Sliding Window Temporal Graph Coloring

George B. Mertzios[1] Hendrik Molter[2] Viktor Zamaraev[1]

[1]Department of Computer Science, Durham University, Durham, UK
[2]Algorithmics and Computational Complexity, TU Berlin, Germany

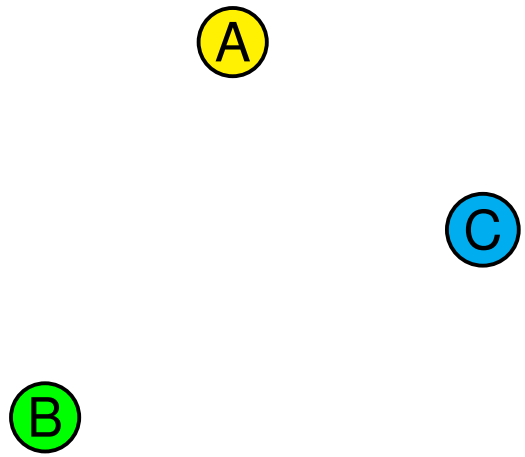
AAAI 2019, Honolulu

This is a preliminary (unfinished) version. Subject to updates.

Introduction
Motivation

Motivating Scenario:

- Mobile agents broadcast information
- When agents meet they can exchange information
- Information can only be exchanged if agents broadcast on **different** channels
- Agents should be able to exchange information within reasonable time windows around their meetings

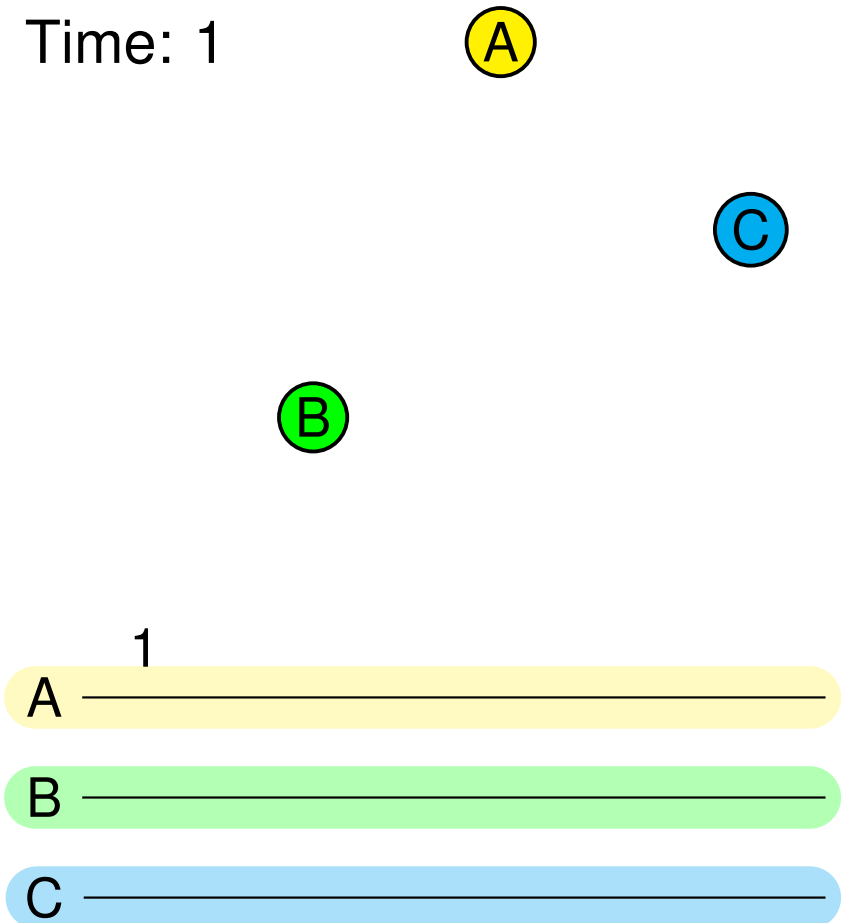


“Channel Assignment Problem”

