

# Routing and connectivity problems in forbidden-transition graphs

Thomas Bellitto

Tuesday, July 4, 2023

Sorbonne Université, LIP6

based on joint works with

Jørgen Bang-Jensen

Bruno Escoffier

Marcin Pilipczuk

Benjamin Bergougnoux

Shaohua Li

Manuel Sorge

Cyril Conchon-Kerjan

Karolina Okrasa

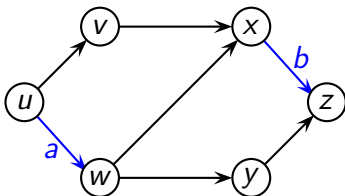
Anders Yeo

- 1 Forbidden-transition graphs
- 2 Edge-coloured graphs
- 3 Parametrized complexity
- 4 Exploring temporal graphs

# Traffic Monitoring

*Traffic Monitoring* : install sensors on the arcs of a directed so we can distinguish a set of possible walks.

Example : we want to distinguish the walks from  $u$  to  $z$ .

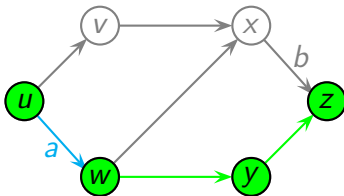


Walk      Signature

# Traffic Monitoring

*Traffic Monitoring* : install sensors on the arcs of a directed so we can distinguish a set of possible walks.

Example : we want to distinguish the walks from  $u$  to  $z$ .

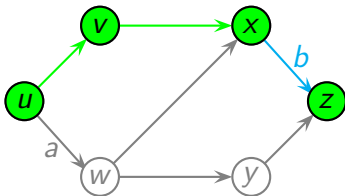


Walk	Signature
$uwyz$	$a$

# Traffic Monitoring

*Traffic Monitoring* : install sensors on the arcs of a directed so we can distinguish a set of possible walks.

Example : we want to distinguish the walks from  $u$  to  $z$ .

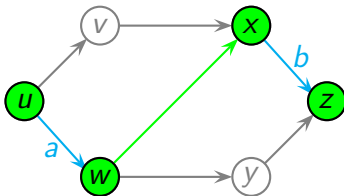


Walk	Signature
$uwyz$	$a$
$uvxz$	$b$

# Traffic Monitoring

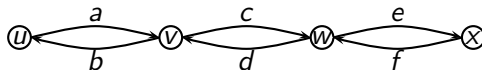
*Traffic Monitoring* : install sensors on the arcs of a directed so we can distinguish a set of possible walks.

Example : we want to distinguish the walks from  $u$  to  $z$ .

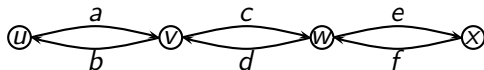


Walk	Signature
$uwyz$	$a$
$uvxz$	$b$
$uwxz$	$ab$

# Limitation of the classical graph model



# Limitation of the classical graph model



Set of possible walks in a road network  $\neq$  set of walks in the graph.

- increases the computation time, decreases the size of the instance we can solve
- lower the quality of the solutions

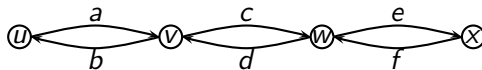
**We need a stronger model!**



# Forbidden-transition graphs

- **Transition** : pair of adjacent distinct edges.
- **Forbidden-transition graph** :  $G = (V, E, T)$  where  $T$  is the set of permitted transitions. Introduced by Kotzig in 1968.
- The walk  $W = (v_1, v_2, \dots, v_k)$  *uses* the transition  $v_i v_{i+1} v_{i+2}$  for all  $i$  such that  $v_i \neq v_{i+2}$ .
- The walk  $W$  is  *$T$ -compatible* if and only if it only uses transitions from  $T$ .

# Examples of applications

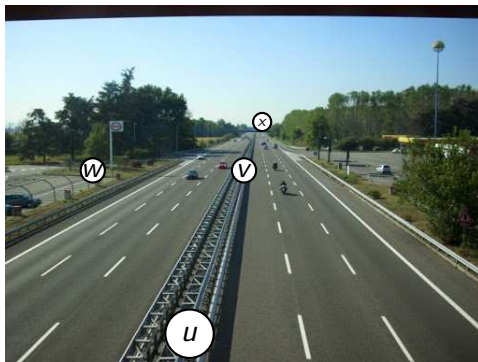


We can forbid  $ab$ ,  $ba$ ,  $cd$ ,  $dc$ ...

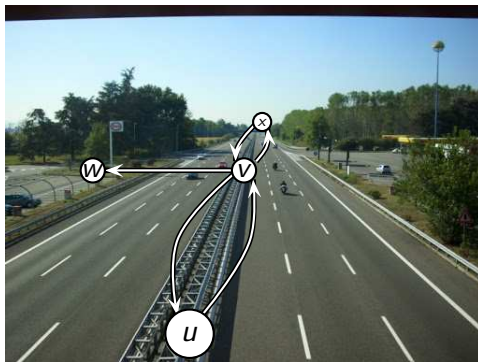
# Examples of applications



# Examples of applications

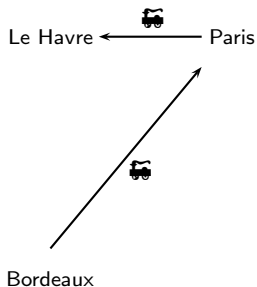


# Examples of applications

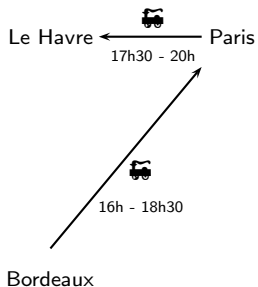


The transition  $uvw$  is forbidden.

# Examples of applications



# Examples of applications



# Project

Project : get a better understanding of forbidden-transition graphs. Develop a toolbox to study important problems on them.

- Eulerian/Hamiltonian cycles, travelling salesman problem.
- Linkage (looking for disjoint paths between a set of pairs of terminals).
- Shortest path, elementary path.
- Connectivity, robustness.
- Many others...



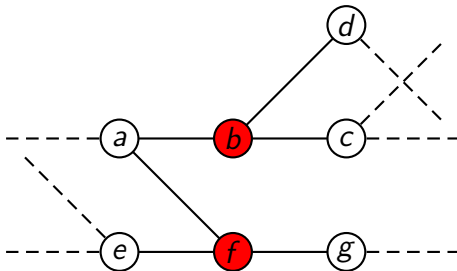
# Difficulty

Problem : everything is much harder is forbidden-transition graphs.

## Theorem (Szeider, 2003)

Let  $G = (V, E, T)$  be a forbidden-transition graph and let  $u$  and  $v$  be vertices of  $V$ . It is NP-complete to determine if there exists an elementary path between  $u$  and  $v$  compatible with  $T$ .

# Quick proof of Szeider's theorem

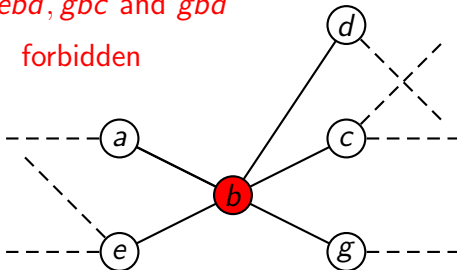


Is there a path between  $u$  and  $v$  that uses at most one red vertex?

# Quick proof of Szeider's theorem

*ebc, ebd, gbc and gbd*

forbidden



Is there a path between  $u$  and  $v$  that uses at most one red vertex?

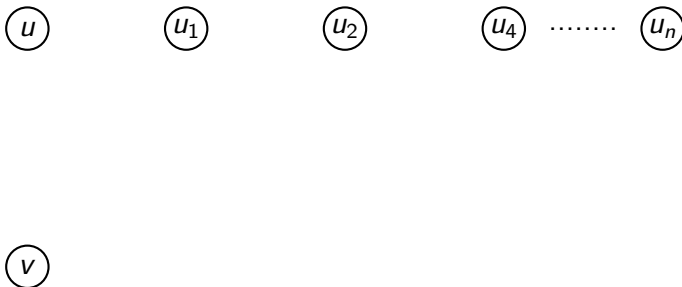
Is there a compatible elementary path between  $u$  and  $v$ ?

