

Counting maximal independent sets in subcubic graphs.

Konstanty Junosza-Szaniawski, Michał Tuczyński

Warsaw University of Technology

SOFSEM 2012

What is known:

Complexity

	Independent sets	Maximal independent sets
$\Delta = 2$	polynomial	polynomial
$\Delta = 3$		
$\Delta = 4$		
$\Delta = 5$		

What is known:

Complexity

	Independent sets	Maximal independent sets
$\Delta = 2$	polynomial	polynomial
$\Delta = 3$		
$\Delta = 4$	#P-complete Vadhan 1997	
$\Delta = 5$	#P-complete Vadhan 1997	

What is known:

Complexity

	Independent sets	Maximal independent sets
$\Delta = 2$	polynomial	polynomial
$\Delta = 3$		
$\Delta = 4$	#P-complete Vadhan 1997	
$\Delta = 5$	#P-complete Vadhan 1997	#P-complete Vadhan 1997

What is known:

Complexity

	Independent sets	Maximal independent sets
$\Delta = 2$	polynomial	polynomial
$\Delta = 3$	#P-complete Greenhill 2000	
$\Delta = 4$	#P-complete Vadhan 1997	
$\Delta = 5$	#P-complete Vadhan 1997	#P-complete Vadhan 1997

What is known:

Complexity

	Independent sets	Maximal independent sets
$\Delta = 2$	polynomial	polynomial
$\Delta = 3$	#P-complete Greenhill 2000	
$\Delta = 4$	#P-complete Vadhan 1997	#P-complete Greenhill 2000
$\Delta = 5$	#P-complete Vadhan 1997	#P-complete Vadhan 1997

What is known:

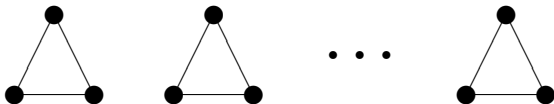
Complexity

	Independent sets	Maximal independent sets
$\Delta = 2$	polynomial	polynomial
$\Delta = 3$	#P-complete Greenhill 2000	???
$\Delta = 4$	#P-complete Vadhan 1997	#P-complete Greenhill 2000
$\Delta = 5$	#P-complete Vadhan 1997	#P-complete Vadhan 1997

What is known:

Moon, Moser 1965

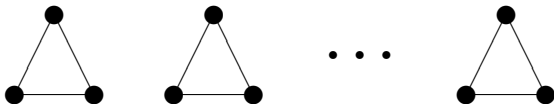
Number of MIS is at most $3^{n/3} = 1.44..^n$



What is known:

Moon, Moser 1965

Number of MIS is at most $3^{n/3} = 1.44..^n$



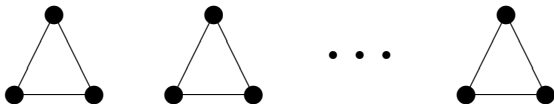
Johnson, Yannakakis 1988

MIS can be generated with polynomial delay

What is known:

Moon, Moser 1965

Number of MIS is at most $3^{n/3} = 1.44..^n$



Johnson, Yannakakis 1988

MIS can be generated with polynomial delay

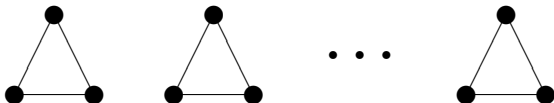
Gaspers, Kratsch, Liedloff 2008, 2012

Maximal independent sets can be counted in time $O^*(1.3642^n)$

What is known:

Moon, Moser 1965

Number of MIS is at most $3^{n/3} = 1.44..^n$



Johnson, Yannakakis 1988

MIS can be generated with polynomial delay

Gaspers, Kratsch, Liedloff 2008, 2012

Maximal independent sets can be counted in time $O^*(1.3642^n)$

Maximal independent sets in a subcubic graph
can be counted in time $O^*(1.3532^n)$ in polynomial space.

Björklund, Husfeldt 2006

Graph can be colored in time $O^*((1+c)^n)$ and polynomial space, it there is an algorithm counting independent sets in time $O^*(c^n)$ and polynomial space.

What is known

Björklund, Husfeldt 2006

Graph can be colored in time $O^*((1+c)^n)$ and polynomial space, it there is an algorithm counting independent sets in time $O^*(c^n)$ and polynomial space.