Motivating Scenario:

- Mobile agents broadcast information
- When agents meet they can exchange information
- Information can only be exchanged if agents broadcast on **different** channels
- Agents should be able to exchange information within reasonable time windows around their meetings

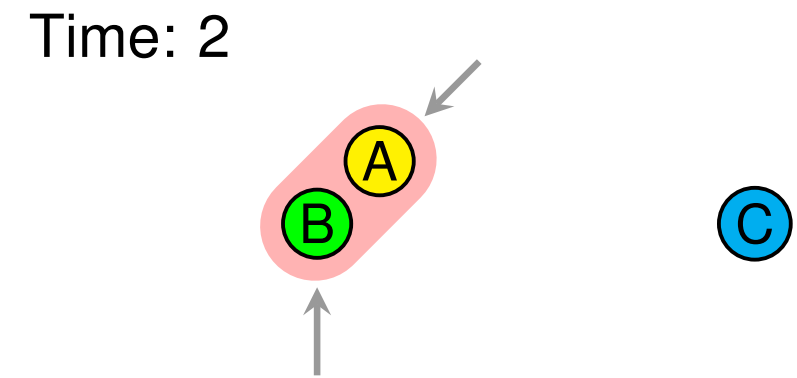
“Channel Assignment Problem”



Motivating Scenario:

- Mobile agents broadcast information
- When agents meet they can exchange information
- Information can only be exchanged if agents broadcast on **different** channels
- Agents should be able to exchange information within reasonable time windows around their meetings

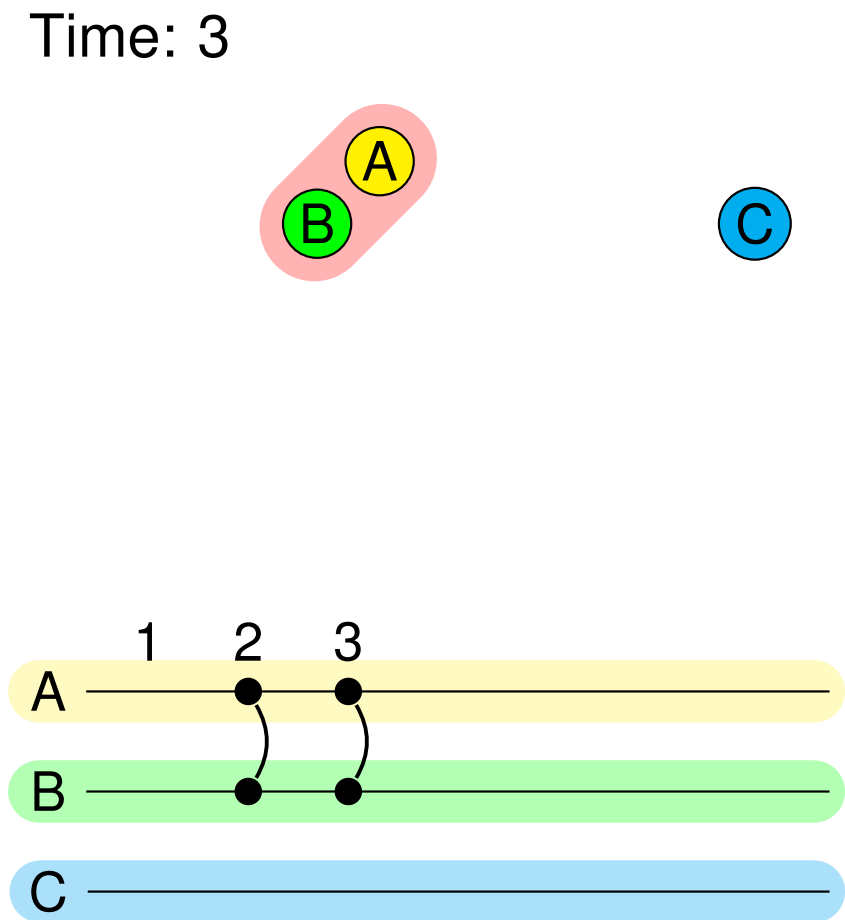
“Channel Assignment Problem”



