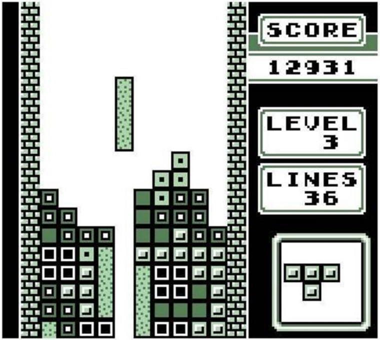
How to count leaves in the trees from Tetris?

Élise Vandomme
Postdoc at the LaCIM (UQAM)

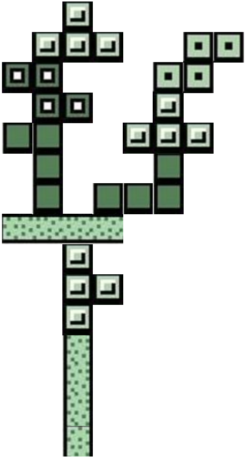
Colloque panquébécois des étudiants de l'ISM
Trois-Rivières – May 2017


TETRIS™
▸1PLAYER 2PLAYER
©1989 Nintendo

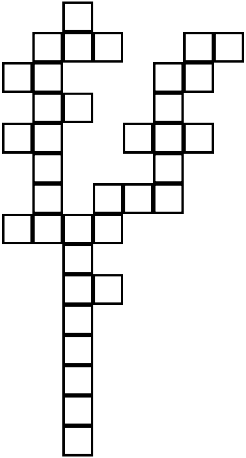

SCORE
12931
LEVEL
3
LINES
36

Trees in Tetris

Trees in Tetris

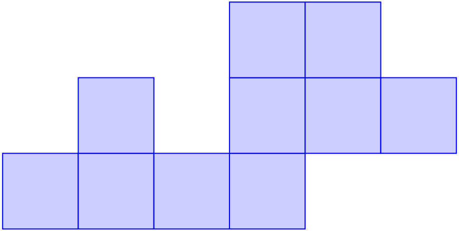


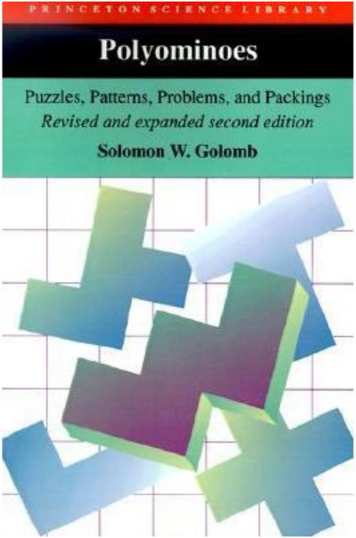
Trees in Tetris



Polyomino

A polyomino is a connected union of unit squares, called cells.



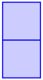


Solomon W. Golomb (1932 - 2016)

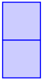
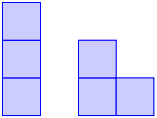
Combinatorial Problems

- Monomino: 

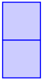
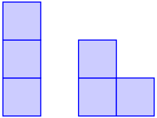
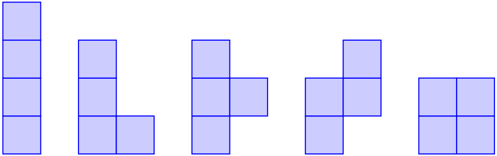
Combinatorial Problems

- Monomino: 
- Domino: 

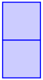
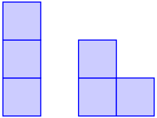
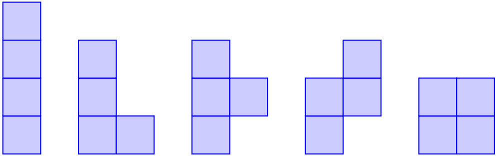
Combinatorial Problems

- Monomino: 
- Domino: 
- Tromino: 

Combinatorial Problems

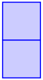
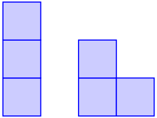
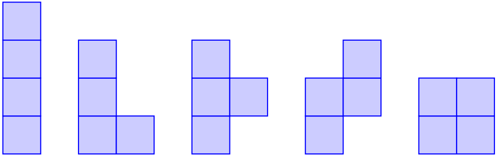
- Monomino: 
- Domino: 
- Tromino: 
- Tetromino: 

Combinatorial Problems

- Monomino: 
- Domino: 
- Tromino: 
- Tetromino: 

How many polyominoes with n cells?

Combinatorial Problems

- Monomino: 
- Domino: 
- Tromino: 
- Tetromino: 

How many polyominoes with n cells?

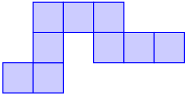
Unknown for $n \geq 57$

Combinatorial Problems

Can we pave the plane with a finite set of polyominoes?

Examples:

- Polyominoes:

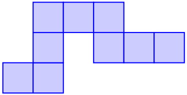


Combinatorial Problems

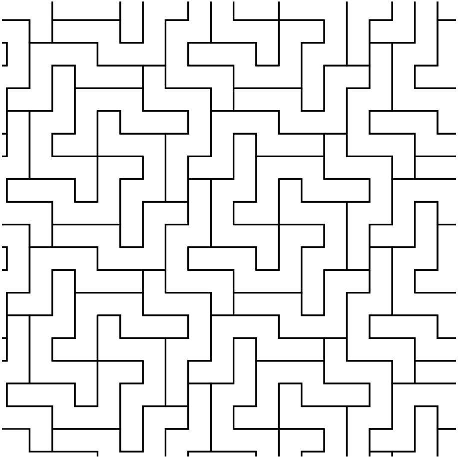
Can we pave the plane with a finite set of polyominoes?

Examples:

- Polyominoes:



Yes, we can!

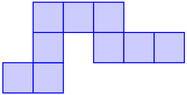


Combinatorial Problems

Can we pave the plane with a finite set of polyominoes?

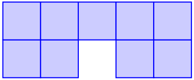
Examples:

- Polyominoes:

  Yes, we can!

- Polyominoes:

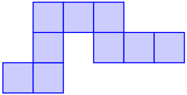
  

Combinatorial Problems

Can we pave the plane with a finite set of polyominoes?

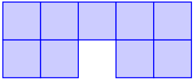
Examples:

- Polyominoes:

  Yes, we can!

- Polyominoes:

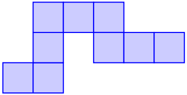
  No, we can't!

Combinatorial Problems

Can we pave the plane with a finite set of polyominoes?

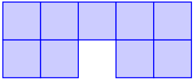
Examples:

- Polyominoes:



Yes, we can!

- Polyominoes:



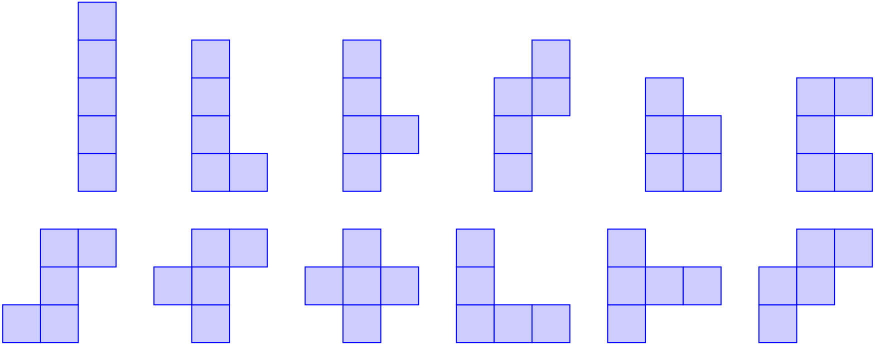
No, we can't!

In general, the problem is undecidable.