# Quick proof of Szeider's theorem

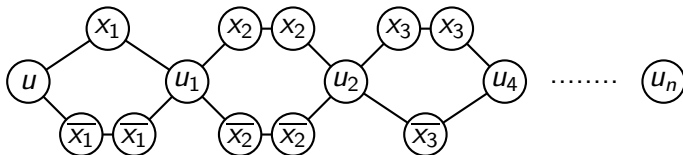
$u$

$v$

# Quick proof of Szeider's theorem

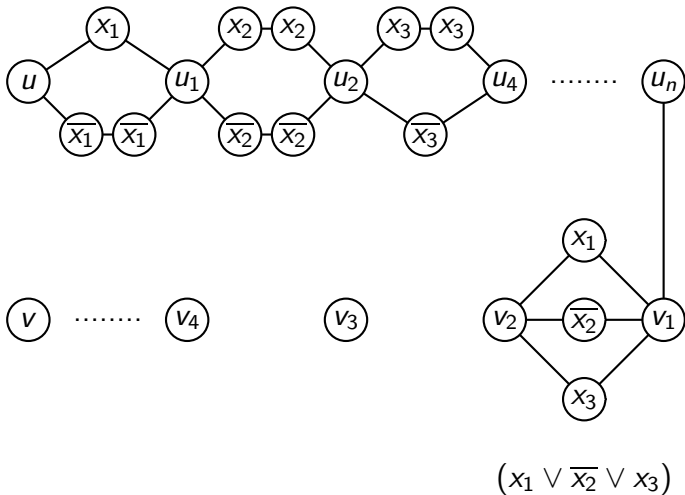


# Quick proof of Szeider's theorem



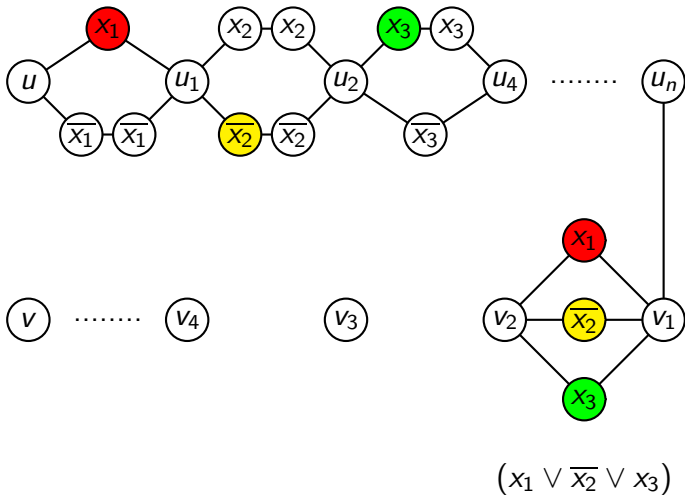


# Quick proof of Szeider's theorem

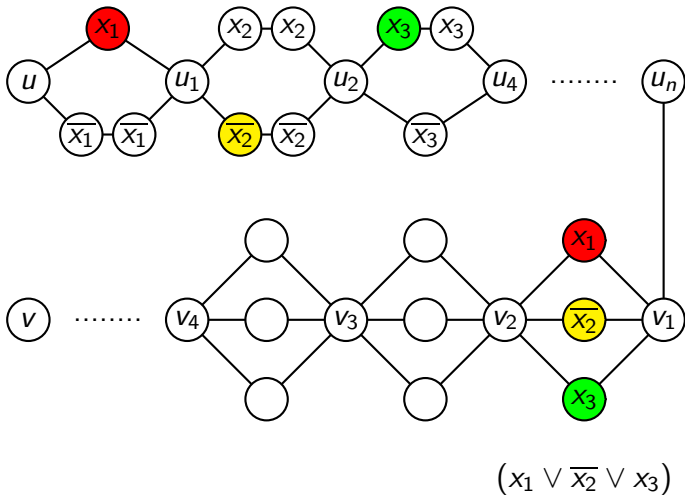




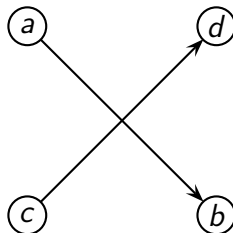
# Quick proof of Szeider's theorem



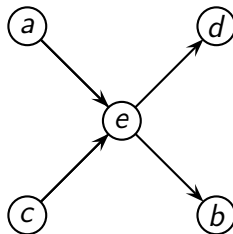
# Quick proof of Szeider's theorem



# Planarization

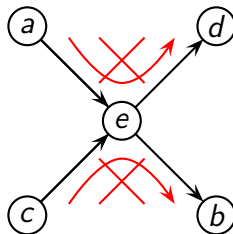


# Planarization



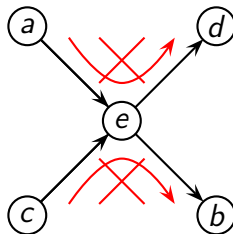
# Planarization

Forbidden transitions :  $aed, ceb$



# Planarization

Forbidden transitions : *aed, ceb*



Theorem (Kratochvíl, Lubiw, Nešetřil, 1991)

Given a graph  $G$  drawn in the plane, it is NP-complete to decide if there exists a non-crossing path between two vertices  $u$  and  $v$ .

# $T$ -connectivity

## $T$ -connectivity

A graph  $G = (V, E)$  is  $T$ -connected if and only if there exists a  $T$ -compatible walk between each pair of vertices.

In this case, we say that  $T$  is a **connecting transition set** of  $G$ .

## Minimum connecting transition set

Smallest set  $T$  such that a graph  $G$  is  $T$ -connected.  
(similar to minimum spanning trees)

# Results (with Benjamin Bergougnoux)

## Complexity

Finding a minimum connecting transition set of a graph is NP-complete.

## Approximation

There exists a  $\frac{3}{2}$ -approximation that runs in time  $O(|V|^2)$ .



- 1 Forbidden-transition graphs
- 2 Edge-coloured graphs
  - Definition
  - Hamiltonicity and Eulerianicity
  - Proper connection number
- 3 Parametrized complexity
- 4 Exploring temporal graphs

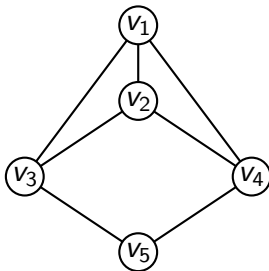
# Variants of forbidden-transition graphs

- Forbidden-transition directed graphs, multigraphs...
- Antidirected walks.
- Forbidden subpaths, subwalks...
- Non-crossing walks.
- Journeys in temporal graphs.
- Properly-coloured walks in edge-coloured graphs. ...

# Edge-colouring

Let  $G = (V, E)$  be an undirected graph.

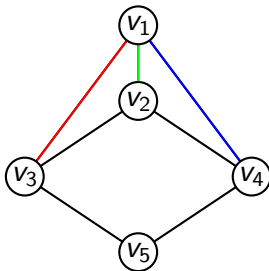
- $k$ -edge-colouring : function  $c : E \mapsto [1, k]$
- proper edge-colouring :  $c(uv) \neq c(vw)$ .



# Edge-colouring

Let  $G = (V, E)$  be an undirected graph.

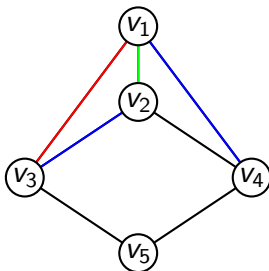
- $k$ -edge-colouring : function  $c : E \mapsto [1, k]$
- proper edge-colouring :  $c(uv) \neq c(vw)$ .



# Edge-colouring

Let  $G = (V, E)$  be an undirected graph.

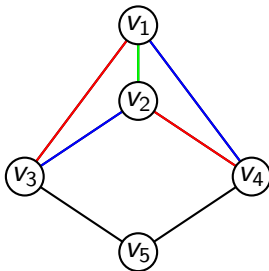
- $k$ -edge-colouring : function  $c : E \mapsto [1, k]$
- proper edge-colouring :  $c(uv) \neq c(vw)$ .



# Edge-colouring

Let  $G = (V, E)$  be an undirected graph.

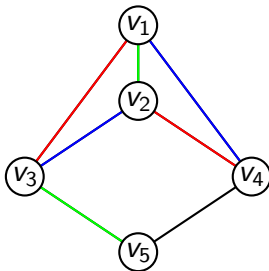
- $k$ -edge-colouring : function  $c : E \mapsto [1, k]$
- proper edge-colouring :  $c(uv) \neq c(vw)$ .



# Edge-colouring

Let  $G = (V, E)$  be an undirected graph.

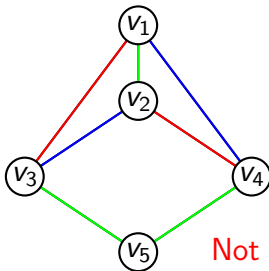
- $k$ -edge-colouring : function  $c : E \mapsto [1, k]$
- proper edge-colouring :  $c(uv) \neq c(vw)$ .



# Edge-colouring

Let  $G = (V, E)$  be an undirected graph.

- $k$ -edge-colouring : function  $c : E \mapsto [1, k]$
- proper edge-colouring :  $c(uv) \neq c(vw)$ .

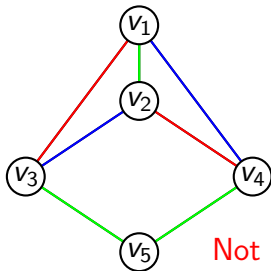




# Edge-colouring

Let  $G = (V, E)$  be an undirected graph.

- $k$ -edge-colouring : function  $c : E \mapsto [1, k]$
- proper edge-colouring :  $c(uv) \neq c(vw)$ .



- Properly-coloured walk : does not use consecutively two edges of the same colour.

# Context

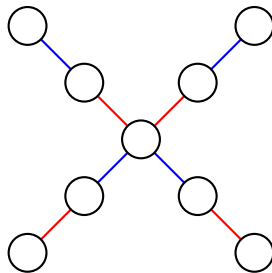
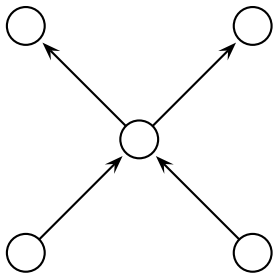
- Introduced by Chen and Daykin in 1976.

# Context

- Introduced by Chen and Daykin in 1976.
- Applications in bioinformatics and chemistry.

# Context

- Introduced by Chen and Daykin in 1976.
- Applications in bioinformatics and chemistry.
- Powerful model.



# An open problem

Problem : does a graph  $G$  have a Hamiltonian cycle?

- Already NP-hard in classical graphs.
- Even harder in edge-coloured or forbidden-transition graphs.

# An open problem

Problem : does a graph  $G$  have a Hamiltonian cycle?

- Already NP-hard in classical graphs.
- Even harder in edge-coloured or forbidden-transition graphs.

What if  $G$  is complete?

# An open problem

Problem : does a graph  $G$  have a Hamiltonian cycle?

- Already NP-hard in classical graphs.
- Even harder in edge-coloured or forbidden-transition graphs.

What if  $G$  is complete?

- Trivial in classical graphs.

# An open problem

Problem : does a graph  $G$  have a Hamiltonian cycle?

- Already NP-hard in classical graphs.
- Even harder in edge-coloured or forbidden-transition graphs.

What if  $G$  is complete?

- Trivial in classical graphs.
- Not easier than in the general case in forbidden-transition graphs.



# An open problem

Problem : does a graph  $G$  have a Hamiltonian cycle?

- Already NP-hard in classical graphs.
- Even harder in edge-coloured or forbidden-transition graphs.

What if  $G$  is complete?

- Trivial in classical graphs.
- Not easier than in the general case in forbidden-transition graphs.
- Polynomial in 2-edge-coloured graphs. Open with  $k \geq 3$  colours.

Also open in edge-coloured complete multipartite graphs even with 2 colours.

# Spanning Eulerian subgraphs

## General case (with Jørgen Bang-Jensen and Anders Yeo)

It is NP-complete to decide if a 2-edge-coloured graph is supereulerian.