Counting independent sets - not necessary maximal

	$\Delta(G) = 3$	arbitrary $\Delta(G)$
Dahllöf, Jonsson, Wahlström 2002	$O^*(1.18..^n)$	$O^*(1.25..^n)$

What is known

Björklund, Husfeldt 2006

Graph can be colored in time $O^*((1+c)^n)$ and polynomial space, it there is an algorithm counting independent sets in time $O^*(c^n)$ and polynomial space.

Counting independent sets - not necessary maximal

	$\Delta(G) = 3$	arbitrary $\Delta(G)$
Dahllöf, Jonsson, Wahlström 2002	$O^*(1.18..^n)$	$O^*(1.25..^n)$
Fürer, Kasiviswanathan 2005	$O^*(1.15..^n)$	$O^*(1.24..^n)$

What is known

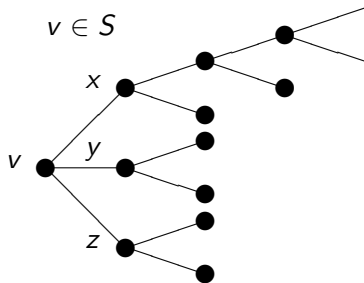
Björklund, Husfeldt 2006

Graph can be colored in time $O^*((1+c)^n)$ and polynomial space, it there is an algorithm counting independent sets in time $O^*(c^n)$ and polynomial space.

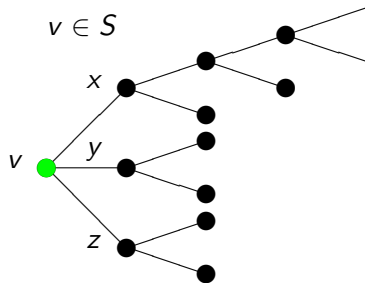
Counting independent sets - not necessary maximal

	$\Delta(G) = 3$	arbitrary $\Delta(G)$
Dahllöf, Jonsson, Wahlström 2002	$O^*(1.18..^n)$	$O^*(1.25..^n)$
Fürer, Kasiviswanathan 2005	$O^*(1.15..^n)$	$O^*(1.24..^n)$
Wahlström 2008	$O^*(1.15..^n)$	$O^*(1.23..^n)$