Motivating Scenario:

- Mobile agents broadcast information
- When agents meet they can exchange information
- Information can only be exchanged if agents broadcast on **different** channels
- Agents should be able to exchange information within reasonable time windows around their meetings

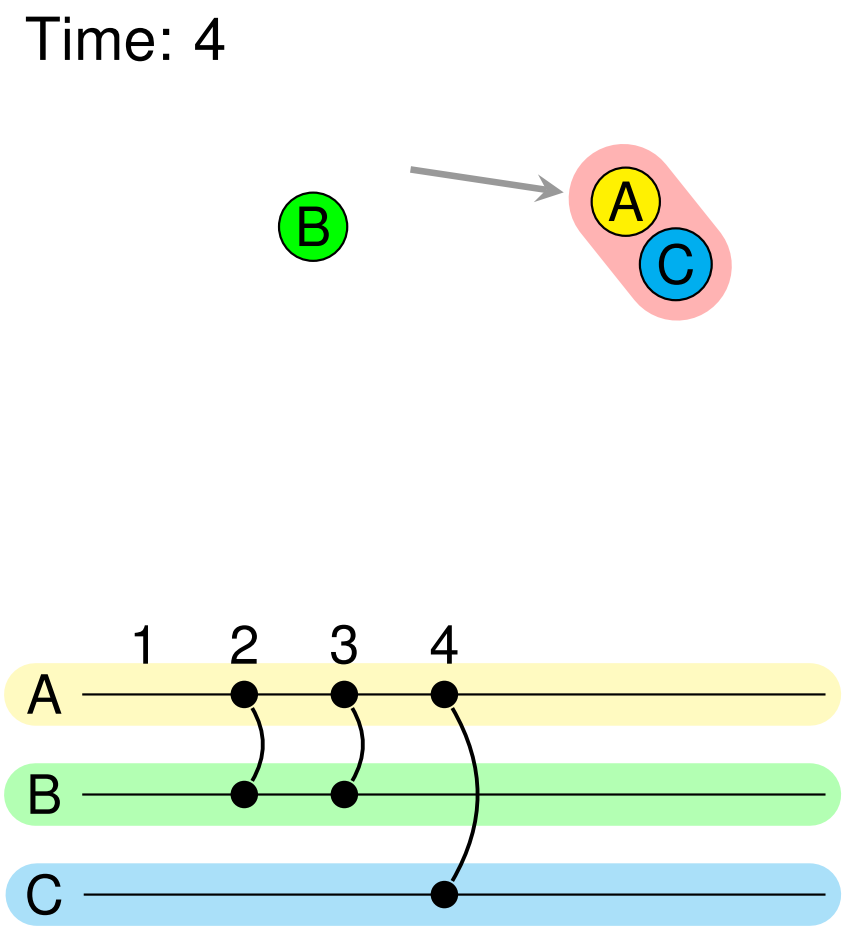
“Channel Assignment Problem”



Motivating Scenario:

- Mobile agents broadcast information
- When agents meet they can exchange information
- Information can only be exchanged if agents broadcast on **different** channels
- Agents should be able to exchange information within reasonable time windows around their meetings

“Channel Assignment Problem”