Combinatorial Problems

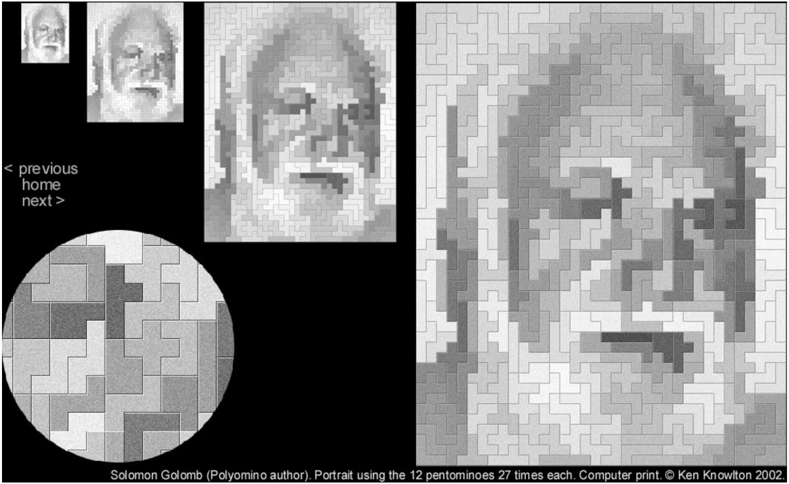
Can we pave a finite area with a finite set of polyominoes?
Example:

- Area: 45×36 rectangle
- Polyominoes: all pentominoes



Combinatorial Problems

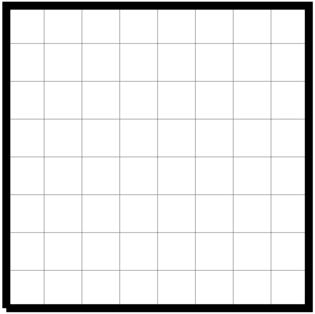
Can we pave a finite area with a finite set of polyominoes?



http://www.knowltonmosaics.com

Combinatorial Problems

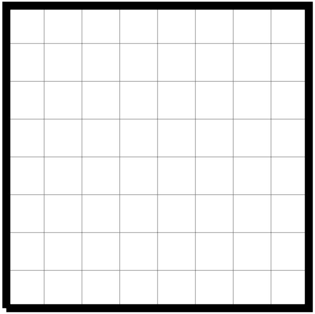
Can we pave a finite area with a finite set of polyominoes?
Example:

- Area: a checkerboard

- Polyominoes: a domino 

Combinatorial Problems

Can we pave a finite area with a finite set of polyominoes?
Example:

- Area: a checkerboard



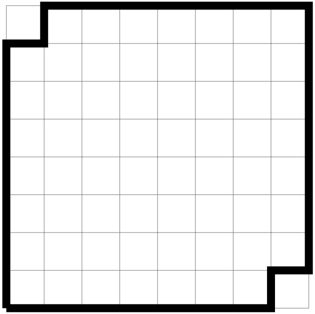
- Polyominoes: a domino 

Yes, we can!

Combinatorial Problems

Can we pave a finite area with a finite set of polyominoes?
Example:

- Area: a checkerboard without two opposite corners

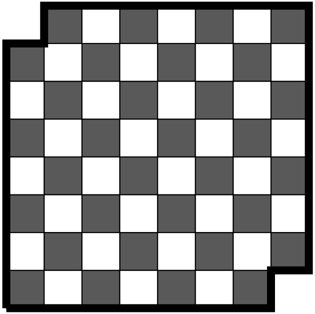


- Polyominoes: a domino 

Combinatorial Problems

Can we pave a finite area with a finite set of polyominoes?
Example:

- Area: a checkerboard without two opposite corners



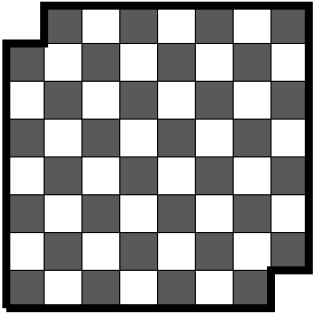
- Polyominoes: a domino 

No, we can't!

Combinatorial Problems

Can we pave a finite area with a finite set of polyominoes?
Example:

- Area: a checkerboard without two opposite corners

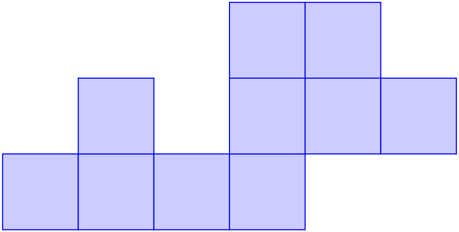


- Polyominoes: a domino 

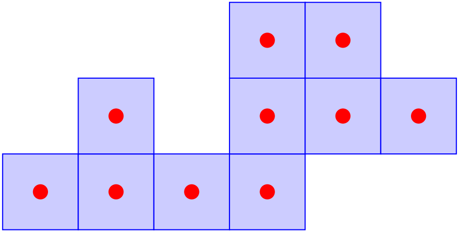
No, we can't!

In general, the problem is NP-complete.

Behind each polyomino hides a graph

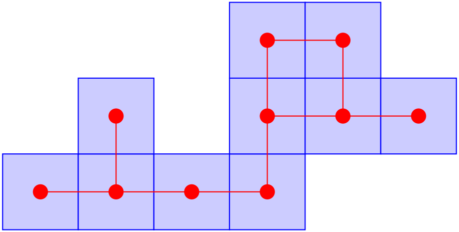


Behind each polyomino hides a graph



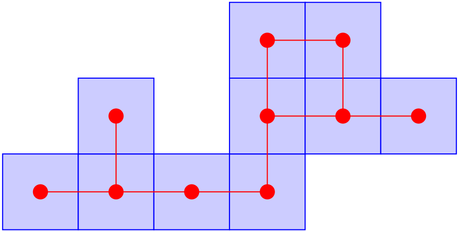
- Vertices are the cells.

Behind each polyomino hides a graph



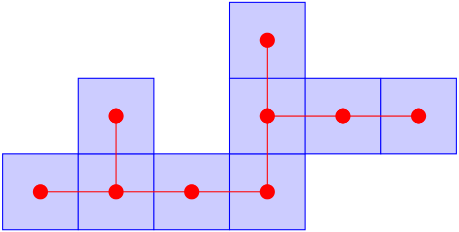
- Vertices are the cells.
- Two vertices are adjacent if their corresponding cells have a common side.

Behind each polyomino hides a graph



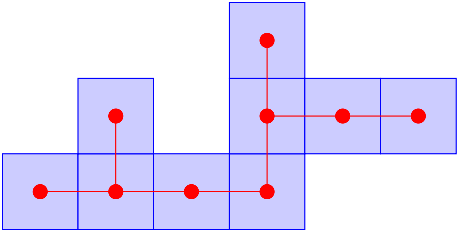
- Vertices are the cells.
- Two vertices are adjacent if their corresponding cells have a common side.
- Such graph can contain cycles.

Tree-like Polyominoes



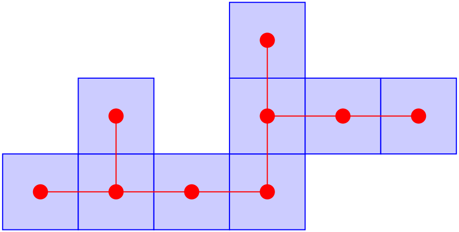
- If there are no cycles, we call it a tree-like polyomino.

Tree-like Polyominoes



- If there are no cycles, we call it a tree-like polyomino.
- This tree-like polyomino has 9 cells and 4 leaves.

Tree-like Polyominoes



- If there are no cycles, we call it a tree-like polyomino.
- This tree-like polyomino has 9 cells and 4 leaves.
- For a given n, what is the maximal number of leaves realized by a tree-like polyomino with n cells?

Maximal number of leaves?

Theorem (Blondin Massé, de Carufel, Goupil, Samson 2017)

For $n \geq 1$, the maximal number of leaves realized by a tree-like polyomino with n cells is given by

$$\ell(n) = \begin{cases} 0 & \text{if } n = 1 \\ 2 & \text{if } n = 2 \\ n - 1 & \text{if } n = 3, 4, 5 \\ \ell(n-4) + 2 & \text{if } n \geq 6 \end{cases}$$

Trees in Minecraft

polyominoes ⇒ polycubes

Trees in Minecraft

tree-like polyominoes ⇒ tree-like polycubes

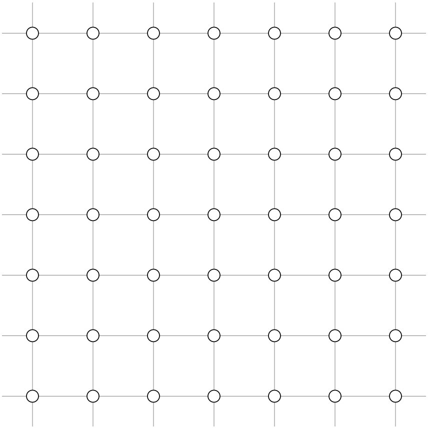
Trees in Minecraft

tree-like polyominoes ⇒ tree-like polycubes

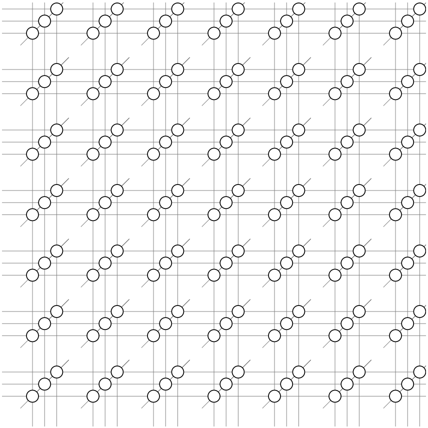
Theorem (Blondin Massé, de Carufel, Goupil, Samson 2017)

The maximal number of leaves realized by a tree-like polycube satisfies a linear recurrence.