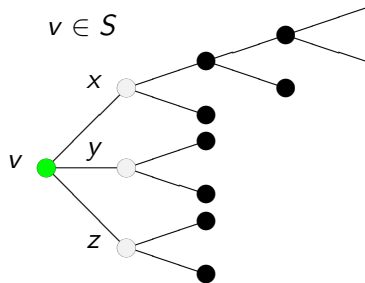
Branching



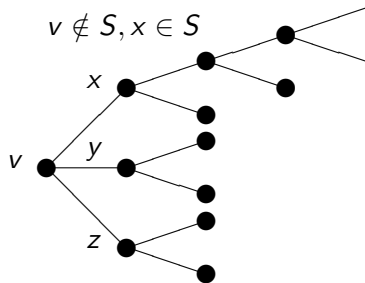
Branching



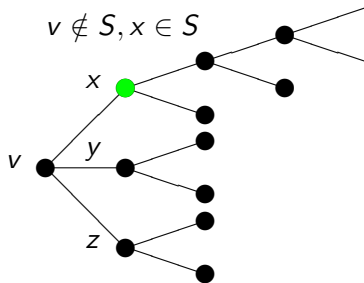
Branching



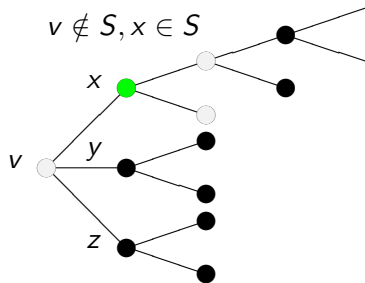
Branching



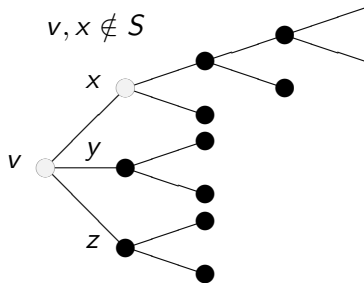
Branching



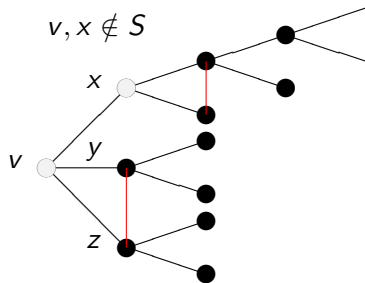
Branching



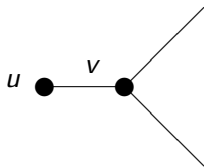
Branching



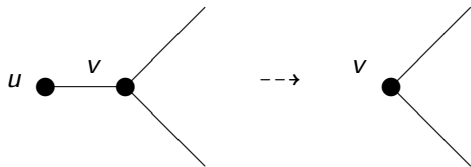
Branching



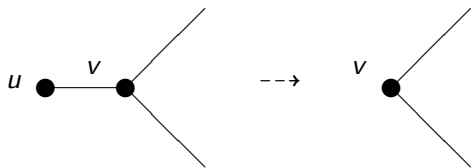
Reduction



Reduction



Reduction



$$c_1(v) := c_1(v) \cdot c_0(u)$$

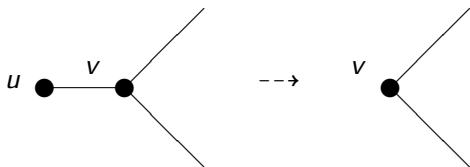
$$c_0(v) := c_0(v) \cdot (c_1(u) + c_0(u))$$

$$c_{\bar{0}}(v) := c_1(u)$$

Reduction

$$\sum_S \prod_{w:w \in S} c_1(w) \prod_{\substack{w:w \notin S, \\ \bar{N}(w) \cap S \neq \emptyset}} c_0(w) \prod_{w:\bar{N}(w) \cap S = \emptyset} c_{\bar{0}}(w)$$

For $c_1 = 1$, $c_0 = 1$, $c_{\bar{0}} = 0$ the sum is equal to the number of MIS.



$$c_1(v) := c_1(v) \cdot c_0(u)$$

$$c_0(v) := c_0(v) \cdot (c_1(u) + c_0(u))$$

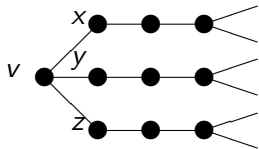
$$c_{\bar{0}}(v) := c_1(u)$$

The algorithm

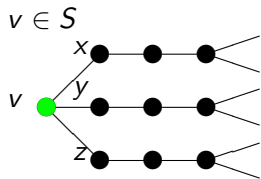
MISCount($G, c_1, c_0, c_{\bar{0}}$)