## Polynomial cases (with Jørgen Bang-Jensen and Anders Yeo)

One can determine in polynomial time if a 2-edge-coloured complete graph is supereulerian. This also holds for extension of  $M$ -complete graphs.

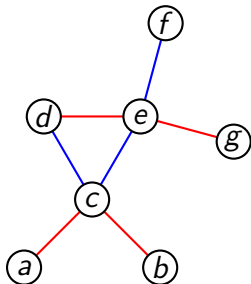
- Open for 2-edge-coloured multipartite graphs.
- Open for 3-edge-coloured complete graphs.

# Proper connectivity

An edge-coloured graph  $G_c = (V, E, c)$  is **properly-connected** if and only if there is a properly-coloured walk from every vertex to every other.

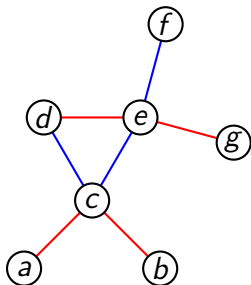
# Proper connectivity

An edge-coloured graph  $G_c = (V, E, c)$  is **properly-connected** if and only if there is a properly-coloured walk from every vertex to every other.



# Proper connectivity

An edge-coloured graph  $G_c = (V, E, c)$  is **properly-connected** if and only if there is a properly-coloured walk from every vertex to every other.



Not properly-connected.

$$f \rightarrow e \rightarrow g$$

$$g \rightarrow e \rightarrow c \rightarrow b$$

$$f \rightarrow e \rightarrow d \rightarrow c \rightarrow b$$

$$a \rightarrow c \rightarrow e \rightarrow d \rightarrow c \rightarrow b$$

No properly-coloured walk  
between  $g$  and  $d$ .

# Proper connection number

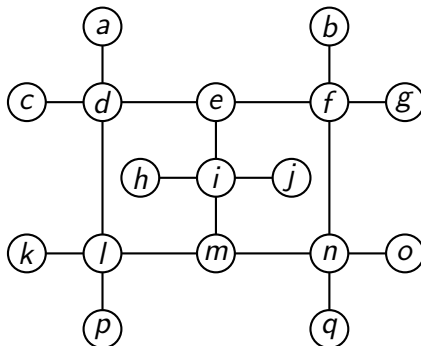
Let  $G = (V, E)$  be a connected graph. A  $k$ -edge-colouring is **connecting** iff  $G_c = (V, E, c)$  is properly-connected.

**Proper-walk connection number : smallest  $k$  such that there exists a connecting  $k$ -edge-colouring of  $G$  ?**

# Proper connection number

Let  $G = (V, E)$  be a connected graph. A  $k$ -edge-colouring is **connecting** iff  $G_c = (V, E, c)$  is properly-connected.

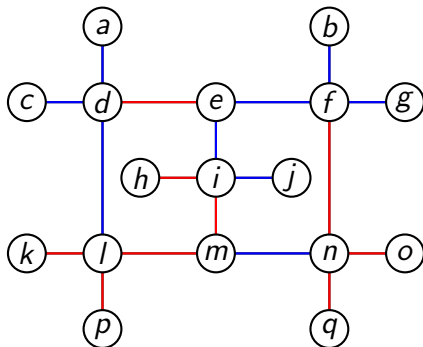
**Proper-walk connection number** : smallest  $k$  such that there exists a connecting  $k$ -edge-colouring of  $G$  ?



# Proper connection number

Let  $G = (V, E)$  be a connected graph. A  $k$ -edge-colouring is **connecting** iff  $G_c = (V, E, c)$  is properly-connected.

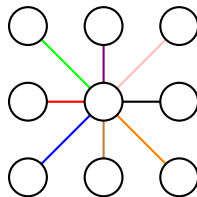
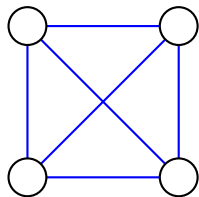
**Proper-walk connection number** : smallest  $k$  such that there exists a connecting  $k$ -edge-colouring of  $G$  ?





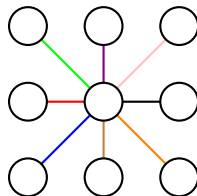
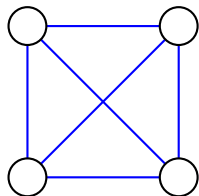
# Easy bounds

- We need between 1 and  $n - 1$  colours.



# Easy bounds

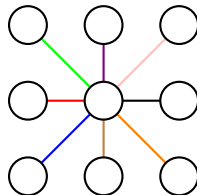
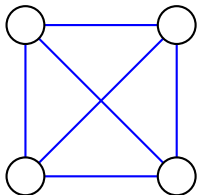
- We need between 1 and  $n - 1$  colours.



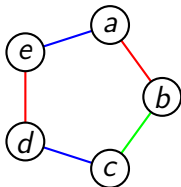
- The **chromatic index** of the graph is always enough.

# Easy bounds

- We need between 1 and  $n - 1$  colours.

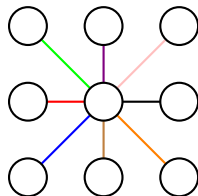
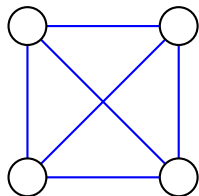


- The **chromatic index** of the graph is always enough. But often unnecessary!

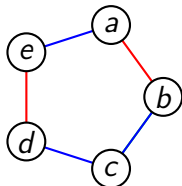


# Easy bounds

- We need between 1 and  $n - 1$  colours.

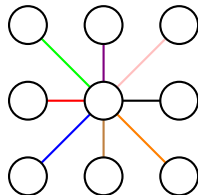
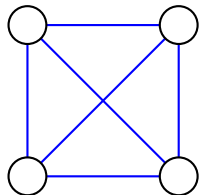


- The **chromatic index** of the graph is always enough. But often unnecessary!

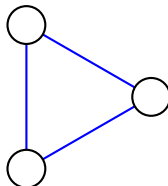
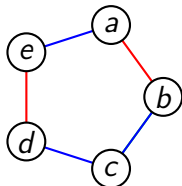


# Easy bounds

- We need between 1 and  $n - 1$  colours.



- The **chromatic index** of the graph is always enough. But often unnecessary!

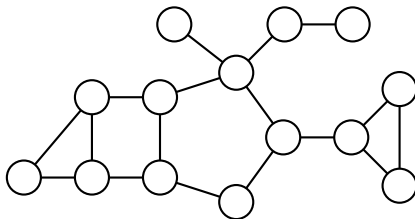


# Connecting 3-edge-colouring

- If  $G$  is a tree, we need  $\Delta(G)$  colours.

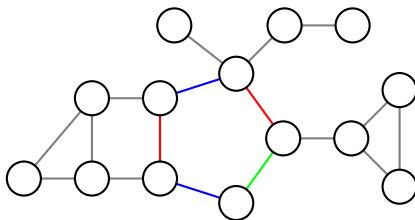
# Connecting 3-edge-colouring

- If  $G$  is a tree, we need  $\Delta(G)$  colours.
- If  $G$  has a cycle :



# Connecting 3-edge-colouring

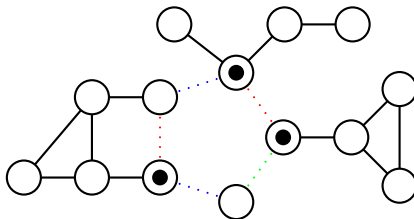
- If  $G$  is a tree, we need  $\Delta(G)$  colours.
- If  $G$  has a cycle :





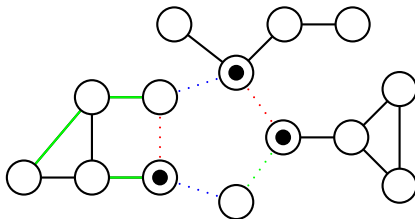
# Connecting 3-edge-colouring

- If  $G$  is a tree, we need  $\Delta(G)$  colours.
- If  $G$  has a cycle :



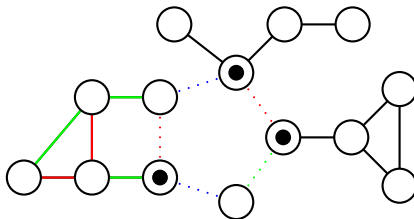
# Connecting 3-edge-colouring

- If  $G$  is a tree, we need  $\Delta(G)$  colours.
- If  $G$  has a cycle :



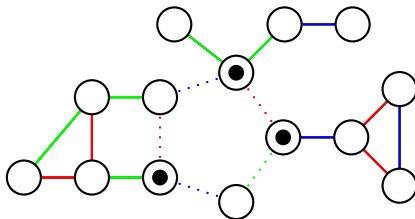
# Connecting 3-edge-colouring

- If  $G$  is a tree, we need  $\Delta(G)$  colours.
- If  $G$  has a cycle :



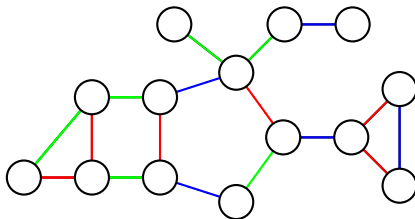
# Connecting 3-edge-colouring

- If  $G$  is a tree, we need  $\Delta(G)$  colours.
- If  $G$  has a cycle :



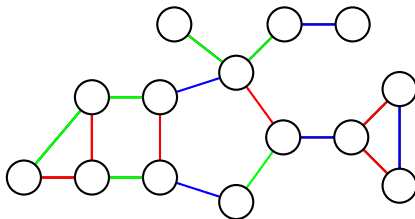
# Connecting 3-edge-colouring

- If  $G$  is a tree, we need  $\Delta(G)$  colours.
- If  $G$  has a cycle : 3 colours are always enough !



# Connecting 3-edge-colouring

- If  $G$  is a tree, we need  $\Delta(G)$  colours.
- If  $G$  has a cycle : 3 colours are always enough !



The complicated part is to determine whether a graph with a cycle requires 3 colors or can be connected with 2.

# Complexity of the problem

## Previous results (Ducoffe, Marinescu-Ghemeci, Popa, 2017)

It is NP-complete to determine if there exists a 2-edge-colouring such that every pair of vertices of a **directed** graph is connected by properly-coloured **elementary paths**.