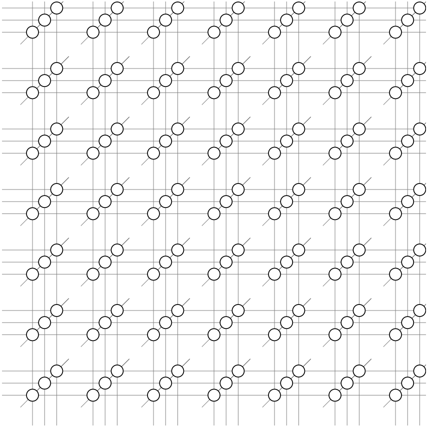
- Tree-like polyominoes are induced subtrees of $\mathbb{Z}^2$.



- Tree-like polyominoes are induced subtrees of $\mathbb{Z}^2$.
- Tree-like polycubes are induced subtrees of $\mathbb{Z}^3$.



- Tree-like polyominoes are induced subtrees of $\mathbb{Z}^2$.
- Tree-like polycubes are induced subtrees of $\mathbb{Z}^3$.

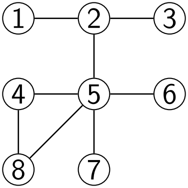


Can we study this problem in other graphs?

Definition

For a graph $G = (V, E)$ and $n \geq 2$

- $\mathcal{T}_n$ = set of induced subtrees with n vertices
- $L_G(n) = \max\{\#\text{ leaves in } T | T \in \mathcal{T}_n\}$
- Leafed sequence of G : $L_G(n)_{n \in \{2,\ldots,|V|\}}$

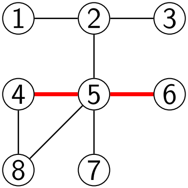


n	2	3	4	5	6	7	8
$L_G(n)$	2						

Definition

For a graph $G = (V, E)$ and $n \geq 2$

- $\mathcal{T}_n$ = set of induced subtrees with n vertices
- $L_G(n) = \max\{\#\text{ leaves in } T | T \in \mathcal{T}_n\}$
- Leafed sequence of G : $L_G(n)_{n \in \{2,\ldots,|V|\}}$

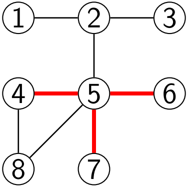


n	2	3	4	5	6	7	8
$L_G(n)$	2	2					

Definition

For a graph $G = (V, E)$ and $n \geq 2$

- $\mathcal{T}_n$ = set of induced subtrees with n vertices
- $L_G(n) = \max\{\#\text{ leaves in } T | T \in \mathcal{T}_n\}$
- Leafed sequence of G : $L_G(n)_{n \in \{2,\ldots,|V|\}}$

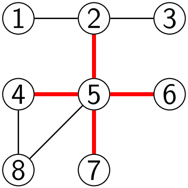


n	2	3	4	5	6	7	8
$L_G(n)$	2	2	3				

Definition

For a graph $G = (V, E)$ and $n \geq 2$

- $\mathcal{T}_n$ = set of induced subtrees with n vertices
- $L_G(n) = \max\{\#\text{ leaves in } T | T \in \mathcal{T}_n\}$
- Leafed sequence of G : $L_G(n)_{n \in \{2,\ldots,|V|\}}$

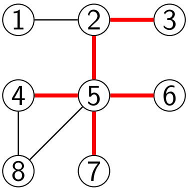


n	2	3	4	5	6	7	8
$L_G(n)$	2	2	3	4			

Definition

For a graph $G = (V, E)$ and $n \geq 2$

- $\mathcal{T}_n$ = set of induced subtrees with n vertices
- $L_G(n) = \max\{\# \text{ leaves in } T | T \in \mathcal{T}_n\}$
- Leafed sequence of G : $L_G(n)_{n \in \{2,\ldots,|V|\}}$

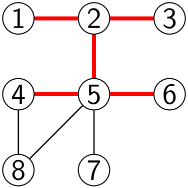


n	2	3	4	5	6	7	8
$L_G(n)$	2	2	3	4	4		

Definition

For a graph $G = (V, E)$ and $n \geq 2$

- $\mathcal{T}_n$ = set of induced subtrees with n vertices
- $L_G(n) = \max\{\#\text{ leaves in } T | T \in \mathcal{T}_n\}$
- Leafed sequence of G : $L_G(n)_{n \in \{2,\ldots,|V|\}}$

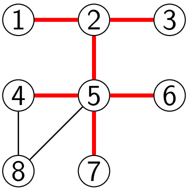


n	2	3	4	5	6	7	8
$L_G(n)$	2	2	3	4	4		

Definition

For a graph $G = (V, E)$ and $n \geq 2$

- $\mathcal{T}_n$ = set of induced subtrees with n vertices
- $L_G(n) = \max\{\#\text{ leaves in } T | T \in \mathcal{T}_n\}$
- Leafed sequence of G : $L_G(n)_{n \in \{2,\ldots,|V|\}}$

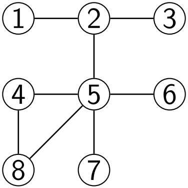


n	2	3	4	5	6	7	8
$L_G(n)$	2	2	3	4	4	5	

Definition

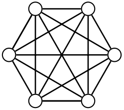
For a graph $G = (V, E)$ and $n \geq 2$

- $\mathcal{T}_n$ = set of induced subtrees with n vertices
- $L_G(n) = \max\{\#\text{ leaves in } T | T \in \mathcal{T}_n\}$
- Leafed sequence of G : $L_G(n)_{n \in \{2,\ldots,|V|\}}$



n	2	3	4	5	6	7	8
$L_G(n)$	2	2	3	4	4	5	$-\infty$

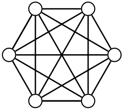
Particular cases

K_6 

$$L_{K_p}(n) = \begin{cases} 2 & \text{if } n = 2 \text{ et } p \geq 2 \\ -\infty & \text{if } 3 \leq n \leq p \end{cases}$$

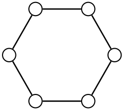
Particular cases

K_6



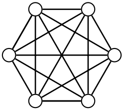
$$L_{K_p}(n) = \begin{cases} 2 & \text{if } n = 2 \text{ et } p \geq 2 \\ -\infty & \text{if } 3 \leq n \leq p \end{cases}$$

$\mathcal{C}_6$

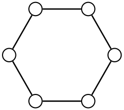


$$L_{C_p}(n) = \begin{cases} 2 & \text{if } 2 \leq n < p \\ -\infty & \text{if } n = p \end{cases}$$

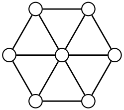
Particular cases

K_6 

$$L_{K_p}(n) = \begin{cases} 2 & \text{if } n = 2 \text{ et } p \geq 2 \\ -\infty & \text{if } 3 \leq n \leq p \end{cases}$$

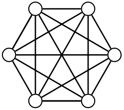
$\mathcal{C}_6$ 

$$L_{C_p}(n) = \begin{cases} 2 & \text{if } 2 \leq n < p \\ -\infty & \text{if } n = p \end{cases}$$

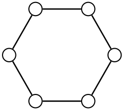
R_6 

$$L_{R_p}(n) = \begin{cases} 2 & \text{if } 2 = n \text{ ou } \lfloor \frac{p}{2} \rfloor + 1 < n < p \\ n-1 & \text{if } 3 \leq n \leq \lfloor \frac{p}{2} \rfloor + 1 \\ -\infty & \text{if } p \leq n \leq p+1 \end{cases}$$

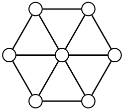
Particular cases

K_6 

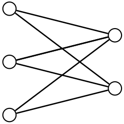
$$L_{K_p}(n) = \begin{cases} 2 & \text{if } n = 2 \text{ et } p \geq 2 \\ -\infty & \text{if } 3 \leq n \leq p \end{cases}$$

$\mathcal{C}_6$ 

$$L_{C_p}(n) = \begin{cases} 2 & \text{if } 2 \leq n < p \\ -\infty & \text{if } n = p \end{cases}$$

R_6 

$$L_{R_p}(n) = \begin{cases} 2 & \text{if } 2 = n \text{ ou } \lfloor \frac{p}{2} \rfloor + 1 < n < p \\ n-1 & \text{if } 3 \leq n \leq \lfloor \frac{p}{2} \rfloor + 1 \\ -\infty & \text{if } p \leq n \leq p+1 \end{cases}$$