1. Reduction($G, c_1, c_0, c_{\bar{0}}$)
2. Choose a vertex v
3. Branch on the vertex v

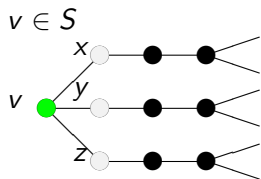
Complexity



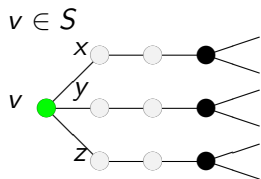
Complexity



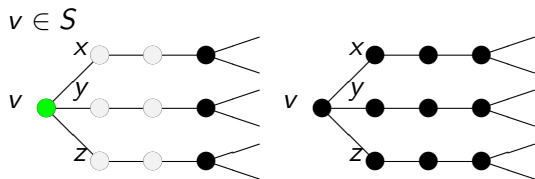
Complexity



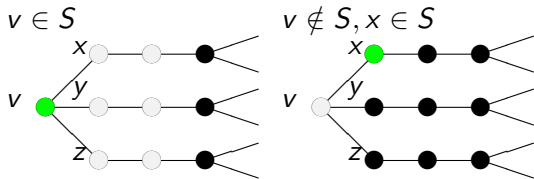
Complexity



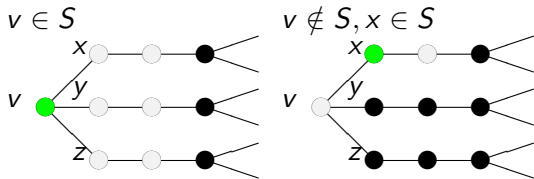
Complexity



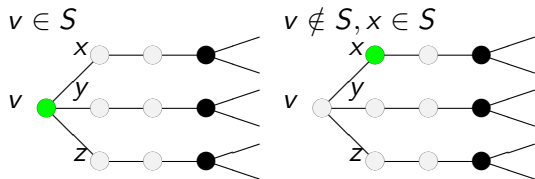
Complexity



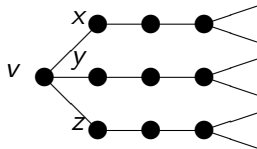
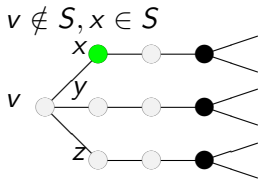
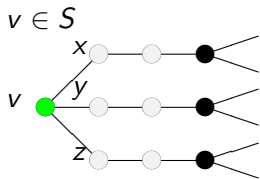
Complexity



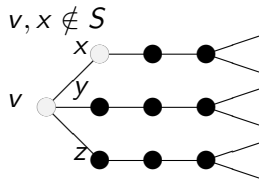
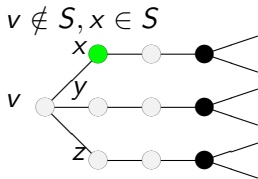
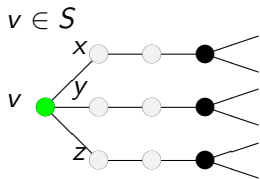
Complexity



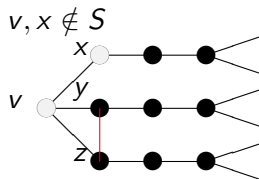
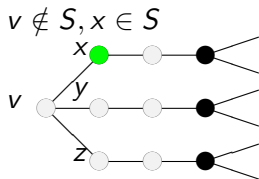
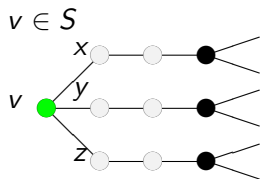
Complexity



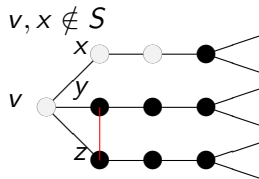
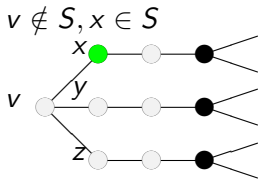
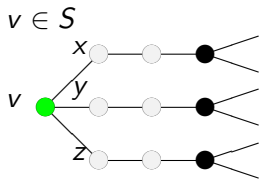
Complexity