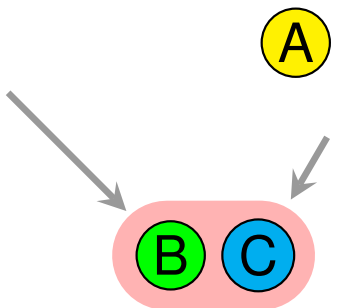


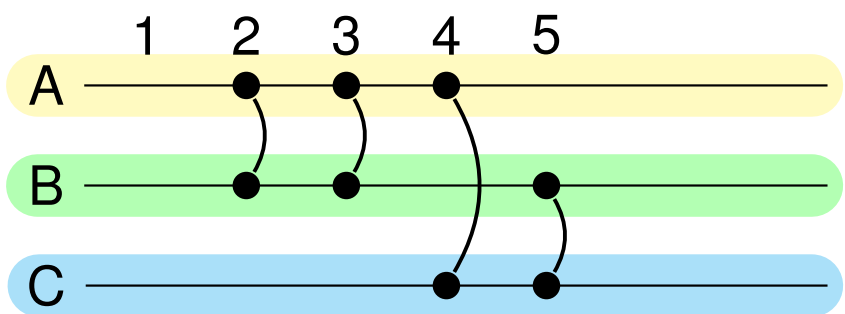
Motivating Scenario:

- Mobile agents broadcast information
- When agents meet they can exchange information
- Information can only be exchanged if agents broadcast on **different** channels
- Agents should be able to exchange information within reasonable time windows around their meetings

“Channel Assignment Problem”

Time: 5

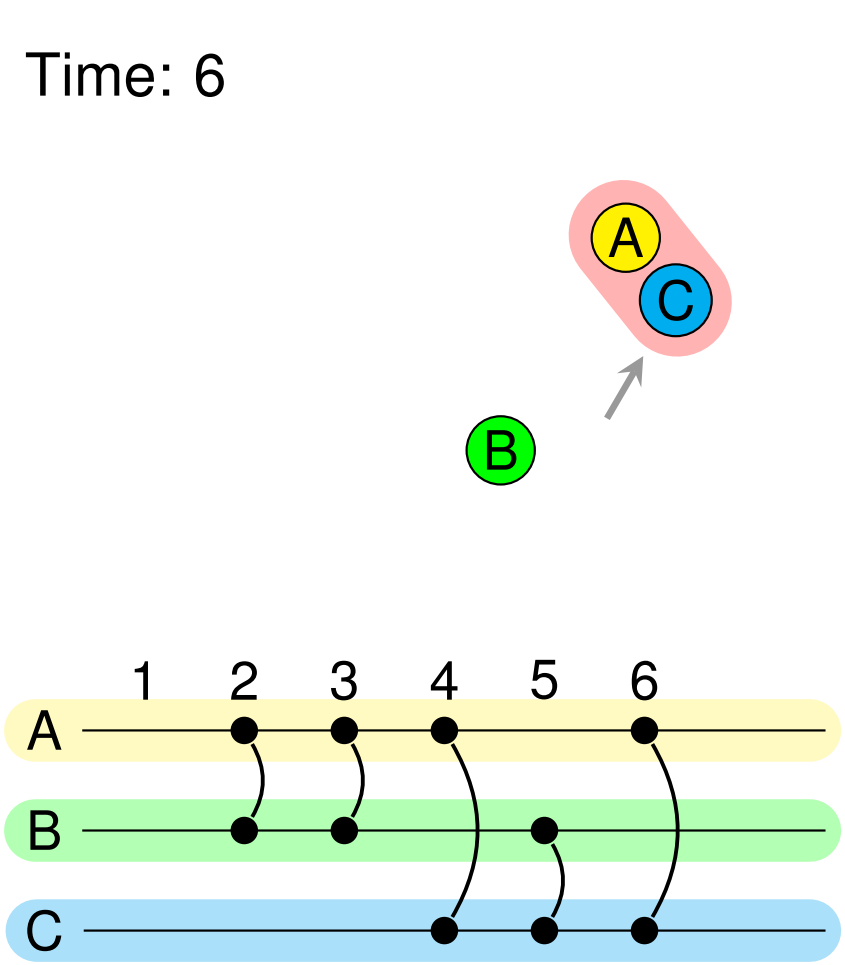




Motivating Scenario:

- Mobile agents broadcast information
- When agents meet they can exchange information
- Information can only be exchanged if agents broadcast on **different** channels
- Agents should be able to exchange information within reasonable time windows around their meetings