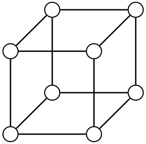
$K_{3,2}$ 

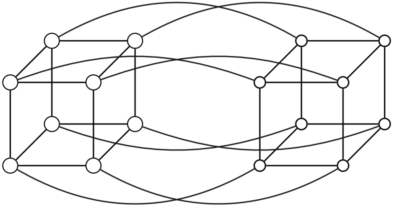
$$L_{K_{p,q}}(n) = \begin{cases} 2 & \text{if } n = 2 \\ n-1 & \text{if } 3 \leq n \leq \max(p,q)+1 \\ -\infty & \text{if } \max(p,q)+1 < n \leq p+q \end{cases}$$

Particular cases: hypercubes

Only partial results:

$$L_{H_3}(n) = \begin{cases} 2 & \text{if } 2 \leq n \leq 3 \\ 3 & \text{if } n = 4 \\ -\infty & \text{if } 5 \leq n \leq 8 \end{cases} \qquad L_{H_4}(n) = \begin{cases} 2 & \text{if } n \in \{2,3\} \\ 3 & \text{if } n \in \{4,6,8\} \\ 4 & \text{if } n \in \{5,7,9\} \\ -\infty & \text{if } 10 \leq n \leq 16 \end{cases}$$





Particular cases: hypercubes

Only partial results:

$$L_{H_3}(n) = \begin{cases} 2 & \text{if } 2 \leq n \leq 3 \\ 3 & \text{if } n = 4 \\ -\infty & \text{if } 5 \leq n \leq 8 \end{cases} \quad L_{H_4}(n) = \begin{cases} 2 & \text{if } n \in \{2,3\} \\ 3 & \text{if } n \in \{4,6,8\} \\ 4 & \text{if } n \in \{5,7,9\} \\ -\infty & \text{if } 10 \leq n \leq 16 \end{cases}$$

Observations:

- The leafed sequence can increase by at most 1 each step.
- The leafed sequence is not always non-decreasing.
- The leafed sequence $L_G(n)_{n\in\{2,\ldots,|V|\}}$ is non-decreasing iff G is a tree.

Complexity

Problem LIS

- Instance: a graph G and two integers $n, k \geq 1$
- Question: Is there an induced subtree of G with n vertices and k leaves?

Complexity

Problem LIS

- Instance: a graph G and two integers $n, k \geq 1$
- Question: Is there an induced subtree of G with n vertices and k leaves?

Theorem (Blondin Massé, de Carufel, Goupil, V. 2017)

The LIS problem is NP-complete.

Complexity

Problem LIS

- Instance: a graph G and two integers $n, k \geq 1$
- Question: Is there an induced subtree of G with n vertices and k leaves?

Theorem (Blondin Massé, de Carufel, Goupil, V. 2017)

The LIS problem is NP-complete.

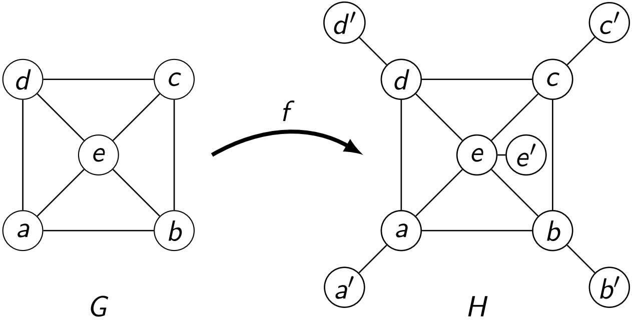
Reduction to the problem:

- Instance: a graph G and an integer $n \geq 1$
- Question: Is there an induced subtree of G with more than n vertices?

which is NP-complet. [Erdös, Saks, Sós 1986]

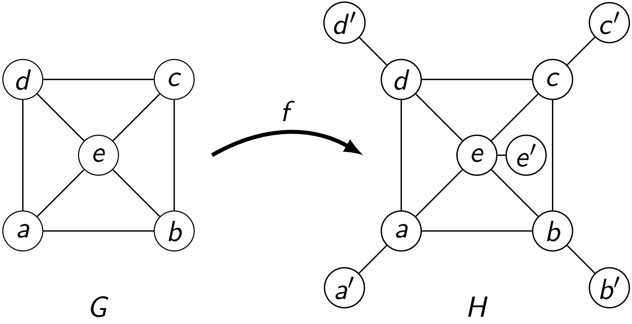
LIS is NP-complete

- Polynomial transformation $f : (G, n) \mapsto (H, 2(n+1), n+1)$



LIS is NP-complete

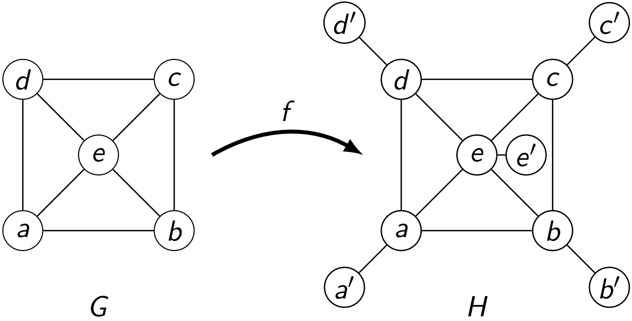
- Polynomial transformation $f : (G, n) \mapsto (H, 2(n+1), n+1)$



- If G has an induced subtree with $> n$ vertices

LIS is NP-complete

- Polynomial transformation $f : (G, n) \mapsto (H, 2(n+1), n+1)$

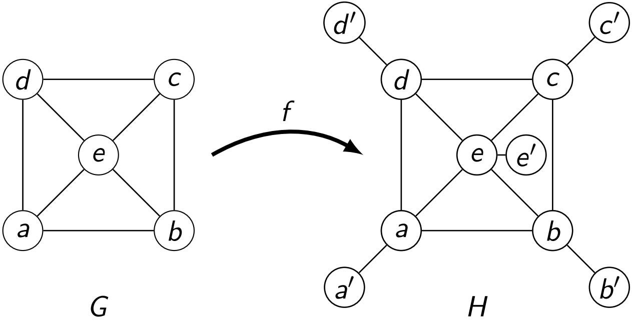


- If G has an induced subtree with $> n$ vertices

$\Rightarrow$ $\exists T$ induced subtree with $n+1$ vertices $\{v_1, \ldots, v_{n+1}\}$

LIS is NP-complete

- Polynomial transformation $f : (G, n) \mapsto (H, 2(n+1), n+1)$



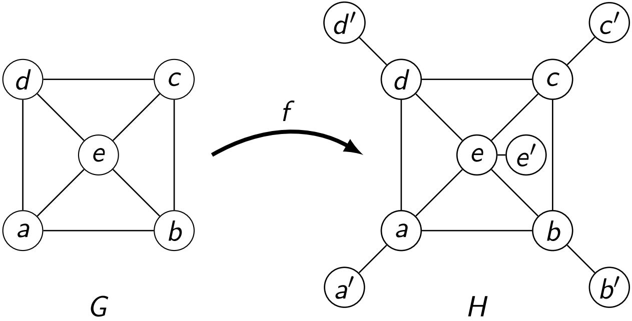
- If G has an induced subtree with $> n$ vertices

$\Rightarrow$ $\exists T$ induced subtree with $n+1$ vertices $\{v_1, \ldots, v_{n+1}\}$

$\Rightarrow$ H has a subtree induced by $\{v_1, \ldots, v_{n+1}\} \cup \{v'_1, \ldots, v'_{n+1}\}$

LIS is NP-complete

- Polynomial transformation $f : (G, n) \mapsto (H, 2(n+1), n+1)$



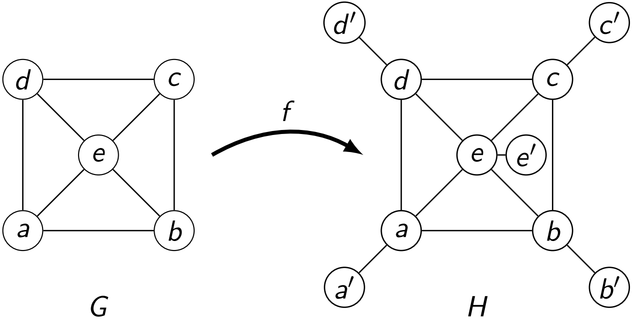
- If G has an induced subtree with $> n$ vertices

$\Rightarrow$ $\exists T$ induced subtree with $n+1$ vertices $\{v_1, \ldots, v_{n+1}\}$

$\Rightarrow$ H has a subtree induced by $\{v_1, \ldots, v_{n+1}\} \cup \{v'_1, \ldots, v'_{n+1}\}$ with $2(n+1)$ vertices and $n+1$ leaves