## Our results (with Jørgen Bang-Jensen and Anders Yeo)

The minimum number of colours required for a connecting edge-colouring of an **undirected** graph  $G$  and an optimal connecting colouring can be found in polynomial time.

- 1 Forbidden-transition graphs
- 2 Edge-coloured graphs
- 3 Parametrized complexity**
  - Parametrized complexity
  - Our results (with S.Li, K.Okrasa, M.Pilipczuk and M.Sorge)
- 4 Exploring temporal graphs



# Context

Let  $G$  be a graph :

What is the chromatic number of  $G$  ?

the size of a maximum clique ?

the size of a minimum dominating set ?

the size of a minimum *vertex cover set* ?

# Context

Let  $G$  be a graph :

What is the chromatic number of  $G$  ?

the size of a maximum clique ?

the size of a minimum dominating set ?

the size of a minimum *vertex cover set* ?

NP-complete

# Context

**Parametrized version** : let  $k$  be a fixed integer

Let  $G$  be a graph :

What is the chromatic number of  $G$  ?

the size of a maximum clique ?

the size of a minimum dominating set ?

the size of a minimum *vertex cover set* ?

# Context

**Parametrized version** : let  $k$  be a fixed integer

Let  $G$  be a graph :

is  $G$   $k$ -colorable ?

is there a clique of size  $k$  ?

a dominating set of size  $k$  ?

a vertex cover set of size  $k$  ?

# Context

**Parametrized version** : let  $k$  be a fixed integer

Let  $G$  be a graph :

is  $G$   $k$ -colorable? **NP-complete**

is there a clique of size  $k$ ?

a dominating set of size  $k$ ?

a vertex cover set of size  $k$ ?

# Context

**Parametrized version** : let  $k$  be a fixed integer

Let  $G$  be a graph :

is  $G$   $k$ -colorable? **NP-complete**

is there a clique of size  $k$ ? **Polynomial**

a dominating set of size  $k$ ? **Polynomial**

a vertex cover set of size  $k$ ? **Polynomial**

# Different ways to be polynomial

Vertex cover of size  $k$  :

- If  $G$  has no edge, **YES**
- If  $k = 0$ , **NO**
- Pick an edge  $uv$  arbitrarily.
  - If  $G \setminus u$  has a vertex cover of size  $k - 1$ , **YES**.
  - If  $G \setminus v$  has a vertex cover of size  $k - 1$ , **YES**.
  - **NO**

# Different ways to be polynomial

Vertex cover of size  $k$  :

- If  $G$  has no edge, **YES**
- If  $k = 0$ , **NO**
- Pick an edge  $uv$  arbitrarily.
  - If  $G \setminus u$  has a vertex cover of size  $k - 1$ , **YES**.
  - If  $G \setminus v$  has a vertex cover of size  $k - 1$ , **YES**.
  - **NO**

Complexity :  $O(2^k \times n^2)$ .



# Different ways to be polynomial

Vertex cover of size  $k$  :

- If  $G$  has no edge, **YES**
- If  $k = 0$ , **NO**
- Pick an edge  $uv$  arbitrarily.
  - If  $G \setminus u$  has a vertex cover of size  $k - 1$ , **YES**.
  - If  $G \setminus v$  has a vertex cover of size  $k - 1$ , **YES**.
  - **NO**

Complexity :  $O(2^k \times n^2)$ .

Brute force algorithm for clique :  $O(n^k \times k^2)$ .

# Different ways to be polynomial

Vertex cover of size  $k$  :

- If  $G$  has no edge, **YES**
- If  $k = 0$ , **NO**
- Pick an edge  $uv$  arbitrarily.
  - If  $G \setminus u$  has a vertex cover of size  $k - 1$ , **YES**.
  - If  $G \setminus v$  has a vertex cover of size  $k - 1$ , **YES**.
  - **NO**

Complexity :  $O(2^k \times n^2)$ .

Brute force algorithm for clique :  $O(n^k \times k^2)$ .

A problem is FPT in  $k$  if we can solve it in time

$O(f(k) \times \text{poly}(n))$ .

# Choosing the parameter

- Parameter related to the solution.
- Any other graph invariant.

## Theorem (Courcelle, 1990)

Any graph property definable in the monadic second-order logic can be decided in linear time on any class of graphs of bounded treewidth.

What makes the problem difficult? On which graph classes can we solve it efficiently?

# Elementary path

Let  $G = (V, E, T)$ ,  $u, v \in V$ . Is there an elementary path between  $u$  and  $v$ ?

- **NP-complete.** (Szeider, 2003)

# Elementary path

Let  $G = (V, E, T)$ ,  $u, v \in V$ . Is there an elementary path between  $u$  and  $v$ ?

- **NP-complete.** (Szeider, 2003)
- **FPT** in the length of the path.  
Is there a path of length at most  $k$ ?

# Elementary path

Let  $G = (V, E, T)$ ,  $u, v \in V$ . Is there an elementary path between  $u$  and  $v$ ?

- **NP-complete.** (Szeider, 2003)
- **FPT** in the length of the path.  
Is there a path of length at most  $k$ ?
- **FPT** in the detour.  
Let  $d$  be the distance between  $u$  and  $v$  in the underlying classical graph. Is there a path of length at most  $d + k$ ?

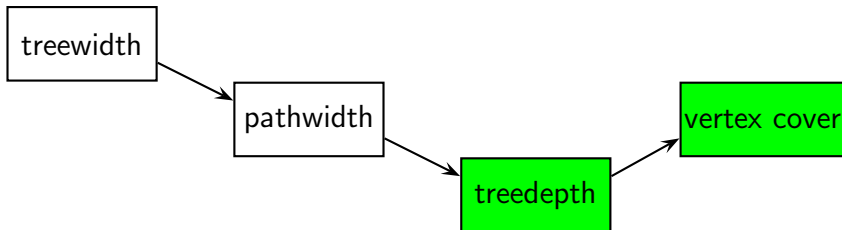
# Other parameters

- FPT in the size of a smallest vertex cover set.
- FPT in the treedepth.

Minimum depth of a spanning forest  $F$  such that every pair of adjacent vertices in  $G$  have an ancestor-descendant relationship in  $F$ .

# Other parameters

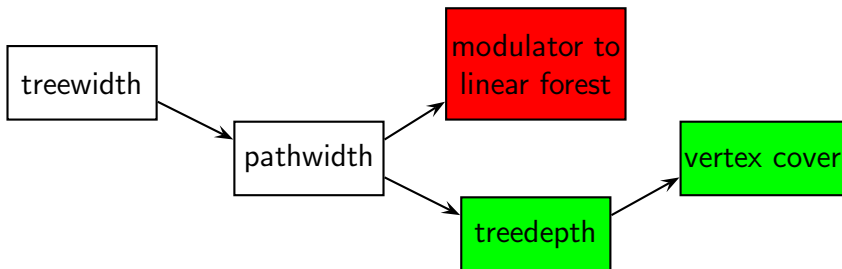
- FPT in the size of a smallest vertex cover set.
- FPT in the treedepth.





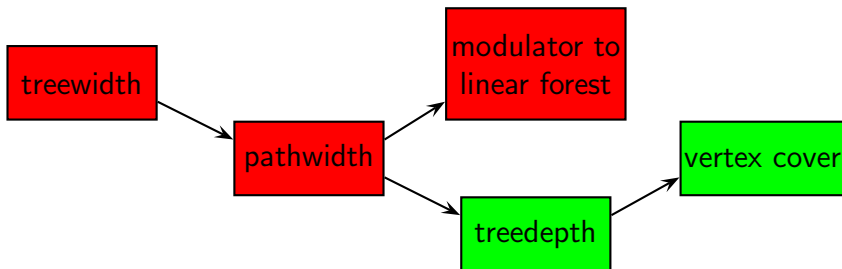
# Other parameters

- FPT in the size of a smallest vertex cover set.
- FPT in the treedepth.



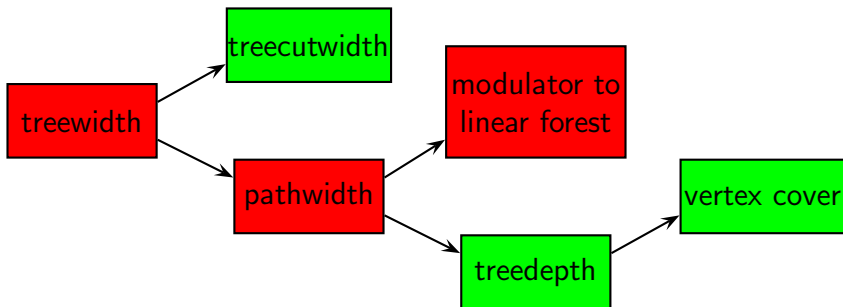
# Other parameters

- FPT in the size of a smallest vertex cover set.
- FPT in the treedepth.



# Other parameters

- FPT in the size of a smallest vertex cover set.
- FPT in the treedepth.



# Other parameters

- **FPT** in the size of a smallest vertex cover set.
- **FPT** in the treedepth.
- **W[1]-hard** in the size of a modulator to a linear forest.
- **W[1]-hard** in the pathwidth.
- **W[1]-hard** in the treewidth.
- **FPT** in the treecutwidth :  

$$k^{\mathcal{O}(k^2)} \cdot n^2 + \mathcal{O}(n^3) + \mathcal{O}((4^k \cdot k!)^{\mathcal{O}(3k+1)}) \cdot n^2.$$
 Analogous of treewidth with edges instead of vertices  
 (highlights small edge cuts in the graph).

# Hamiltonian cycles

Compatible Hamiltonian cycles and paths in forbidden-transition graphs :

- **W[1]-hard** in the size of a modulator to treewidth 2.

# Hamiltonian cycles

Compatible Hamiltonian cycles and paths in forbidden-transition graphs :

- **W[1]-hard** in the size of a modulator to treewidth 2.
- **W[1]-hard** in the pathwidth
- **W[1]-hard** in the treewidth.

# Hamiltonian cycles

Compatible Hamiltonian cycles and paths in forbidden-transition graphs :

- **W[1]-hard** in the size of a modulator to treewidth 2.
- **W[1]-hard** in the pathwidth
- **W[1]-hard** in the treewidth.