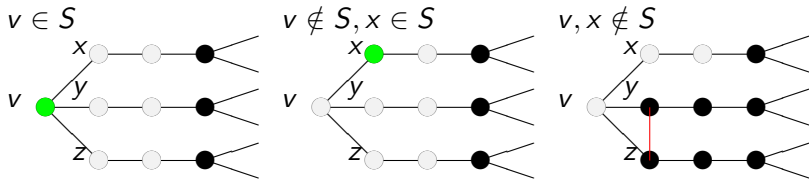
Complexity



Complexity

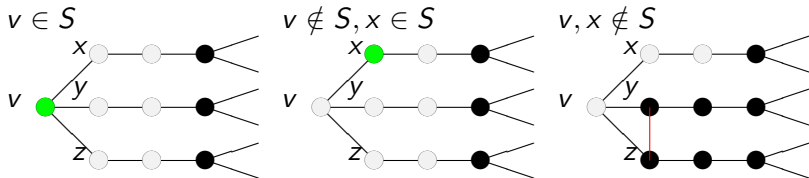


Complexity



$$T(n_3) = T(n_3 - 4) + T(n_3 - 4) + T(n_3 - 2)$$

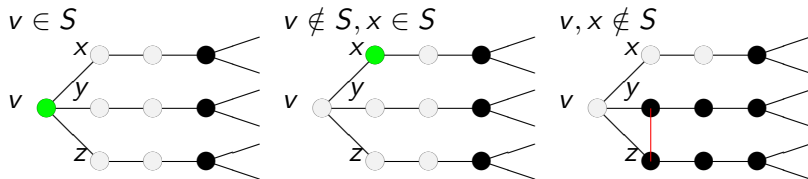
Complexity



$$T(n_3) = T(n_3 - 4) + T(n_3 - 4) + T(n_3 - 2)$$

$$T(n_3) = O^*(1.41..^{n_3})$$

Complexity

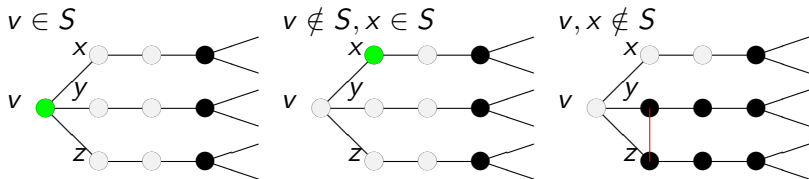


$$T(n_3) = T(n_3 - 4) + T(n_3 - 4) + T(n_3 - 2)$$

$$T(n_3) = O^*(1.41..^{n_3})$$

$$T(n) = O^*(1.41..^{\frac{1}{4} \cdot n}) =$$

Complexity

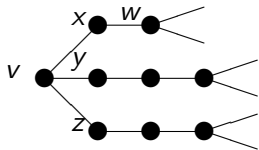


$$T(n_3) = T(n_3 - 4) + T(n_3 - 4) + T(n_3 - 2)$$

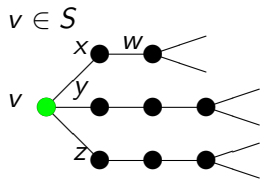
$$T(n_3) = O^*(1.41..^{n_3})$$

$$T(n) = O^*(1.41..^{\frac{1}{4} \cdot n}) = O^*(1.09..^n)$$

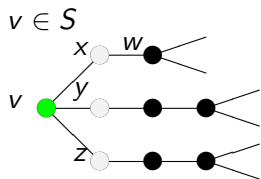
Complexity



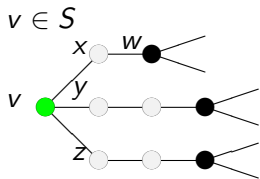
Complexity



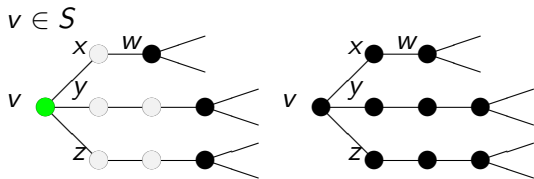
Complexity



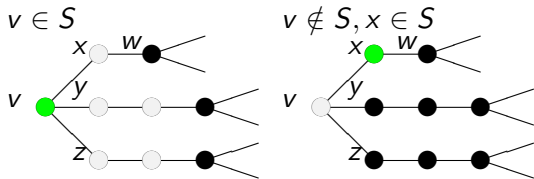
Complexity



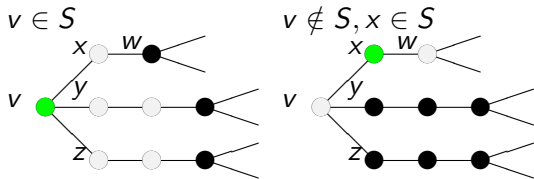
Complexity



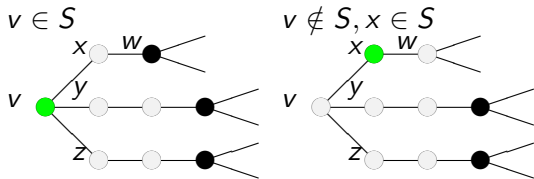
Complexity



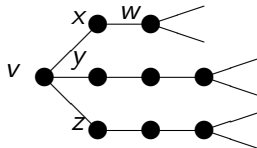
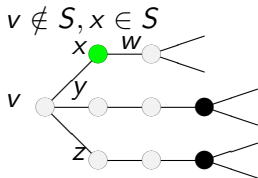
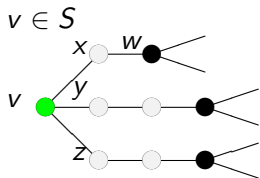
Complexity