LIS is NP-complete

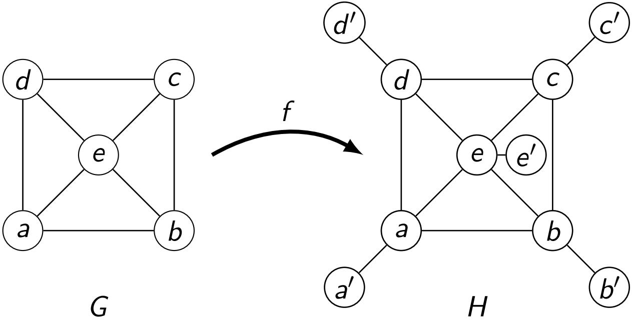
- Polynomial transformation $f : (G, n) \mapsto (H, 2(n+1), n+1)$



- If H has an induced subtree T with $2(n+1)$ vertices and $n+1$ leaves

LIS is NP-complete

- Polynomial transformation $f : (G, n) \mapsto (H, 2(n+1), n+1)$

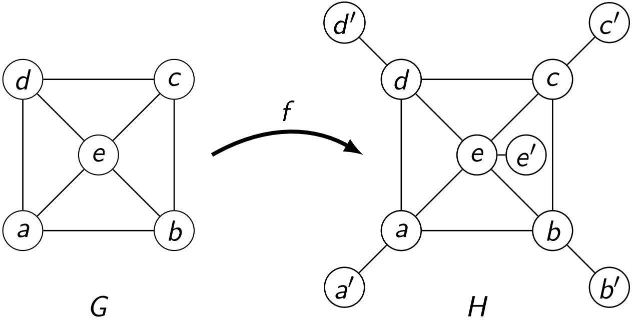


- If H has an induced subtree T with $2(n+1)$ vertices and $n+1$ leaves

$\Rightarrow$ T has $\geq n+1$ vertices in V

LIS is NP-complete

- Polynomial transformation $f : (G, n) \mapsto (H, 2(n+1), n+1)$



- If H has an induced subtree T with $2(n+1)$ vertices and $n+1$ leaves

$\Rightarrow$ T has $\geq n+1$ vertices in V

$\Rightarrow$ G has an induced subtree with $> n$ vertices

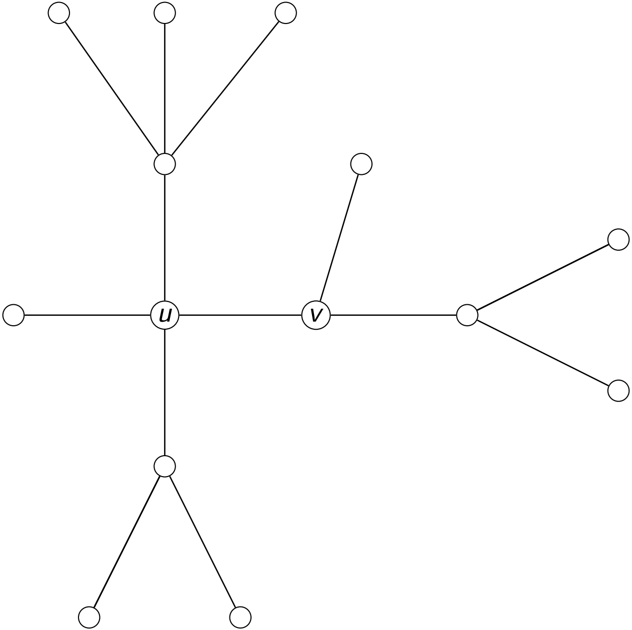
What about trees?

Theorem (Blondin Massé, de Carufel, Goupil, V. 2017)

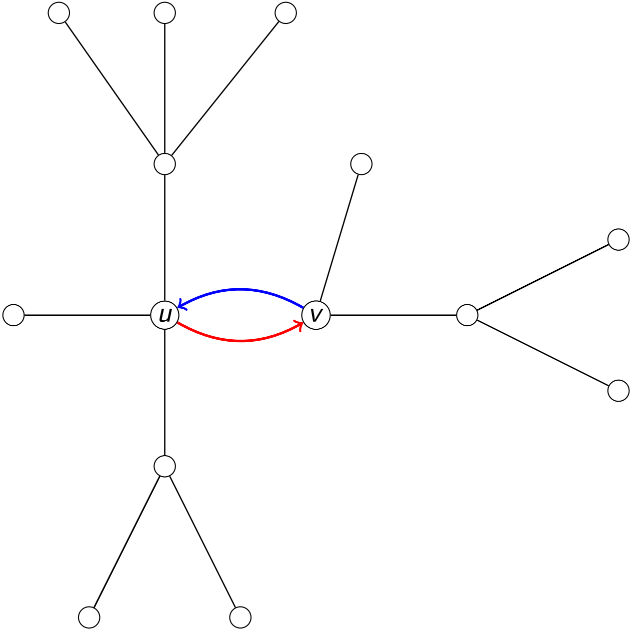
For a tree T with m vertices, the leafed sequence $L_T(n)$ is computed in polynomial time and space.