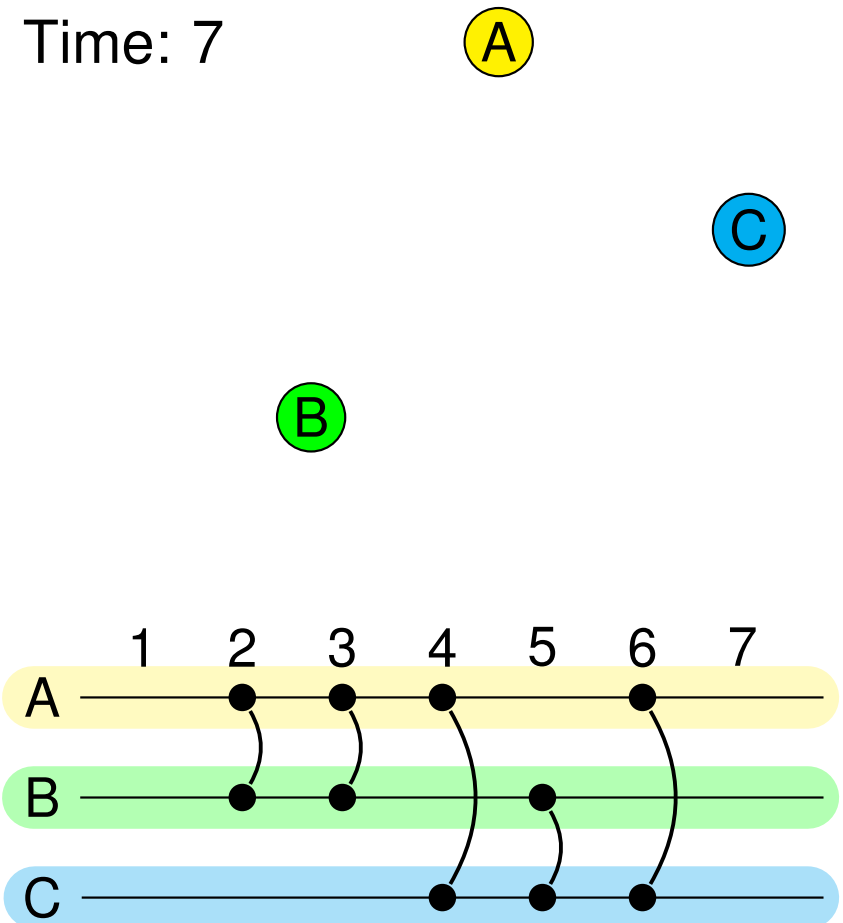
“Channel Assignment Problem”



Motivating Scenario:

- Mobile agents broadcast information
- When agents meet they can exchange information
- Information can only be exchanged if agents broadcast on **different** channels
- Agents should be able to exchange information within reasonable time windows around their meetings

“Channel Assignment Problem”



Introduction
Motivation II

- Channel Assignment Problems are often modeled as **graph coloring problems**
- Movement of agents / changes over time are modeled as a **temporal graph**
- Naturally leads to a temporal graph coloring problem
- Time windows around meetings of agents $\rightarrow$ “sliding windows”

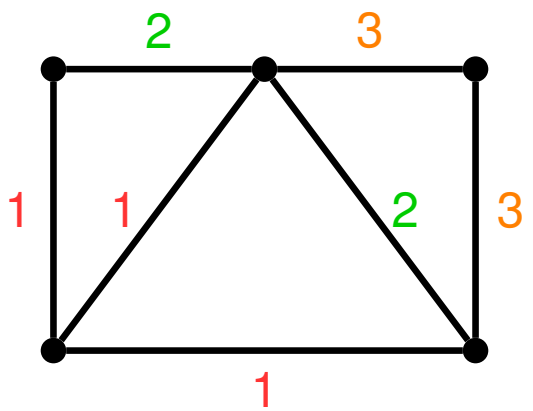
Temporal Graph

A **temporal graph** $(G = (V, E), \lambda)$ is defined as a graph $G = (V, E)$ with a labeling function $\lambda : E \to 2^{\mathbb{N}}$ that assigns time labels to edges.

