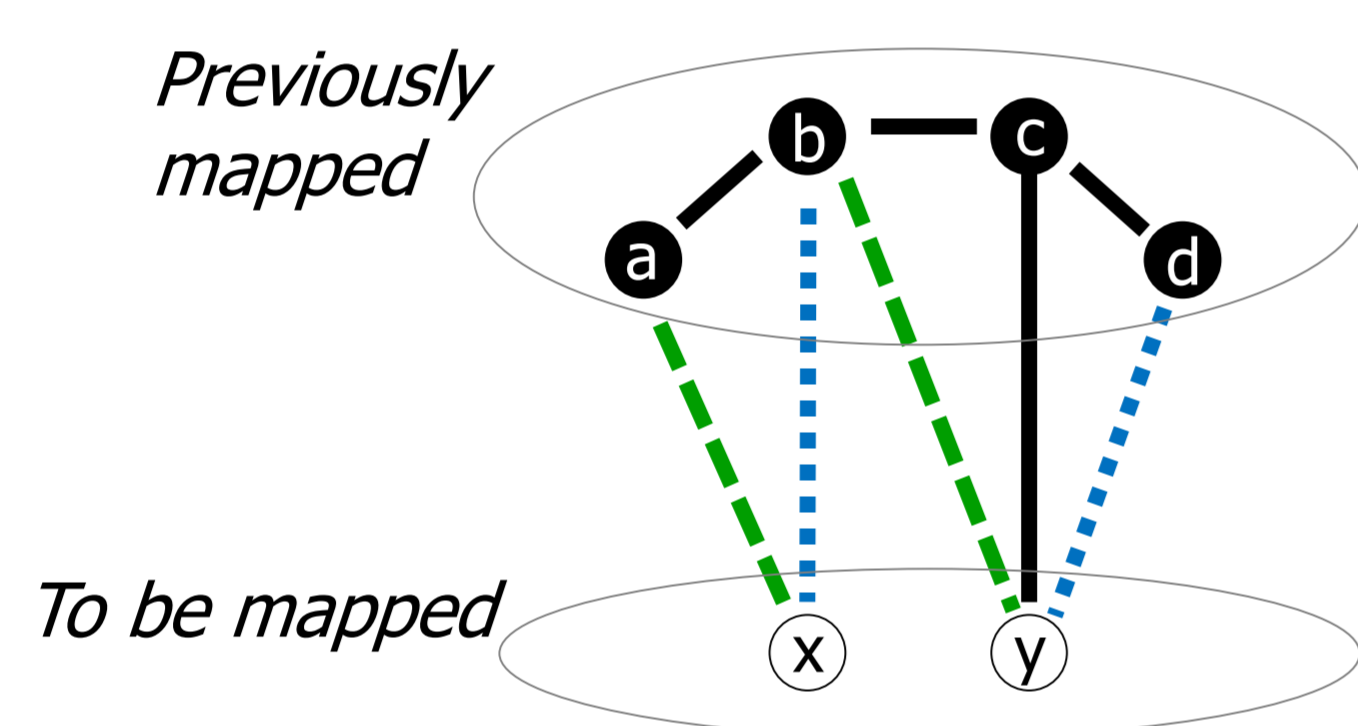
Pruning search space

Dynamic Node Ordering

Symmetry Analysis

Dynamic Node Ordering

Optimised search space traversal during exhaustive search by determining best order of node mapping



- Find subgraph instances by creating partial mappings
- Recursively expand partial mapping by selecting most constrained node