Algorithm based on the dynamic programming paradigm

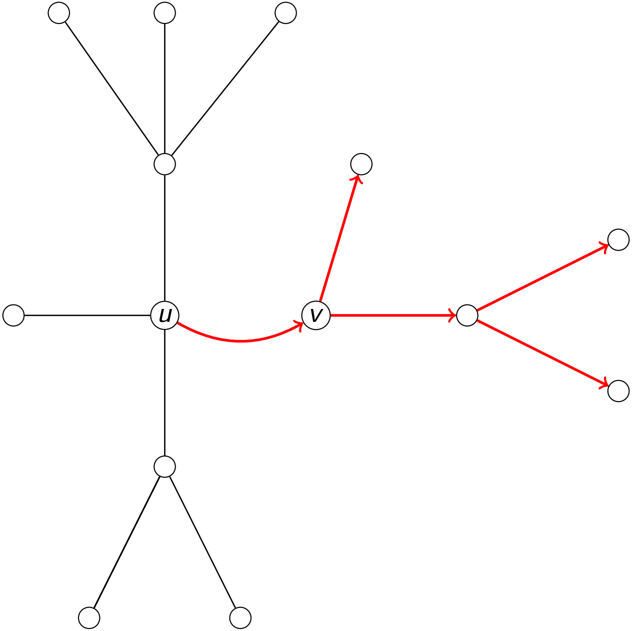
Idea of the algorithm



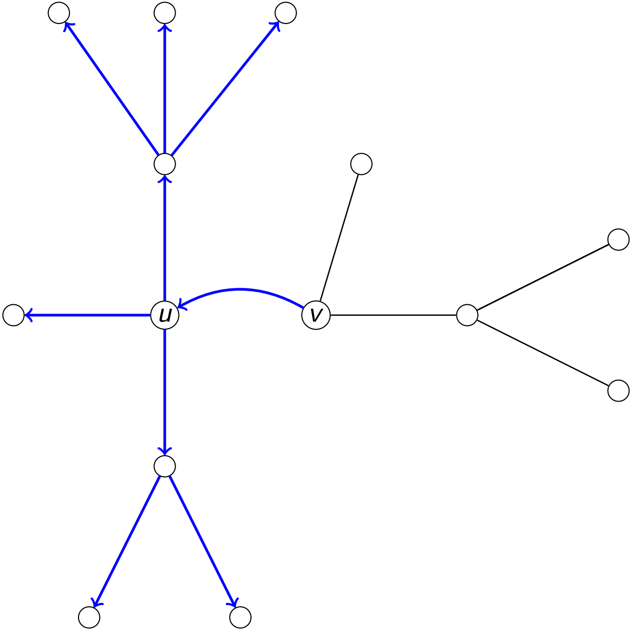
Idea of the algorithm



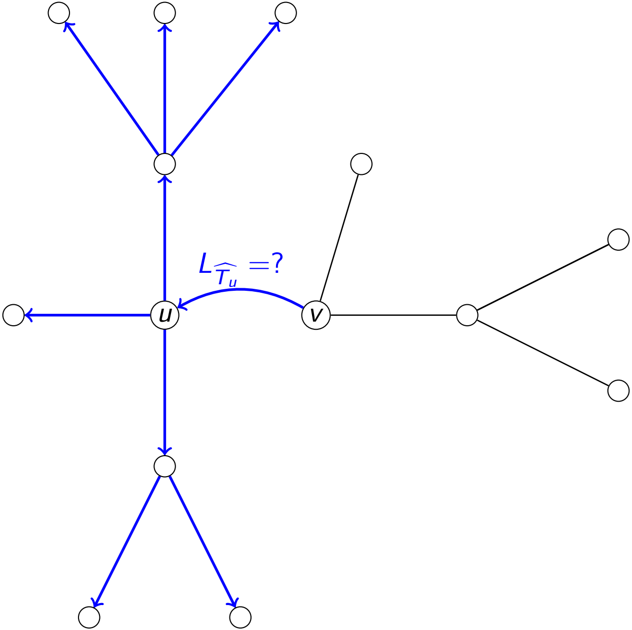
Idea of the algorithm



Idea of the algorithm



Idea of the algorithm



Idea of the algorithm

For a directed tree $\widehat{T_u}$ rooted in u

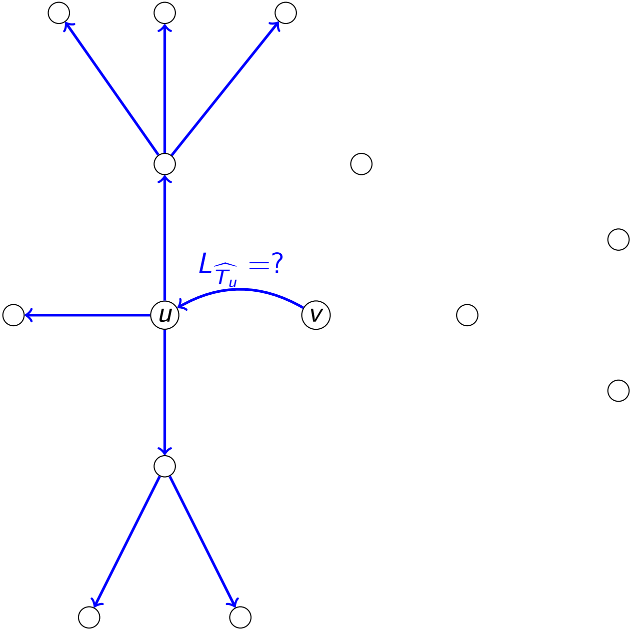
- $f(\widehat{T_u}) = \#\{x \in \widehat{T_u} \mid \deg^+(x) = 0\}$
- $L_{\widehat{T_u}}(n) = \max\{f(\widehat{T'_u}) : \widehat{T'_u} \subseteq \widehat{T}_u, |\widehat{T'_u}| = n\}$
- Generalization to a directed forest $\widehat{F}$ with k connected components $\widehat{F}_i$:

$$L_{\widehat{F}}(n) = \max \left\{ \sum_{i=1}^{k} L_{\widehat{F}_i}(\lambda(i)) \;\middle|\; \lambda \in C(n, k) \right\}$$

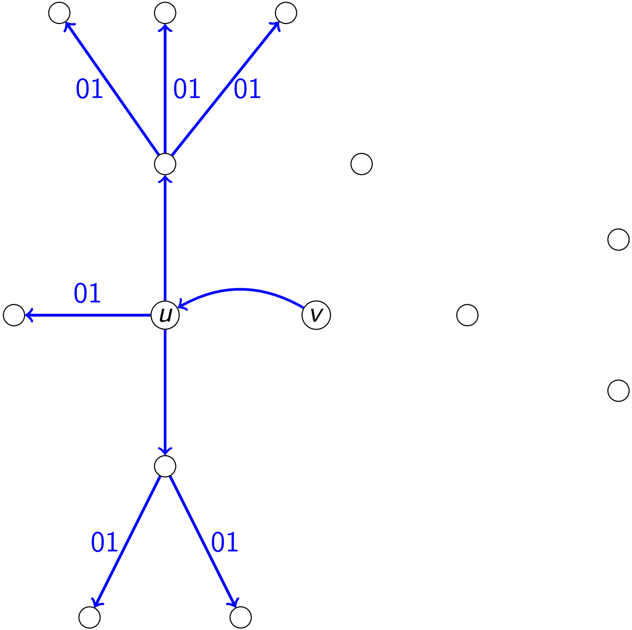
- If $\widehat{F}$ is the forest of the subtrees rooted in the children of u,

$$L_{\widehat{T_u}}(n) = \begin{cases} n & \text{if } n = 0, 1 \\ L_{\widehat{F}}(n-1) & \text{otherwise} \end{cases}$$

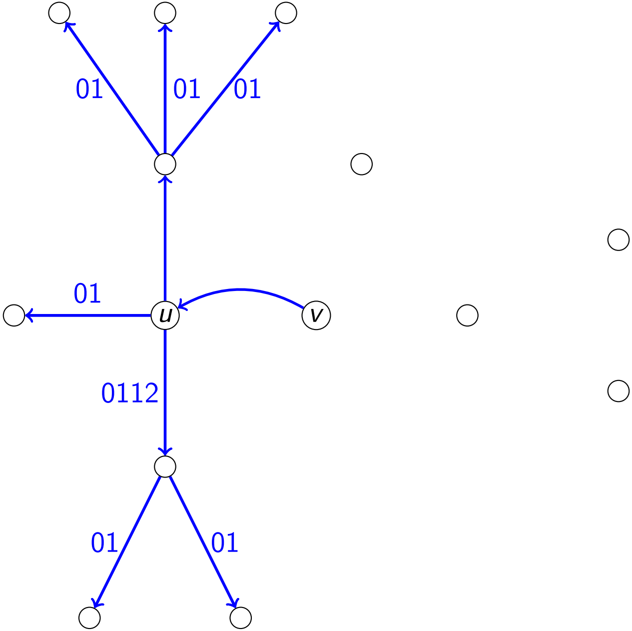
Idea of the algorithm



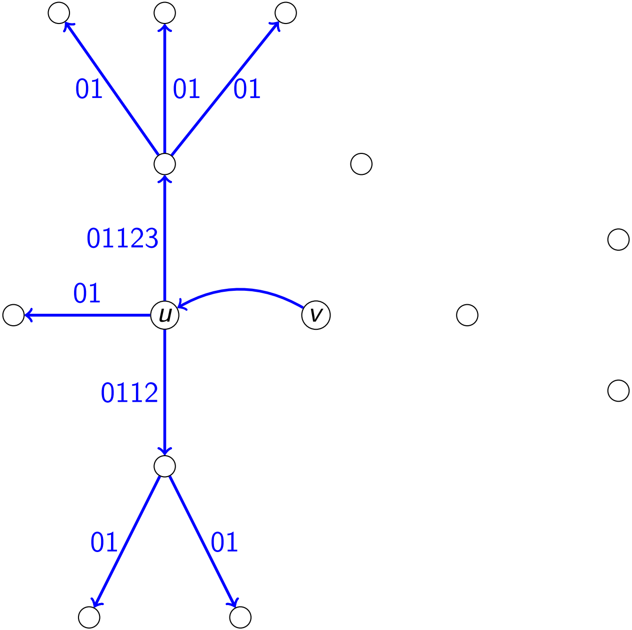
Idea of the algorithm



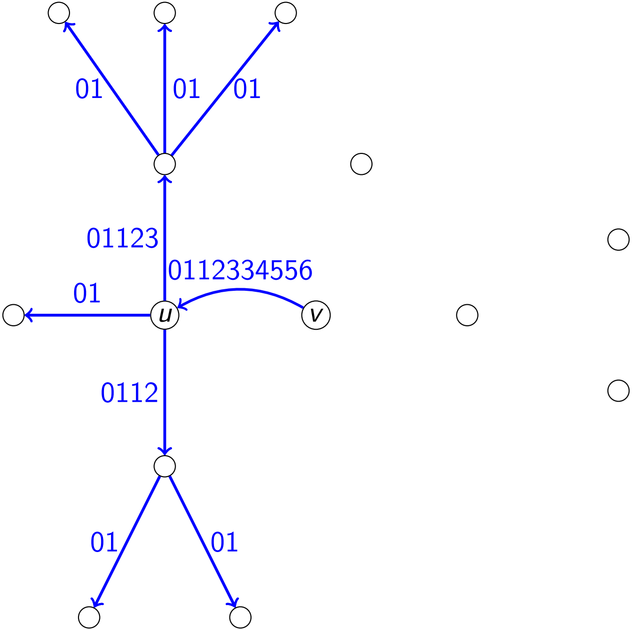
Idea of the algorithm



Idea of the algorithm



Idea of the algorithm



Idea of the algorithm

Assuming $L_{\widehat{F}_i}$ are known,

- Naive computation of $L_{\widehat{F}}$ via

$$L_{\widehat{F}}(n) = \max\left\{ \sum_{i=1}^{k} L_{\widehat{F}_i}(\lambda(i)) \;\middle|\; \lambda \in C(n, k) \right\}$$

  is not polynomial

Idea of the algorithm

Assuming $L_{\widehat{F_i}}$ are known,

- Naive computation of $L_{\widehat{F}}$ via

$$L_{\widehat{F}}(n) = \max \left\{ \sum_{i=1}^{k} L_{\widehat{F_i}}(\lambda(i)) \;\middle|\; \lambda \in C(n, k) \right\}$$

