Maximum Common Subgraph

Algorithms and Algorithm Portfolios

Who? Paulius Dilkas [1] Ciaran McCreesh [2]

From? [1]University of Edinburgh, UK

[2]University of Glasgow, UK

When? LJMS 8

Outline

The Problem

Algorithms

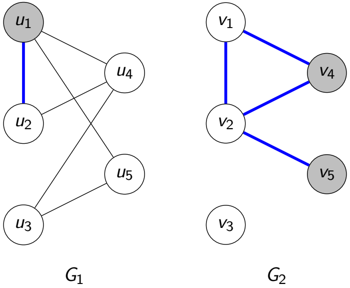
Algorithm Selection

Results & Observations

Maximum Common Subgraph

Definition

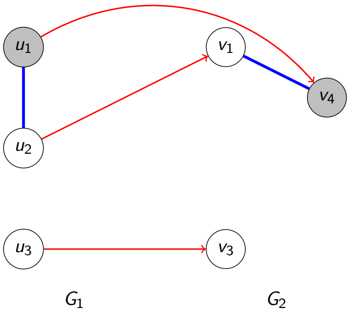
A *maximum common (induced) subgraph* between graphs G_1 and G_2 is a graph $G_3 = (V_3, E_3)$ such that G_3 is isomorphic to induced subgraphs of both G_1 and G_2 with $|V_3|$ maximised.



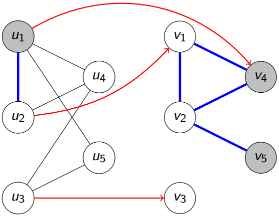
Maximum Common Subgraph

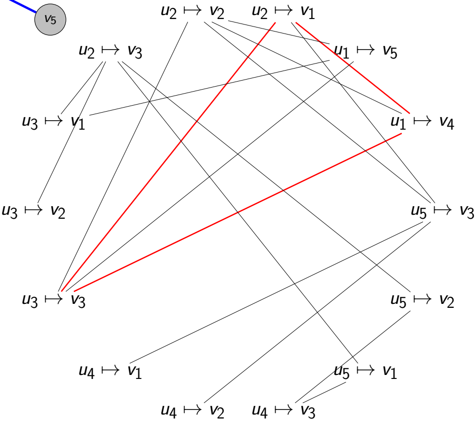
Definition

A *maximum common (induced) subgraph* between graphs G_1 and G_2 is a graph $G_3 = (V_3, E_3)$ such that G_3 is isomorphic to induced subgraphs of both G_1 and G_2 with $|V_3|$ maximised.