Properly-coloured Hamiltonian cycles :

- **FPT** in the treewidth.  
 $2^{\mathcal{O}(k)} \cdot (|V(G)| + |V(\mathcal{T})| + \ell)$   
 where  $k$  is the treewidth,  $\mathcal{T}$  is the tree of the decomposition and  $\ell$  is the number of colours of the edge-colouring.

- 1 Forbidden-transition graphs
- 2 Edge-coloured graphs
- 3 Parametrized complexity
- 4 Exploring temporal graphs
  - Definitions
  - Restlessness
  - Our results (with Cyril Conchon-Kerjan and Bruno Escoffier)



# Our model

## Temporal graph

A *temporal graph* is defined by

- a set of vertices  $V$
- a sequence of set of edges  $E_1, E_2, \dots$  that may or may not be finite

The graph  $G_t = (V, E_t)$  is called the *snapshot* at time  $t$ .

# Our model

## Temporal graph

A *temporal graph* is defined by

- a set of vertices  $V$
- a sequence of set of edges  $E_1, E_2, \dots$  that may or may not be finite

The graph  $G_t = (V, E_t)$  is called the *snapshot* at time  $t$ .

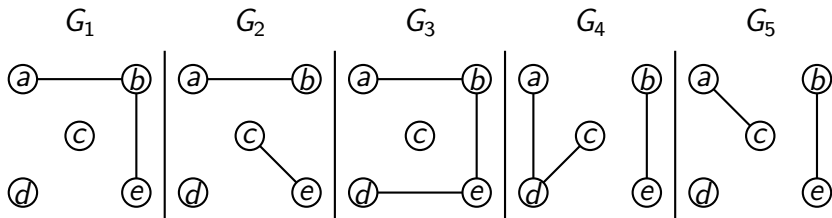
## Journey and exploration

A *journey* in a temporal graph is a sequence of vertices

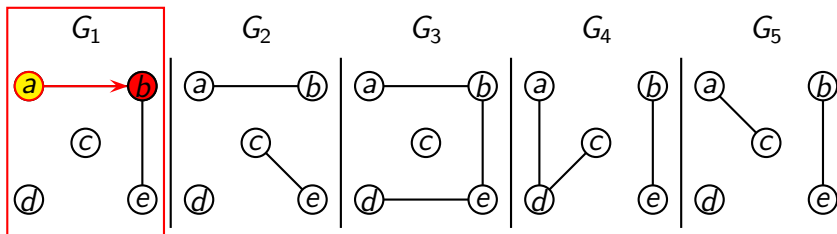
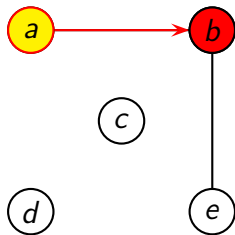
$v_0, v_1, \dots$  such that for all  $i$ ,  $v_i = v_{i+1}$  or  $v_i v_{i+1} \in E_i$ .

Given a temporal graph, the problem of *graph exploration* is to find a journey that goes through all the vertices

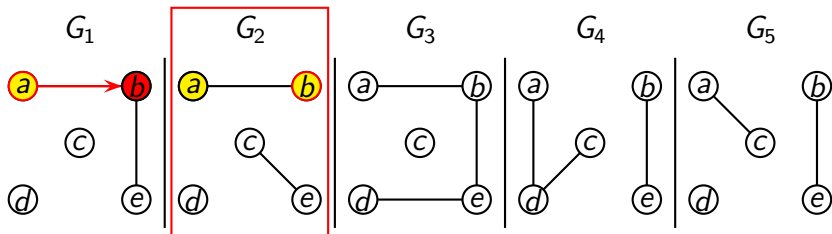
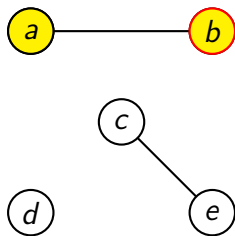
# An example



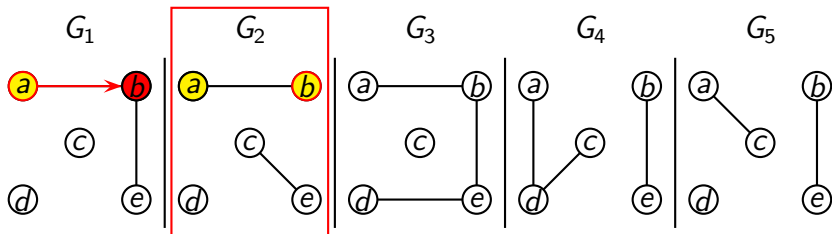
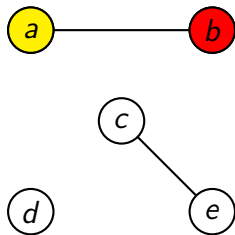
# An example



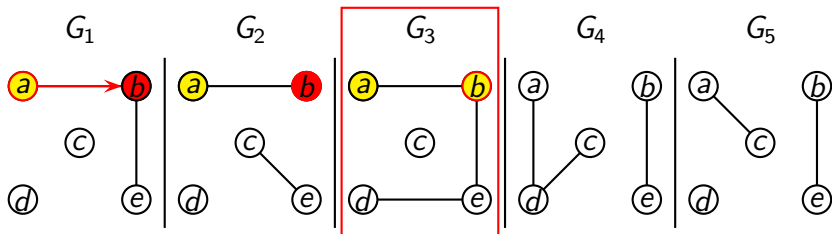
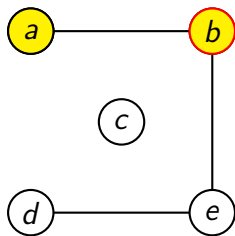
# An example



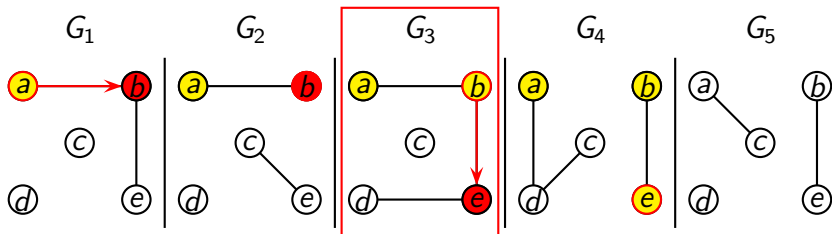
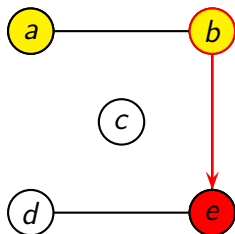
# An example



# An example

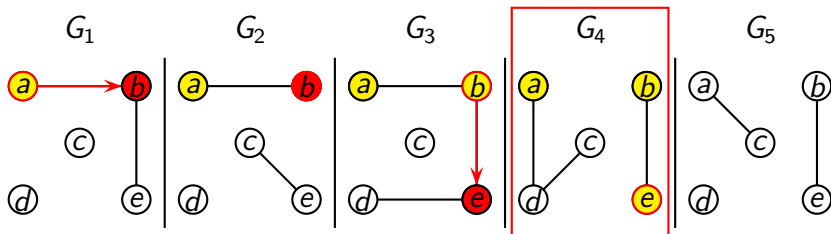
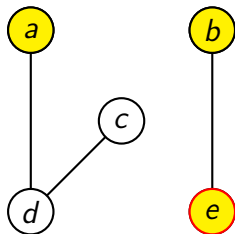


# An example

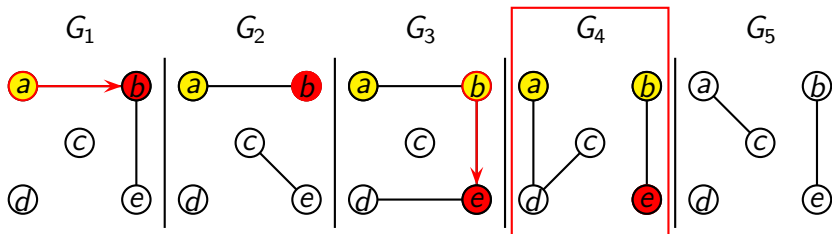
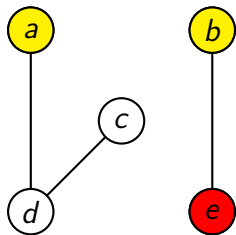




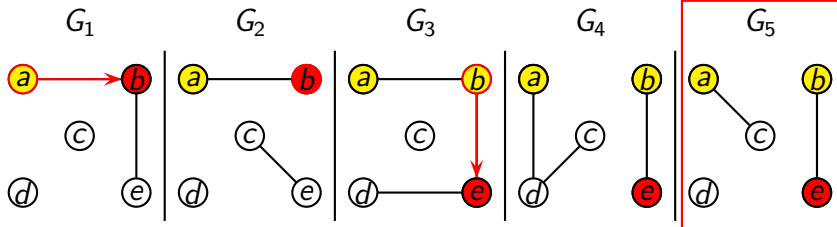
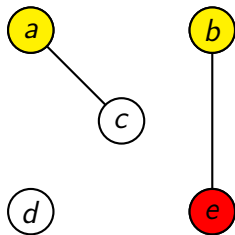
# An example



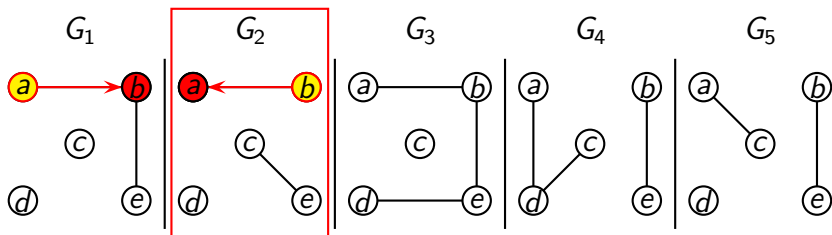
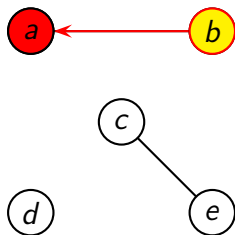
# An example