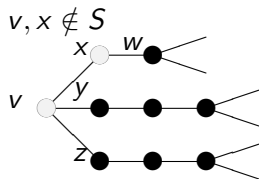
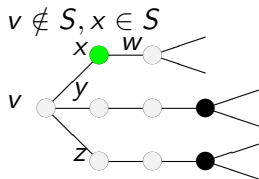
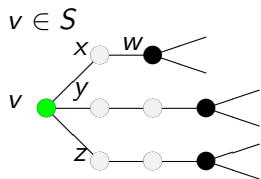
Complexity



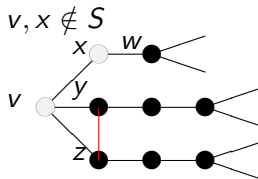
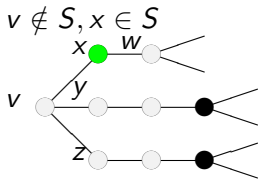
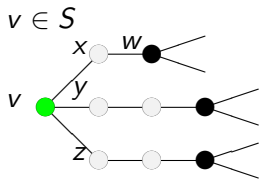
Complexity



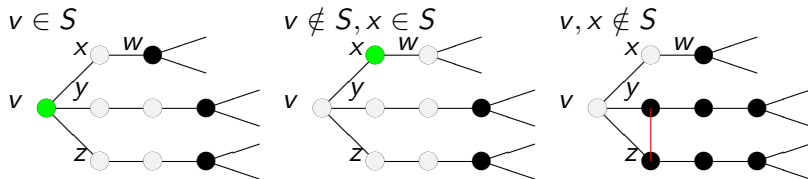
Complexity



Complexity

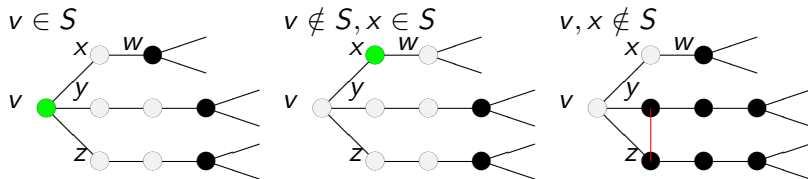


Complexity



$$T(n_3) = T(n_3 - 4) + T(n_3 - 6) + T(n_3 - 2)$$

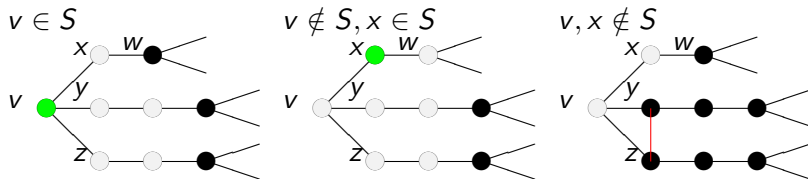
Complexity



$$T(n_3) = T(n_3 - 4) + T(n_3 - 6) + T(n_3 - 2)$$

$$T(n_3) = O^*(1.36..^{n_3})$$

Complexity

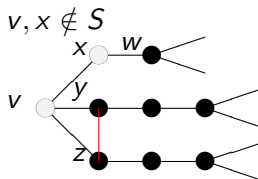
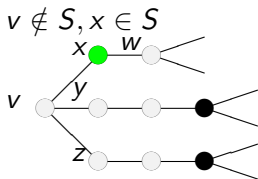
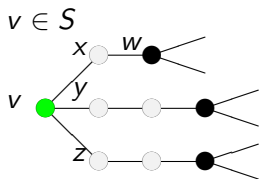


$$T(n_3) = T(n_3 - 4) + T(n_3 - 6) + T(n_3 - 2)$$

$$T(n_3) = O^*(1.36..n_3)$$

$$T(n) = O^*(1.36..^{\frac{1}{3.5}}n) =$$

Complexity

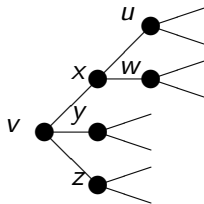


$$T(n_3) = T(n_3 - 4) + T(n_3 - 6) + T(n_3 - 2)$$

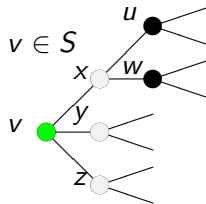
$$T(n_3) = O^*(1.36..n_3)$$

$$T(n) = O^*(1.36..^{\frac{1}{3.5}}n) = O^*(1.09..n)$$

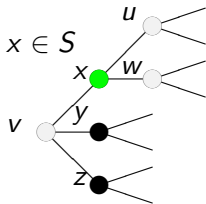
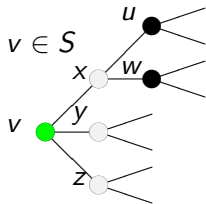
Complexity