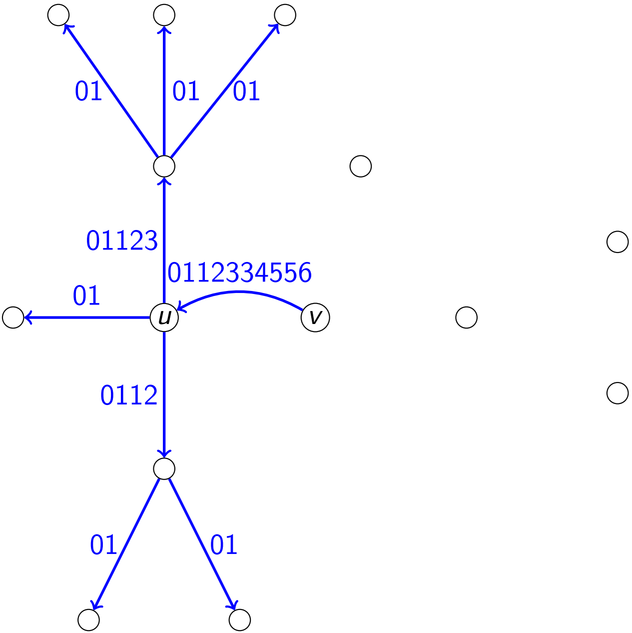
  is not polynomial

- Computation of $L_{\widehat{F}}$ via

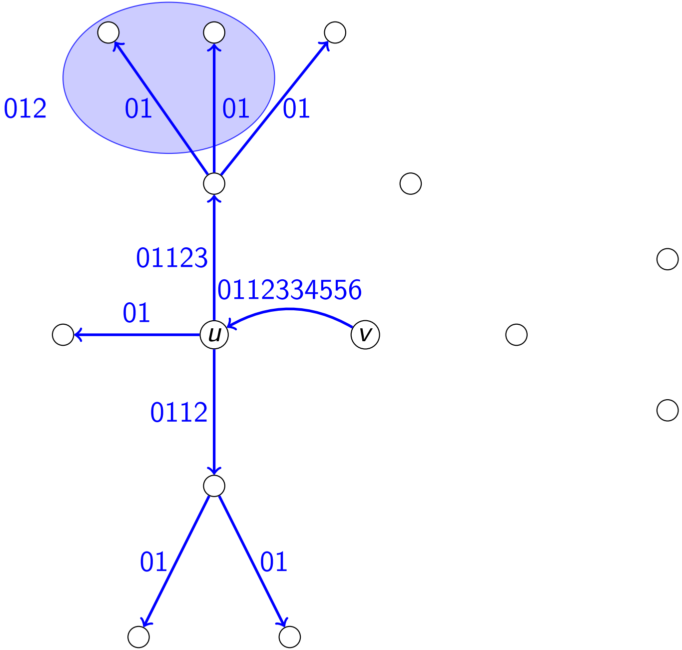
$$L_{\widehat{F}}(n) = \max\{L_{\widehat{F-F_1}}(i) + L_{\widehat{F_1}}(n-i) \mid 0 \leq i \leq n\}$$

  in time $\Theta(k|F|^2)$

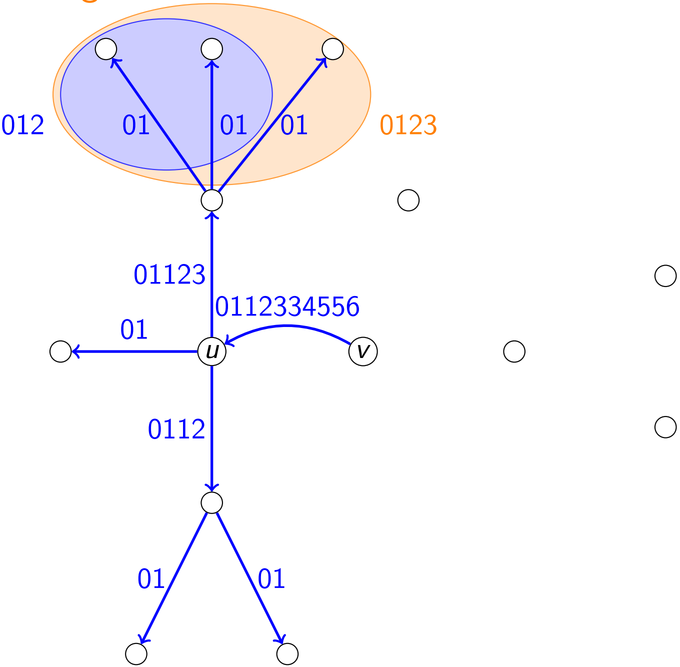
Idea of the algorithm



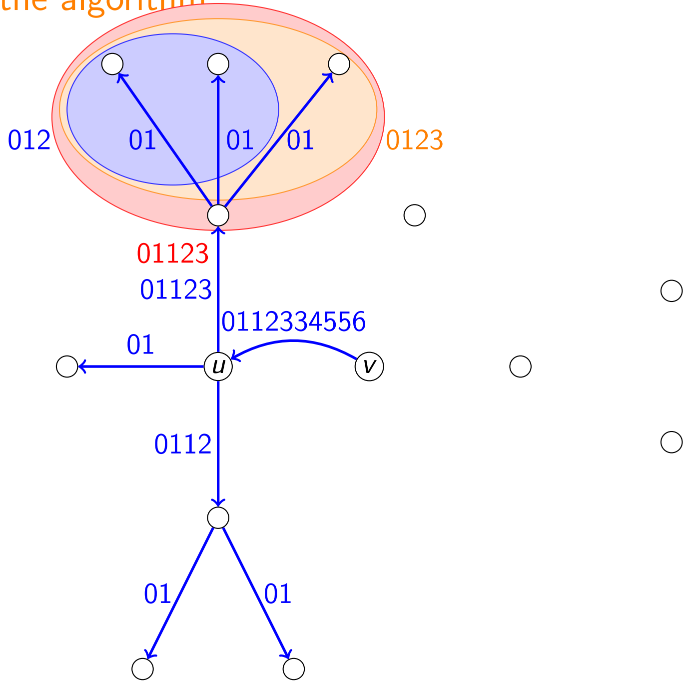
Idea of the algorithm



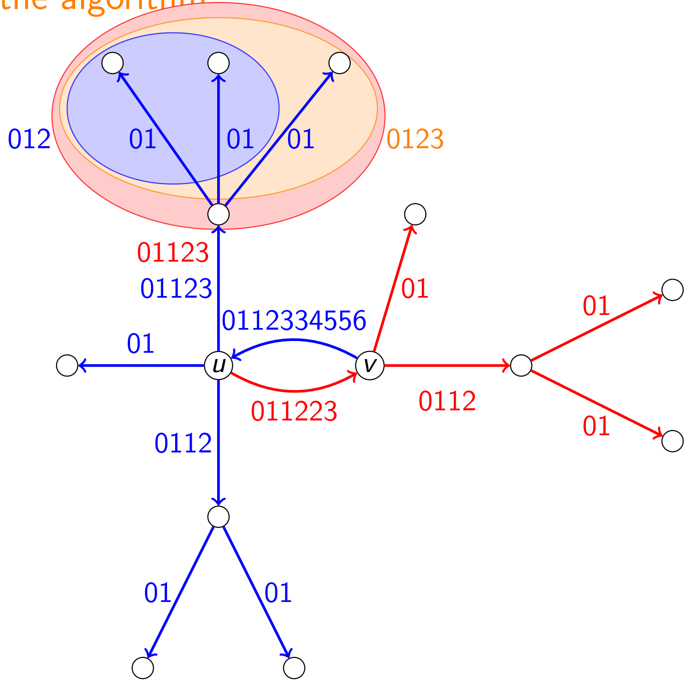
Idea of the algorithm



Idea of the algorithm



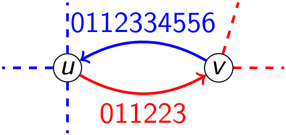
Idea of the algorithm



Idea of the algorithm

- Maximal number of leaves in the subtrees with n vertices, containing the edge $\{u, v\}$:

$$L_{\{u,v\}}(n) = \max\left\{ L_{\widehat{T_u}}(i) + L_{\widehat{T_v}}(n-i) \;\middle|\; 1 \leq i \leq n-1 \right\}$$

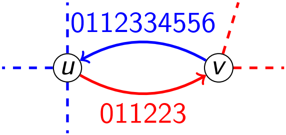


$$L_{\{u,v\}} = 01223445667 7889$$

Idea of the algorithm

- Maximal number of leaves in the subtrees with n vertices, containing the edge $\{u, v\}$:

$$L_{\{u,v\}}(n) = \max\left\{L_{\widehat{T_u}}(i) + L_{\widehat{T_v}}(n-i) \;\middle|\; 1 \leq i \leq n-1\right\}$$

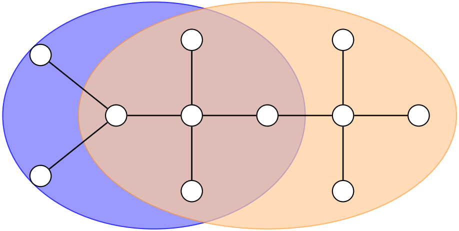


$$L_{\{u,v\}} = 012234456677889$$

- $L_T(n) = \max\left\{L_{\{u,v\}}(n) \;\middle|\; \{u,v\} \in E\right\}$

Could we improve the time and space complexity?