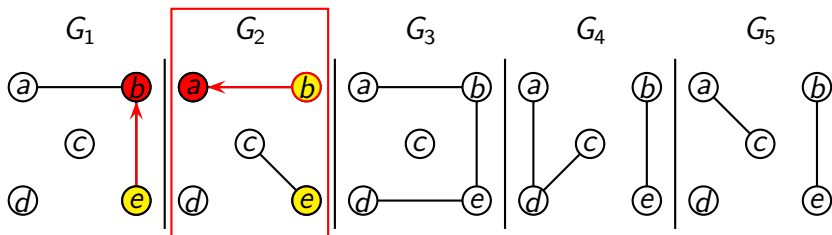
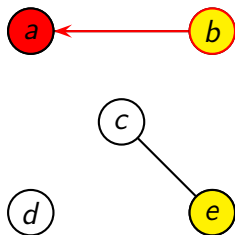
# An example



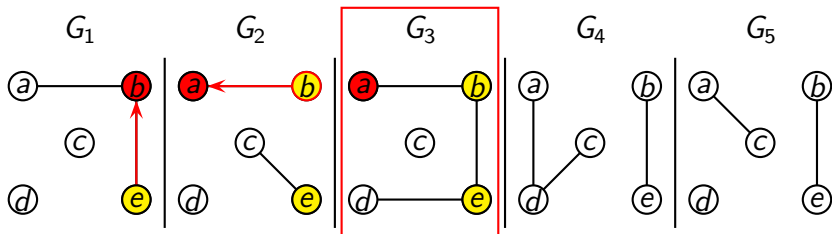
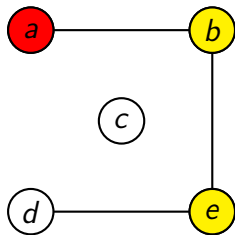
# An example



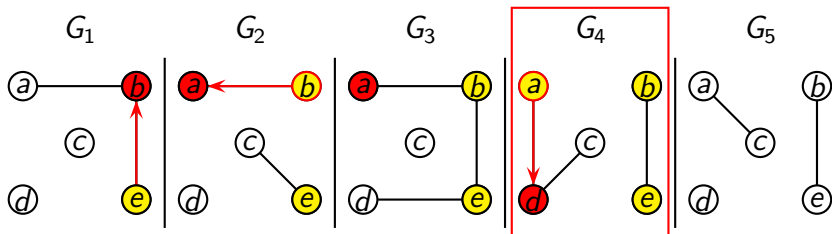
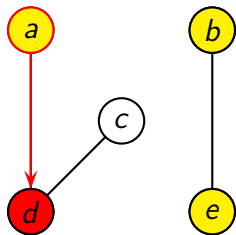
# An example



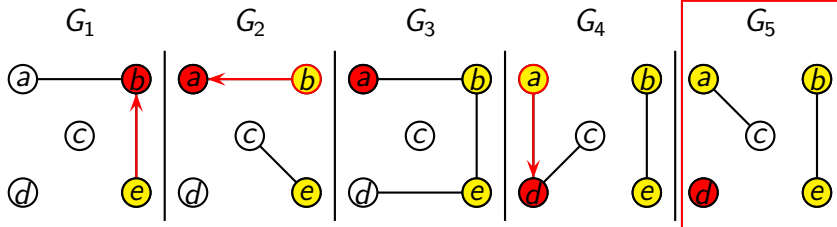
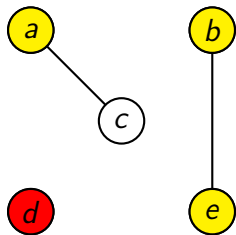
# An example



# An example

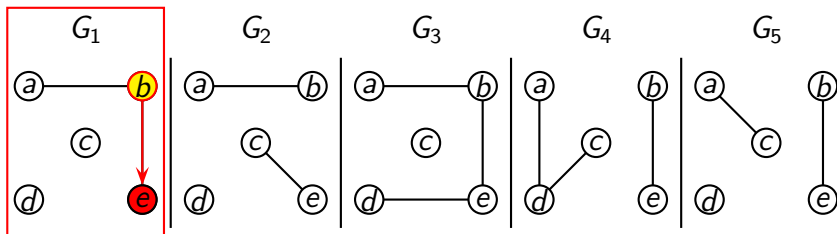
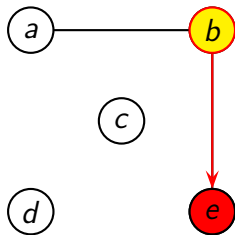


# An example

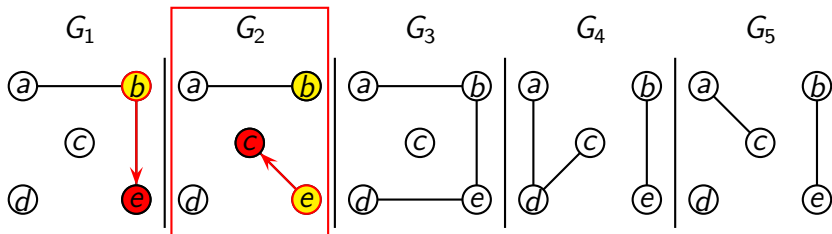
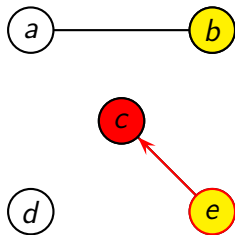




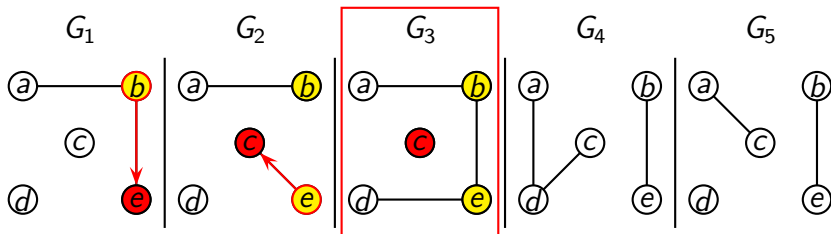
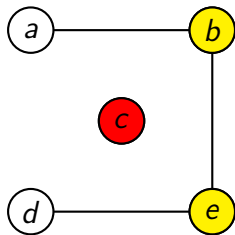
# An example



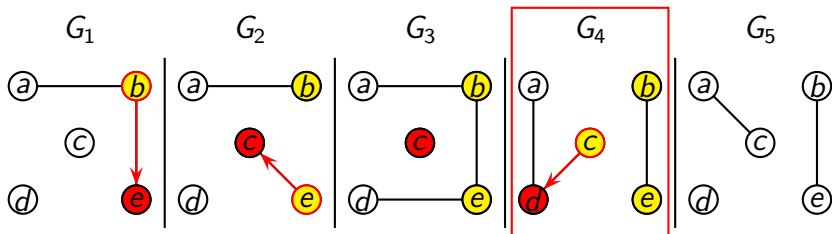
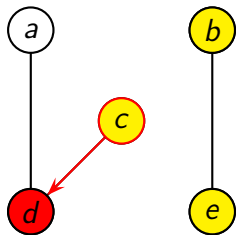
# An example



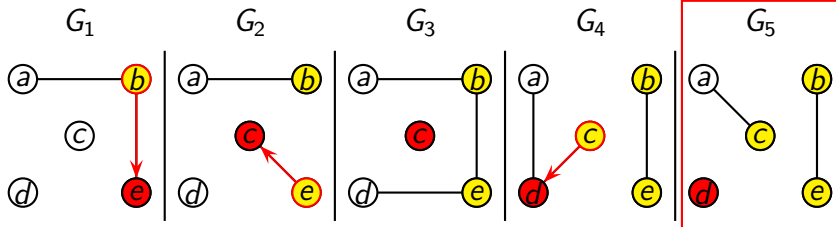
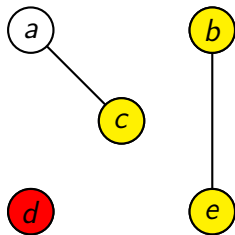
# An example



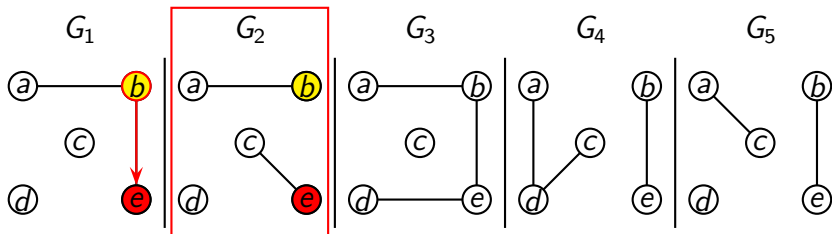
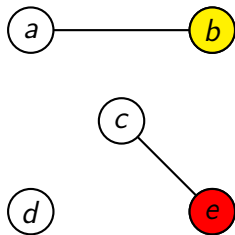
# An example



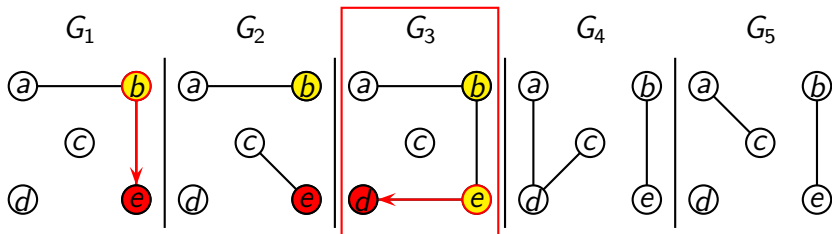
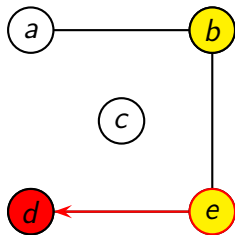
# An example