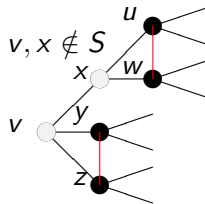
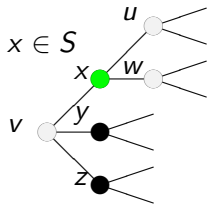
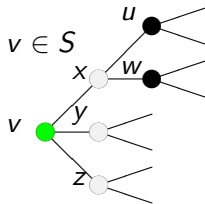
Complexity



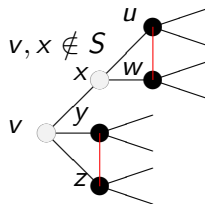
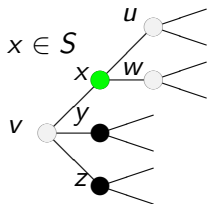
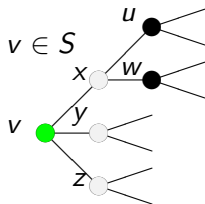
Complexity



Complexity

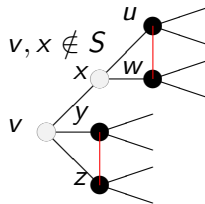
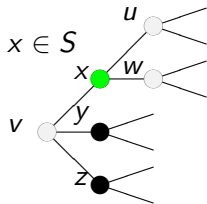
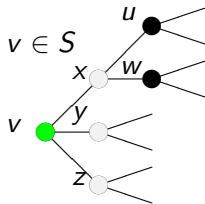


Complexity



$$T(n_3) = T(n_3 - 10) + T(n_3 - 10) + T(n_3 - 2)$$

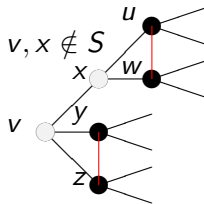
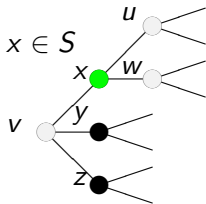
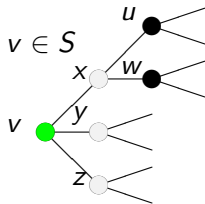
Complexity



$$T(n_3) = T(n_3 - 10) + T(n_3 - 10) + T(n_3 - 2)$$

$$T(n_3) = O^*(1.21..^{n_3})$$

Complexity



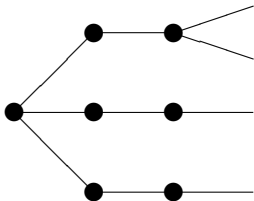
$$T(n_3) = T(n_3 - 10) + T(n_3 - 10) + T(n_3 - 2)$$

$$T(n_3) = O^*(1.21..^{n_3})$$

$$T(n) = O^*(1.21..^n)$$

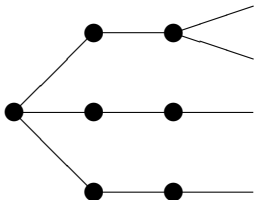
Lemma on density

$$\frac{2m}{n} > 2.25$$

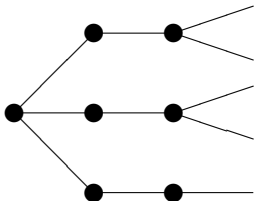


Lemma on density

$$\frac{2m}{n} > 2.25$$

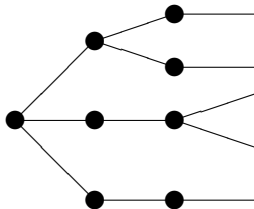
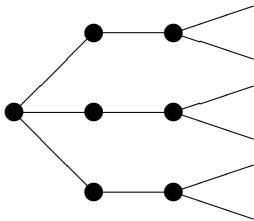