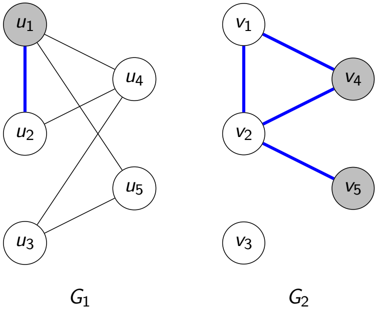


Clique Encoding





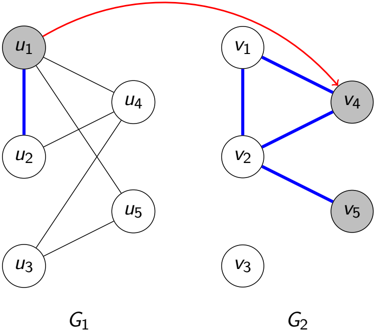
McSplit: a Branch and Bound Algorithm



Upper bound: 4
Partial solution:
$\{\}$

Label	G_1	G_2
0	u_2, u_3, u_4, u_5	v_1, v_2, v_3
1	u_1	v_4, v_5

McSplit: a Branch and Bound Algorithm

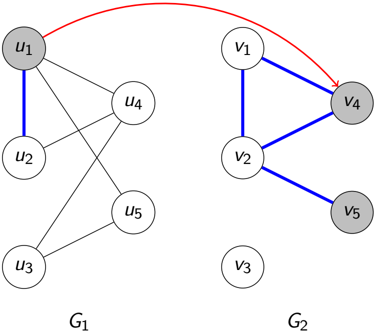


Upper bound: 4
Partial solution:
$\{\}$

Label	G_1	G_2
0	u_2, u_3, u_4, u_5	v_1, v_2, v_3
1	u_1	v_4, v_5

Decision: $u_1 \mapsto v_4$

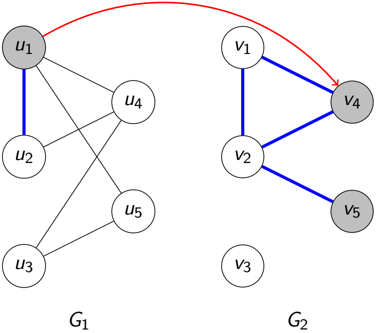
McSplit: a Branch and Bound Algorithm



Upper bound: 4
Partial solution:
$\{u_1 \mapsto v_4\}$

Label	G_1	G_2
00	u_3	v_3
01	u_4, u_5	$\emptyset$
02	u_2	v_1, v_2
10	$\emptyset$	v_5

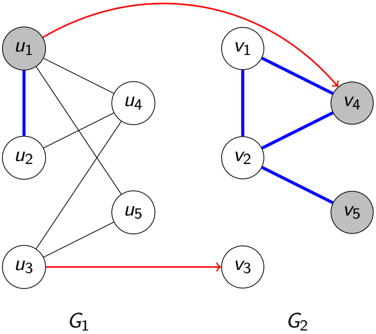
McSplit: a Branch and Bound Algorithm



Upper bound: $1 + 2$
Partial solution:
$\{u_1 \mapsto v_4\}$

Label	G_1	G_2
00	u_3	v_3
01	u_2	v_1, v_2

McSplit: a Branch and Bound Algorithm

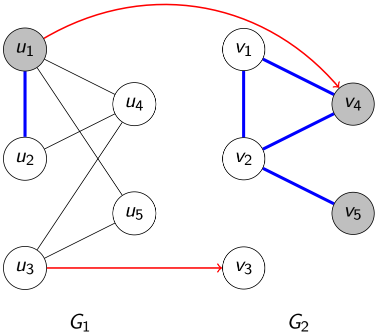


Upper bound: $1 + 2$
Partial solution:
$\{u_1 \mapsto v_4\}$

Label	G_1	G_2
00	u_3	v_3
01	u_2	v_1, v_2

Decision: $u_3 \mapsto v_3$

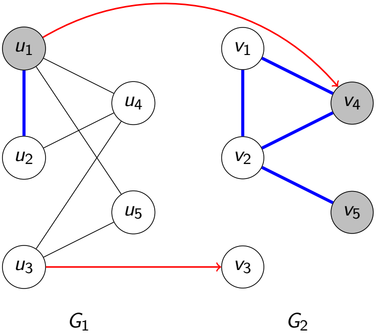
McSplit: a Branch and Bound Algorithm



Upper bound: $1 + 2$
Partial solution:
$\{u_1 \mapsto v_4, u_3 \mapsto v_3\}$

Label	G_1	G_2
010	u_2	v_1, v_2
011	u_4, u_5	$\emptyset$

McSplit: a Branch and Bound Algorithm



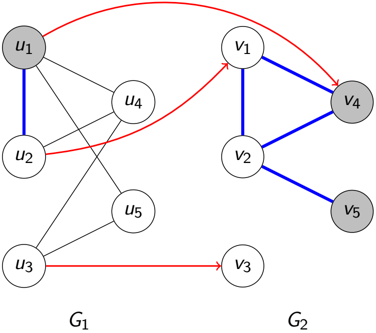
Upper bound: $2 + 1$

Partial solution:

$\{u_1 \mapsto v_4, u_3 \mapsto v_3\}$

Label	G_1	G_2
010	u_2	v_1, v_2

McSplit: a Branch and Bound Algorithm



Upper bound: $2 + 1$
Partial solution:
$\{u_1 \mapsto v_4, u_3 \mapsto v_3\}$

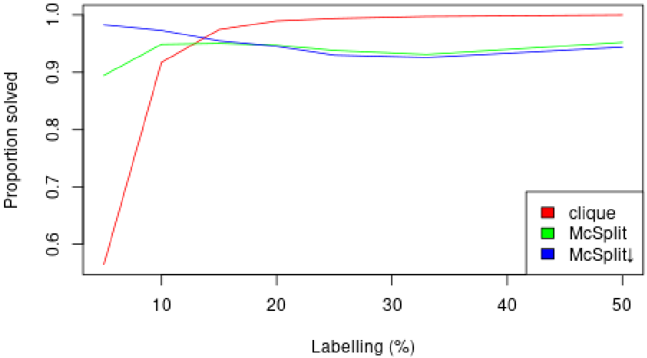
Label	G_1	G_2
010	u_2	v_1, v_2

Decision: $u_2 \mapsto v_1$

Found a solution!