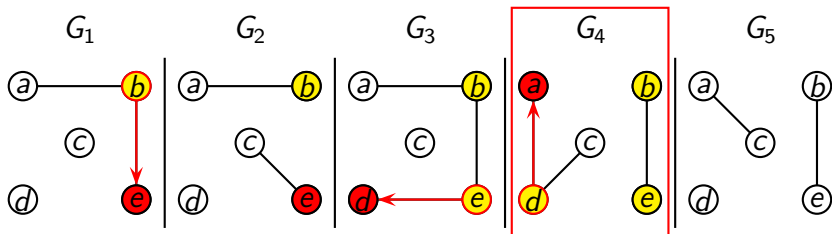
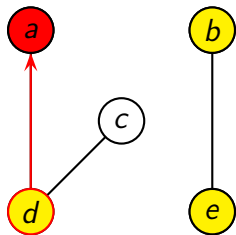
# An example



# An example

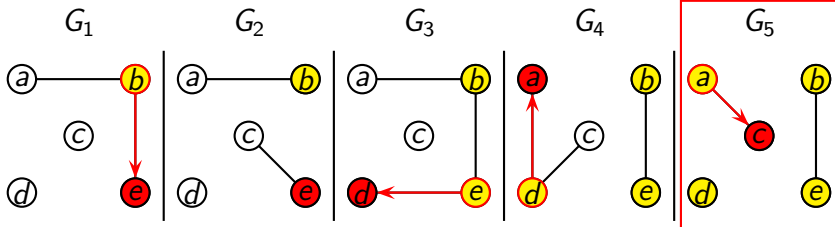
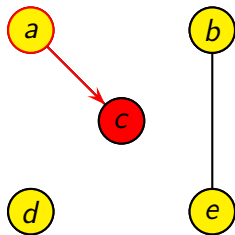


# An example

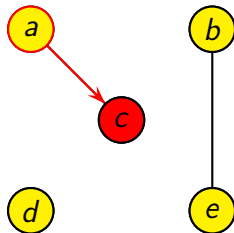




# An example



# An example



NP-complete in the lifespan of the graph !

# Time bounds

How much time can it take to explore a graph ?

- If every  $G_i$  is connected  $O(n^2)$ .

# Time bounds

How much time can it take to explore a graph ?

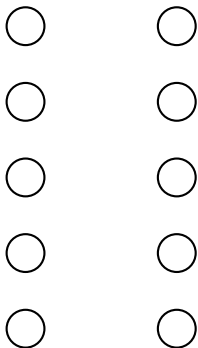
- If every  $G_i$  is connected  $O(n^2)$ .
- In the general case, it might never be possible.

# Time bounds

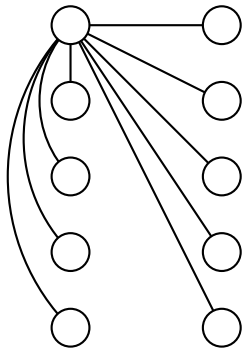
## How much time can it take to explore a graph ?

- If every  $G_i$  is connected  $O(n^2)$ .
- In the general case, it might never be possible.
- If the graph is periodic, possible iff the footprint is connected.  $O(pn)$

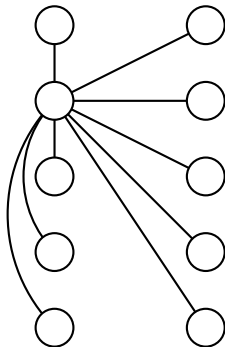
# Erlebach, Hoffmann and Kammer's construction



# Erlebach, Hoffmann and Kammer's construction

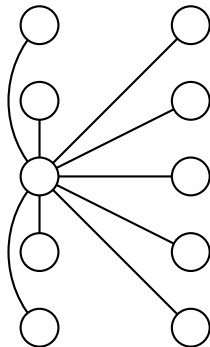


# Erlebach, Hoffmann and Kammer's construction

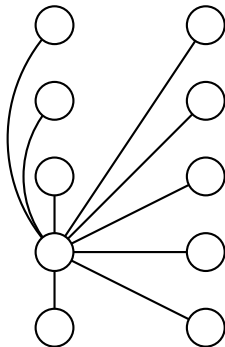




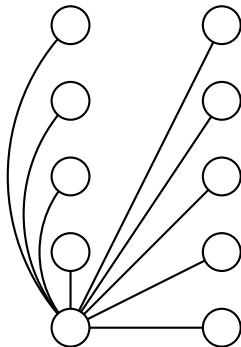
# Erlebach, Hoffmann and Kammer's construction



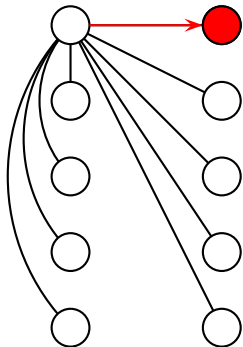
# Erlebach, Hoffmann and Kammer's construction



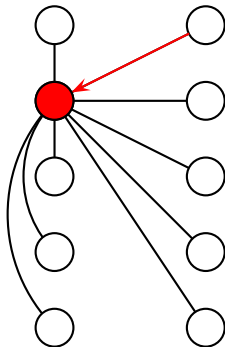
# Erlebach, Hoffmann and Kammer's construction



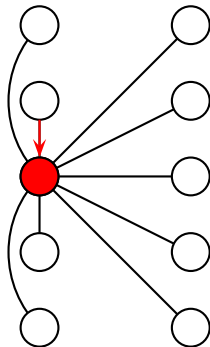
# Erlebach, Hoffmann and Kammer's construction



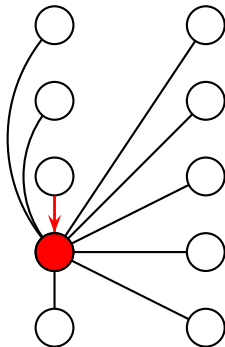
# Erlebach, Hoffmann and Kammer's construction



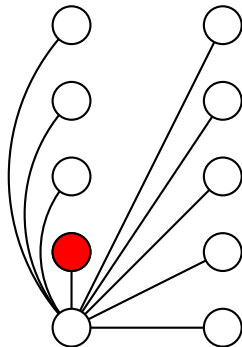
# Erlebach, Hoffmann and Kammer's construction



# Erlebach, Hoffmann and Kammer's construction

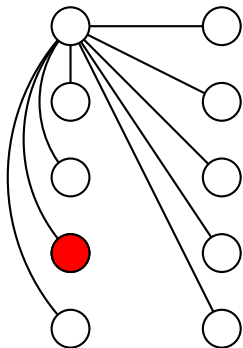


# Erlebach, Hoffmann and Kammer's construction

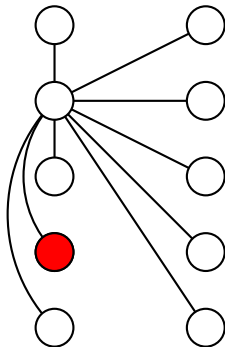




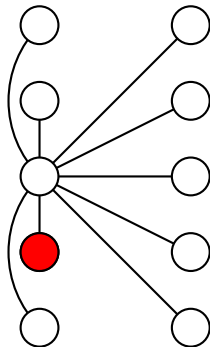
# Erlebach, Hoffmann and Kammer's construction



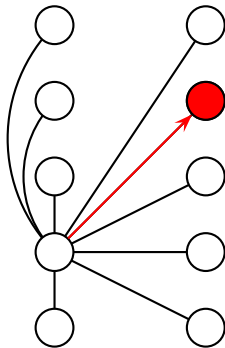
# Erlebach, Hoffmann and Kammer's construction



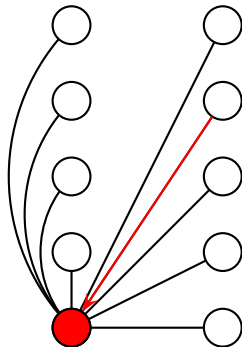
# Erlebach, Hoffmann and Kammer's construction



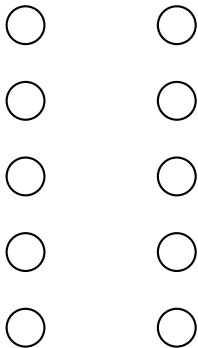
# Erlebach, Hoffmann and Kammer's construction



# Erlebach, Hoffmann and Kammer's construction



# Erlebach, Hoffmann and Kammer's construction



The  $O(n^2)$ -bound is tight

# Restlessness

## Definition

A journey is  $\Delta$ -restless is not allowed to wait  $\Delta$  steps in a row.  
For example :

- routing in non delay tolerant network, with memory limitation on the nodes ;
- the spread of the virus if patients recover in  $\Delta$  steps.

# Restlessness

## Definition

A journey is  $\Delta$ -restless is not allowed to wait  $\Delta$  steps in a row.  
For example :

- routing in non delay tolerant network, with memory limitation on the nodes ;
- the spread of the virus if patients recover in  $\Delta$  steps.

Restless exploration is not always possible even if every snapshot is connected (cf previous construction).



# Restlessness

## Definition

A journey is  $\Delta$ -restless is not allowed to wait  $\Delta$  steps in a row.  
For example :

- routing in non delay tolerant network, with memory limitation on the nodes ;
- the spread of the virus if patients recover in  $\Delta$  steps.

Restless exploration is not always possible even if every snapshot is connected (cf previous construction).

## Our problem

Given a periodic temporal graph, can we explore it restlessly ?

# Edge-coloring and temporality

## Strict alternance

In a  $k$ -edge-colored graph, a walk is *strictly alternating* if the edge between the vertices  $i$  and  $i + 1$  is colored  $i \bmod k$ .

# Edge-coloring and temporality

## Strict alternance

In a  $k$ -edge-colored graph, a walk is *strictly alternating* if the edge between the vertices  $i$  and  $i + 1$  is colored  $i \bmod k$ .

Let  $G_t = (V, (E_1, E_2, \dots, E_k))$  be a  $k$ -periodic temporal graph.

Let  $G_c$  be a  $k$ -edge-colored multigraph with :

- same vertex set as  $G_t$
- an edge  $uv$  colored  $i$  for every  $uv \in E_i$

# Edge-coloring and temporality

## Strict alternance

In a  $k$ -edge-colored graph, a walk is *strictly alternating* if the edge between the vertices  $i$  and  $i + 1$  is colored  $i \bmod k$ .

Let  $G_t = (V, (E_1, E_2, \dots, E_k))$  be a  $k$ -periodic temporal graph.