$$\frac{2m}{n} > \frac{16}{7}$$



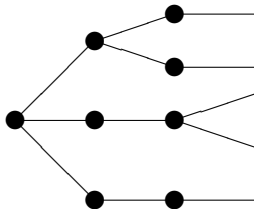
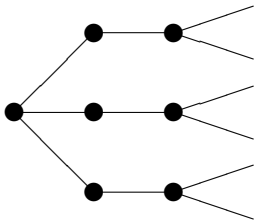
Lemma on density

$$\frac{2m}{n} > \frac{7}{3}$$

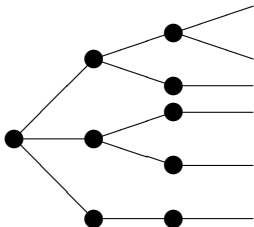


Lemma on density

$$\frac{2m}{n} > \frac{7}{3}$$

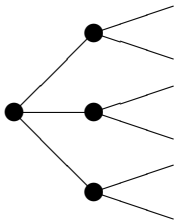


$$\frac{2m}{n} > \frac{28}{11}$$



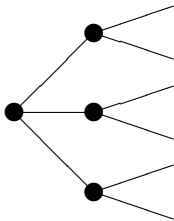
Lemma on density

$$\frac{2m}{n} > \frac{8}{3}$$

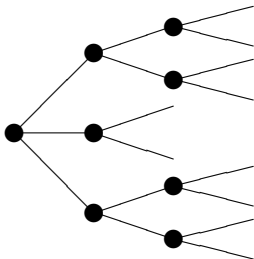


Lemma on density

$$\frac{2m}{n} > \frac{8}{3}$$



$$\frac{2m}{n} > 2.8$$



Complexity - Measure and Conquer

Let μ be a measure of a graph, let G_i for $i = 1..t$ be graphs obtained in the i -th branch of recursive call of an algorithm \mathcal{A} , let τ_0 be the largest root of

$$\sum_{i=1}^k x^{-(\mu(G) - \mu(G_i))} = 1$$

then the algorithm \mathcal{A} runs in time $O^*(\tau_0^{\mu(G)})$.

Complexity - the measure

$$\mu(n_2, n_3) = \begin{cases} \mu_0(n_2, n_3) & \text{if } n = 0 \text{ lub } \frac{2m}{n} = 2 \\ \mu_1(n_2, n_3) & \text{if } 2 < \frac{2m}{n} \leq 2\frac{1}{4} \\ \mu_2(n_2, n_3) & \text{if } 2\frac{1}{4} < \frac{2m}{n} \leq 2\frac{2}{7} \\ \dots & \\ \mu_7(n_2, n_3) & \text{if } 2\frac{14}{17} < \frac{2m}{n} \leq 3. \end{cases}$$

$$\mu_i(n_2, n_3) = w_{i,2}n_2 + w_{i,3}n_3$$

i	p_{i-1}	p_i	$w_{i,2}$	$w_{i,3}$
0		2	0	0
1	2	$2\frac{1}{4}$	0	0.5
2	$2\frac{1}{4}$	$2\frac{2}{7}$	0.018460	0.444619
3	$2\frac{2}{7}$	$2\frac{1}{3}$	0.037065	0.3981058
4	$2\frac{1}{3}$	$2\frac{6}{11}$	0.0520754	0.368086
5	$2\frac{6}{11}$	$2\frac{4}{5}$	0.082954	0.342353
6	$2\frac{4}{5}$	$2\frac{14}{17}$	0.140599	0.327942
7	$2\frac{14}{17}$	3	0.154465	0.324971

Main result

Maximal independent sets in subcubic graph can be counted in time $O^*(1.2526..^n)$

Main result

Maximal independent sets in subcubic graph can be counted in time $O^*(1.2526..^n)$

Future work

- Count minimal models of 2-SAT formula, with variables appearing no more than 3 times.

Main result

Maximal independent sets in subcubic graph can be counted in time $O^*(1.2526..^n)$

Future work

- Count minimal models of 2-SAT formula, with variables appearing no more than 3 times.
- Count maximal independent set in graphs with maximum degree 4.

Main result

Maximal independent sets in subcubic graph can be counted in time $O^*(1.2526..^n)$

Future work

- Count minimal models of 2-SAT formula, with variables appearing no more than 3 times.
- Count maximal independent set in graphs with maximum degree 4.

THANK YOU