Backtrack to confirm optimality

Which Is Better?



(Per-Instance) Algorithm Selection

(G_1, G_2)

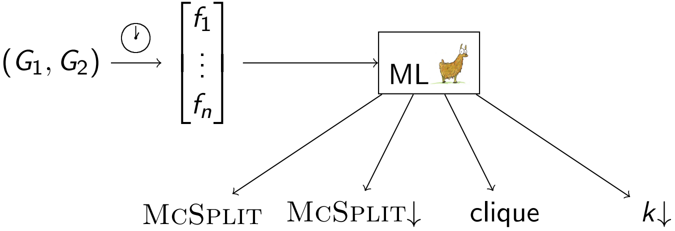
(Per-Instance) Algorithm Selection

$$(G_1, G_2) \longrightarrow \begin{bmatrix} f_1 \\ \vdots \\ f_n \end{bmatrix}$$

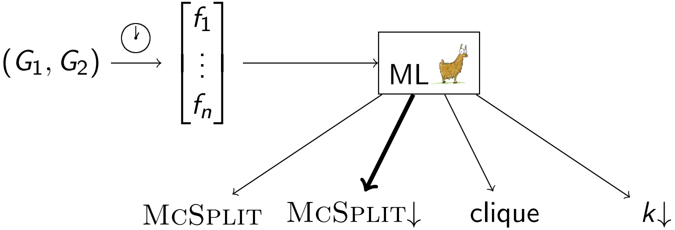
(Per-Instance) Algorithm Selection

$$(G_1, G_2) \longrightarrow \begin{bmatrix} f_1 \\ \vdots \\ f_n \end{bmatrix} \longrightarrow \boxed{\text{ML}}$$

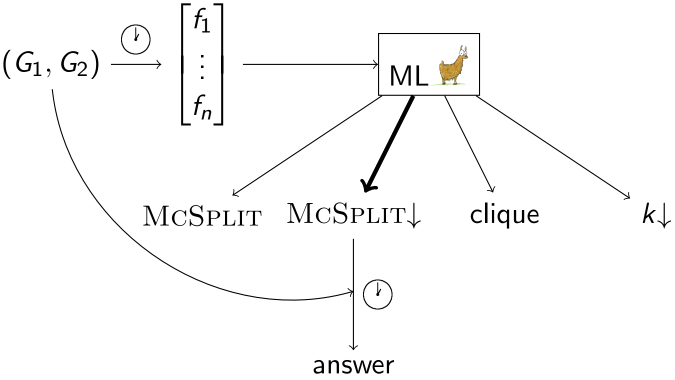
(Per-Instance) Algorithm Selection



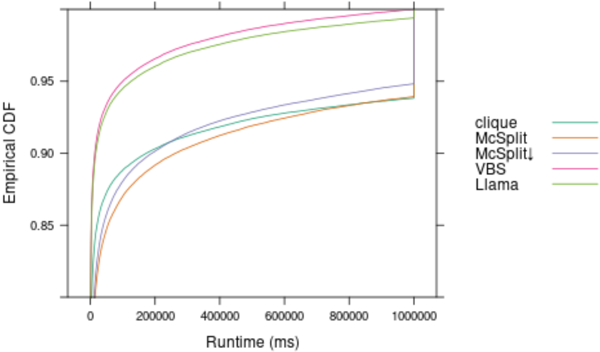
(Per-Instance) Algorithm Selection



(Per-Instance) Algorithm Selection



Overall Performance



Observations

- Most important features:
  - labelling percentage
  - standard deviation of degrees (for both graphs)
- Looking at a single feature is not enough

Observations

- Most important features:
  - labelling percentage
  - standard deviation of degrees (for both graphs)
- Looking at a single feature is not enough

Thank You!