Let  $G_c$  be a  $k$ -edge-colored multigraph with :

- same vertex set as  $G_t$
- an edge  $uv$  colored  $i$  for every  $uv \in E_i$

- $k$ -restless journey in  $G_t =$  properly-colored walk in  $G_c$ .
- 1-restless journey in  $G_t =$  strictly alternating walk in  $G_c$ .

# Our results

## Theorem

One can decide in polynomial time whether a 2-periodic temporal graph can be 1-restlessly explored.

# Our results

## Theorem

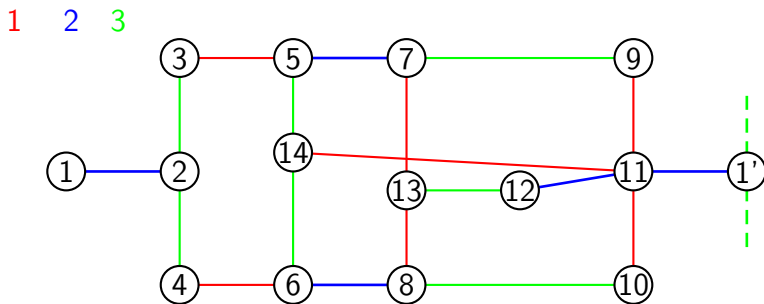
One can decide in polynomial time whether a 2-periodic temporal graph can be 1-restlessly explored.

## Theorem

For every  $p \geq 3$ , it is NP-complete to decide whether a  $p$ -periodic temporal graph can be 1-restlessly explored.

Our results (with Cyril Conchon-Kerjan and Bruno Escoffier)

# Idea of the proof

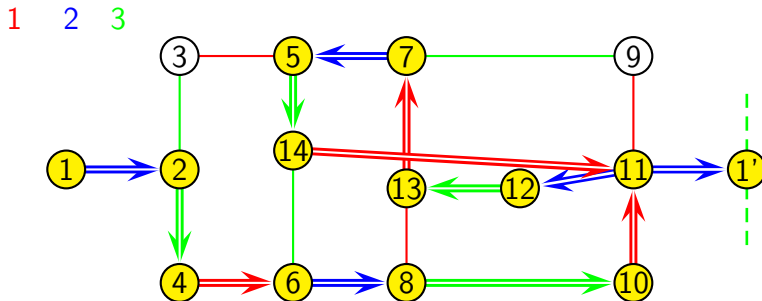






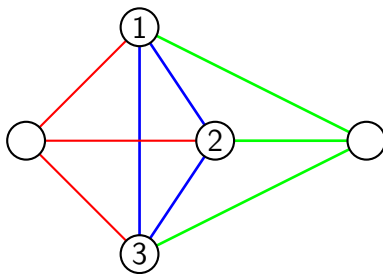
Our results (with Cyril Conchon-Kerjan and Bruno Escoffier)

# Idea of the proof



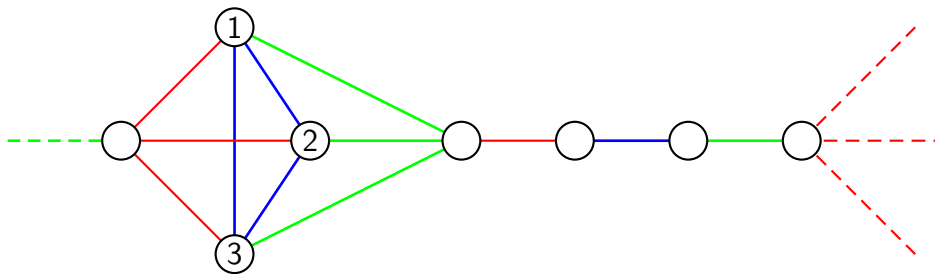
Our results (with Cyril Conchon-Kerjan and Bruno Escoffier)

# Idea of the proof

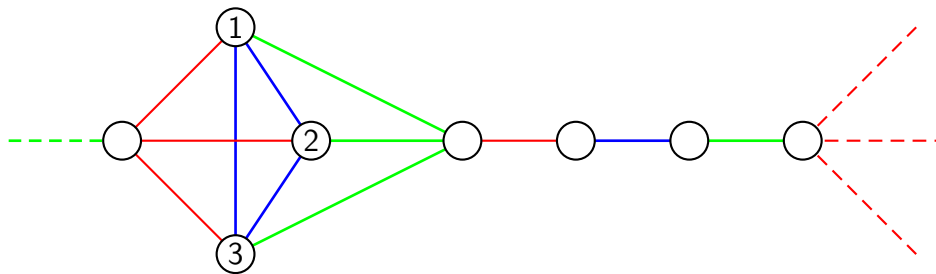


Our results (with Cyril Conchon-Kerjan and Bruno Escoffier)

# Idea of the proof



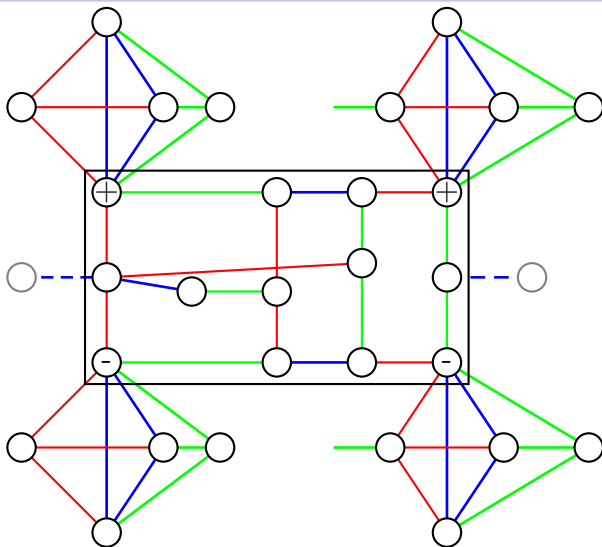
# Idea of the proof



3-SAT-(2,2) : 3-sat where every variable has exactly 2 positive and 2 negative occurrences. NP-complete (Berman, Karpinski, Scott)

Our results (with Cyril Conchon-Kerjan and Bruno Escoffier)

# Idea of the proof

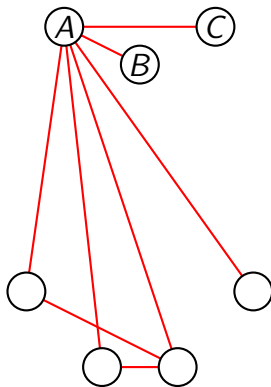


# The trap

Our results still hold if the graph has to be connected all the time.

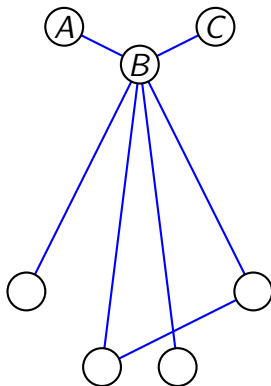
# The trap

Our results still hold if the graph has to be connected all the time.



# The trap

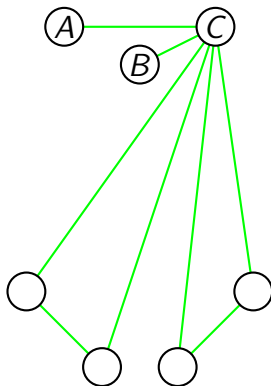
Our results still hold if the graph has to be connected all the time.





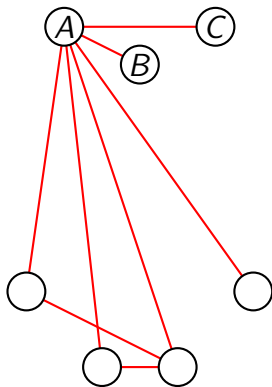
# The trap

Our results still hold if the graph has to be connected all the time.



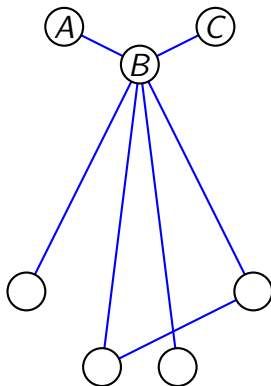
# The trap

Our results still hold if the graph has to be connected all the time.



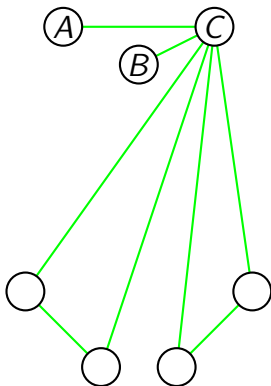
# The trap

Our results still hold if the graph has to be connected all the time.



# The trap

Our results still hold if the graph has to be connected all the time.



# Time bounds

## Upper bound

Any restlessly-explorable  $p$ -periodic temporal graph can be explored restlessly in at most  $pn^2$  steps.

## Lower bound

For every  $p \geq 2$  there are families of restlessly-explorable  $p$ -periodic temporal graphs that require  $\Omega(pn^2)$  steps to be explored restlessly.



# Open questions

- Every explorable  $p$ -periodic graph can be explored in time  $pn^2$  and some require  $\frac{pn^2}{18}$ . Can we close the gap?

# Open questions

- Every explorable  $p$ -periodic graph can be explored in time  $pn^2$  and some require  $\frac{pn^2}{18}$ . Can we close the gap?
- What about  $k$ -restlessness with  $k > 1$ ?



# Open questions

- Every explorable  $p$ -periodic graph can be explored in time  $pn^2$  and some require  $\frac{pn^2}{18}$ . Can we close the gap?
- What about  $k$ -restlessness with  $k > 1$ ?
- FPT algorithm for the NP-complete case?

# Open questions

- Every explorable  $p$ -periodic graph can be explored in time  $pn^2$  and some require  $\frac{pn^2}{18}$ . Can we close the gap?
- What about  $k$ -restlessness with  $k > 1$ ?
- FPT algorithm for the NP-complete case?
- What if the graph cannot change too much between two consecutive frames?

# Open questions

- Every explorable  $p$ -periodic graph can be explored in time  $pn^2$  and some require  $\frac{pn^2}{18}$ . Can we close the gap?
- What about  $k$ -restlessness with  $k > 1$ ?
- FPT algorithm for the NP-complete case?
- What if the graph cannot change too much between two consecutive frames?
- We could study criteria stronger than connectivity or explorability.

Thank you !