- We can not hope to obtain a procedure computing $L_T(n)$ which deletes leaves successively.
- Counter-example :



$L_T(7) = 5$ et $L_T(9) = 6$

From graph theory to combinatorics on words

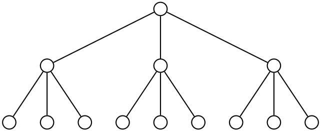
- Each tree has a non-decreasing leaf sequence $L = (L_i)_{i \geq 2}$.
- For any graph, $L_{i+1} - L_i \leq 1$.

The difference word $\Delta L := (L_{i+1} - L_i)_{i \geq 3}$ is a binary word.

From graph theory to combinatorics on words

- Each tree has a non-decreasing leaf sequence $L = (L_i)_{i \geq 2}$.
- For any graph, $L_{i+1} - L_i \leq 1$.

The difference word $\Delta L := (L_{i+1} - L_i)_{i \geq 3}$ is a binary word.

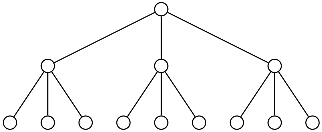


- Leafed sequence: $L = 223445567789$
- Difference word: $\Delta L = 01101011011$

From graph theory to combinatorics on words

- Each tree has a non-decreasing leaf sequence $L = (L_i)_{i \geq 2}$.
- For any graph, $L_{i+1} - L_i \leq 1$.

The difference word $\Delta L := (L_{i+1} - L_i)_{i \geq 3}$ is a binary word.

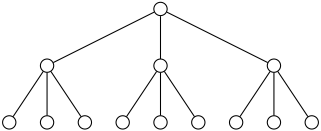


- Leafed sequence: $L = 223445567789$
- Difference word: $\Delta L = \ 1101011011$

From graph theory to combinatorics on words

- Each tree has a non-decreasing leaf sequence $L = (L_i)_{i \geq 2}$.
- For any graph, $L_{i+1} - L_i \leq 1$.

The difference word $\Delta L := (L_{i+1} - L_i)_{i \geq 3}$ is a binary word.



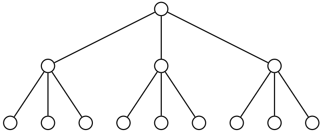
- Leafed sequence: $L = 223445567789$
- Difference word: $\Delta L = \; 1101011011$

The language of a class of graphs is the set of all their difference words.

From graph theory to combinatorics on words

- Each tree has a non-decreasing leaf sequence $L = (L_i)_{i \geq 2}$.
- For any graph, $L_{i+1} - L_i \leq 1$.

The difference word $\Delta L := (L_{i+1} - L_i)_{i \geq 3}$ is a binary word.



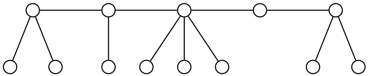
- Leafed sequence: $L = 223445567789$
- Difference word: $\Delta L = \ \ 1101011011$

The language of a class of graphs is the set of all their difference words.

Language of Trees =???

Caterpillars

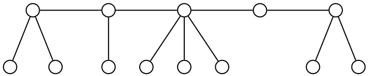
Caterpillars



$L = 223455667778$ and $\Delta L = 1110101001$

What is the language of caterpillars ?

Caterpillars

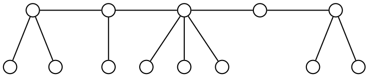


$$L = 223455667778 \text{ and } \Delta L = 1110101001$$

What is the language of caterpillars ?

Language of caterpillars $\subsetneq$ Language of trees

Caterpillars



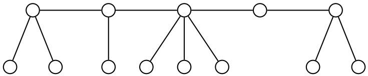
$$L = 223455667778 \text{ and } \Delta L = 1110101001$$

What is the language of caterpillars ?

Language of caterpillars $\subsetneq$ Language of trees

- Enumerate caterpillars

Caterpillars



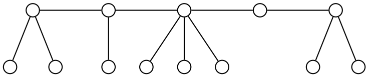
$$L = 223455667778 \text{ and } \Delta L = 1110101001$$

What is the language of caterpillars ?

Language of caterpillars $\subsetneq$ Language of trees

- Enumerate caterpillars
- Get their associated difference words

Caterpillars



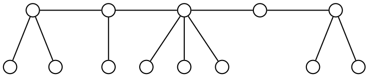
$L = 223455667778$ and $\Delta L = 1110101001$

What is the language of caterpillars ?

Language of caterpillars $\subsetneq$ Language of trees

- Enumerate caterpillars
- Get their associated difference words
- ε, 0, 1, 00, 10, 11, 0000, 1000, 1001, 1010, 1100, 1101, ...

Caterpillars



$L = 223455667778$ and $\Delta L = 1110101001$

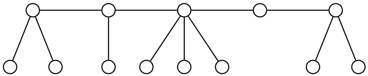
What is the language of caterpillars ?

Language of caterpillars $\subsetneq$ Language of trees

- Enumerate caterpillars
- Get their associated difference words
- ε, 0, 1, 00, 10, 11, 0000, 1000, 1001, 1010, 1100, 1101, ...
- Count them by length

n	0	1	2	3	4	5	6	7	8	9
	1	2	3	5	8	14	23	41	70	125

Caterpillars



$L = 223455667778$ and $\Delta L = 1110101001$

What is the language of caterpillars ?

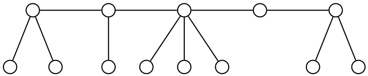
Language of caterpillars $\subsetneq$ Language of trees

- Enumerate caterpillars
- Get their associated difference words
- ε, 0, 1, 00, 10, 11, 0000, 1000, 1001, 1010, 1100, 1101, ...
- Count them by length

n	0	1	2	3	4	5	6	7	8	9
	1	2	3	5	8	14	23	41	70	125

- Check OEIS

Caterpillars



$L = 223455667778$ and $\Delta L = 1110101001$

What is the language of caterpillars ?

Language of caterpillars $\subsetneq$ Language of trees

- Enumerate caterpillars
- Get their associated difference words
- ε, 0, 1, 00, 10, 11, 0000, 1000, 1001, 1010, 1100, 1101, ...
- Count them by length

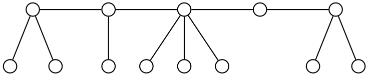
n	0	1	2	3	4	5	6	7	8	9
	1	2	3	5	8	14	23	41	70	125

- Check OEIS It is A194850!

Prefix normal word

A prefix normal word $w_1 \cdots w_n$ is a binary word such that $\forall \ell \leq n, i \leq n - \ell,$

$$|w_1 \cdots w_\ell|_1 \geq |w_i \cdots w_{i+\ell}|_1.$$

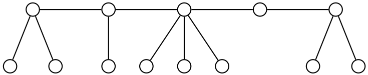


$L = 223455667778$ and $\Delta L = 1110101001$

Prefix normal word

A prefix normal word $w_1 \cdots w_n$ is a binary word such that $\forall \ell \leq n, i \leq n - \ell$,

$$|w_1 \cdots w_\ell|_1 \geq |w_i \cdots w_{i+\ell}|_1.$$



$$L = 223455667778 \text{ and } \Delta L = 1110101001$$

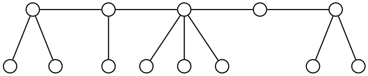
Conjecture:

$$\text{Language of Caterpillars} = \{\text{prefix normal words}\}$$

Prefix normal word

A prefix normal word $w_1 \cdots w_n$ is a binary word such that $\forall \ell \leq n, i \leq n - \ell$,

$$|w_1 \cdots w_\ell|_1 \geq |w_i \cdots w_{i+\ell}|_1.$$



$$L = 223455667778 \text{ and } \Delta L = 1110101001$$

Conjecture:

$$\text{Language of Caterpillars} = \{\text{prefix normal words}\}$$

Proposition (Blondin Massé, de Carufel, Goupil, Lapointe, V.)

$$\text{Language of Caterpillars} \supseteq \{\text{prefix normal words}\}$$

Perspectives

Perspectives

- Determine when an integer sequence corresponds to a tree or to a graph

Perspectives

- Determine when an integer sequence corresponds to a tree or to a graph
- Find an application in Chemistry or Biology

Perspectives

- Determine when an integer sequence corresponds to a tree or to a graph
- Find an application in Chemistry or Biology
- Extend the results obtained in $\mathbb{Z}^2$ to the infinite triangular grid (polyiamonds) and to the infinite hexagonal grid (polyhexes)