Candidates for $x =$

\cap {green neighbours of a }
{blue neighbours of b }

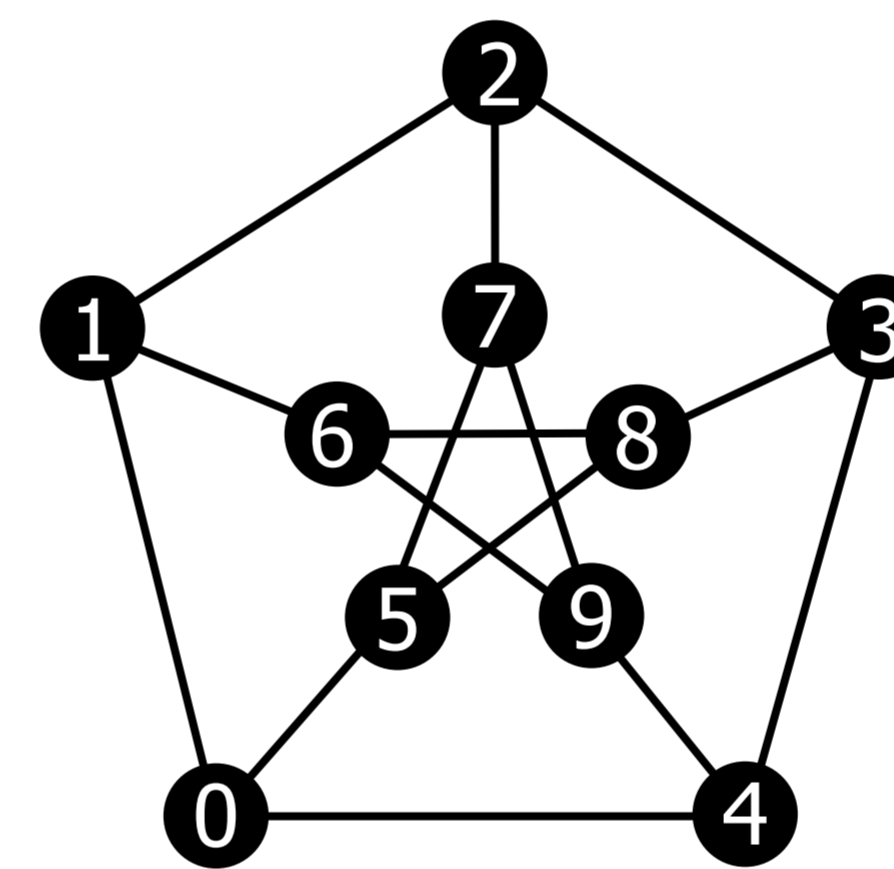
Candidates for $y =$

\cap {green neighbours of b }
{black neighbours of c }
{blue neighbours of d }

- Limiting branching factor in search space

Symmetry Analysis

Customised symmetry analysis of the subgraph to avoid examining symmetric instances



- 1) Determine generating permutations P
- 2) Extract stabiliser chains
- 3) Generate node index constraints C from coset representatives

$$P = \begin{Bmatrix} (3\ 7)(4\ 5)(8\ 9) \\ (2\ 6)(3\ 8)(4\ 5)(7\ 9) \\ (1\ 4)(2\ 3)(6\ 9)(7\ 8) \\ (0\ 3\ 1\ 4\ 2)(5\ 8\ 6\ 9\ 7) \end{Bmatrix}$$

$$C = \begin{Bmatrix} ID_0 < ID_1 & ID_0 < ID_6 & ID_1 < ID_4 \\ ID_0 < ID_2 & ID_0 < ID_7 & ID_1 < ID_5 \\ ID_0 < ID_3 & ID_0 < ID_8 & ID_2 < ID_6 \\ ID_0 < ID_4 & ID_0 < ID_9 & ID_3 < ID_7 \\ ID_0 < ID_5 \end{Bmatrix}$$

- Pruning search space by constraints

Results

Comparing ISMAGS to reference algorithms in a biological network (1.2K nodes) and a social network (7K nodes)

| Network & subgraph | Time | Speedup factor per algorithm | | | |
|---------------------|-----------|------------------------------|--------|----------|--------|
| | | ISMAGS | VF2 | GK | G-Trie |
| PPI network | | | | | |
| 3-clique | 8.81 ms | 132.79 | 58.04 | 0.17 | 4.48 |
| 9-clique | 92.21 ms | 14509 | 41.36 | 7.28 | 4.68 |
| Petersen | 733.25 ms | 9348 | 73.11 | ∞ | 451.25 |
| Wikipedia elections | | | | | |
| 3-clique | 410.19 ms | 456.35 | 68.12 | 0.22 | 4.37 |
| Tetrahedron | 320.72 ms | 47.58 | 331.66 | 1.38 | 3.20 |
| G4 | 448.54 ms | 18.21 | 287.23 | 2.24 | 1.29 |

Improvement over VF2 & GK ?

- Optimised implementation of partial mapping generation and expansion

Improvement over G-Trie ?

- Dynamic node order reduces backtracking
- Better scaling: larger gains for larger subgraphs

Improvement over ISMA ?

- Symmetry constraints avoid exploring symmetric parts of search space

➤➤➤ ISMAGS

Speeding up subgraph enumeration
Enabling larger graph analysis
Opening future research possibilities

M. Houbraken, S. Demeyer, T. Michoel, P. Audenaert, D. Colle, M. Pickavet (2014) The Index-Based Subgraph Matching Algorithm with General Symmetries (ISMAGS): Exploiting Symmetry for Faster Subgraph Enumeration, PLOS ONE (accepted).