Temporal Graph

A **temporal graph** $(G = (V, E), \lambda)$ is defined as a graph $G = (V, E)$ with a labeling function $\lambda : E \to 2^{\mathbb{N}}$ that assigns time labels to edges.

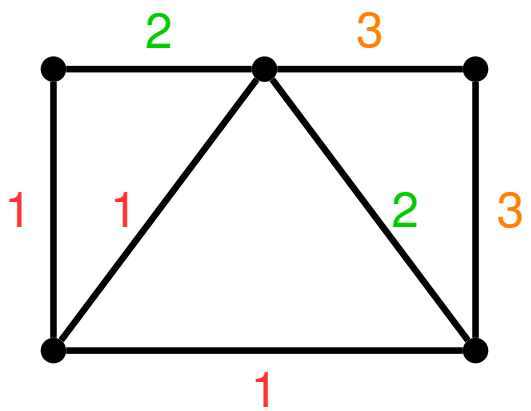
(G, λ):



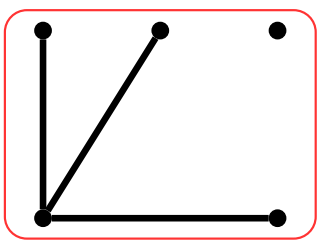
Temporal Graph

A **temporal graph** $(G = (V, E), \lambda)$ is defined as a graph $G = (V, E)$ with a labeling function $\lambda : E \to 2^{\mathbb{N}}$ that assigns time labels to edges.

(G, λ):

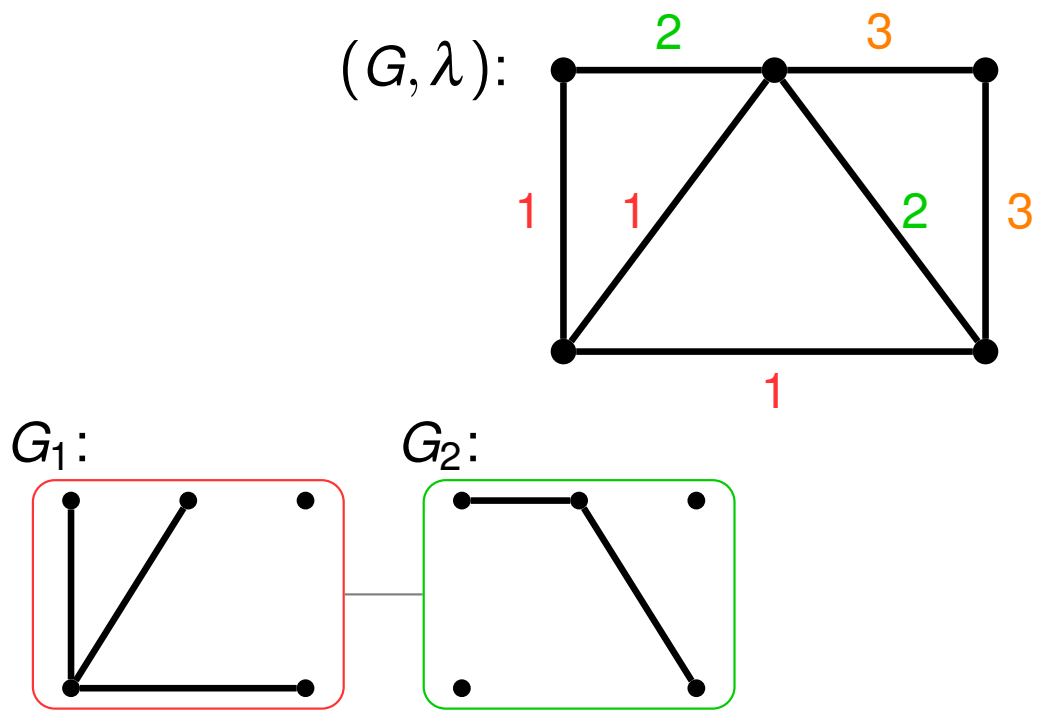


G_1:



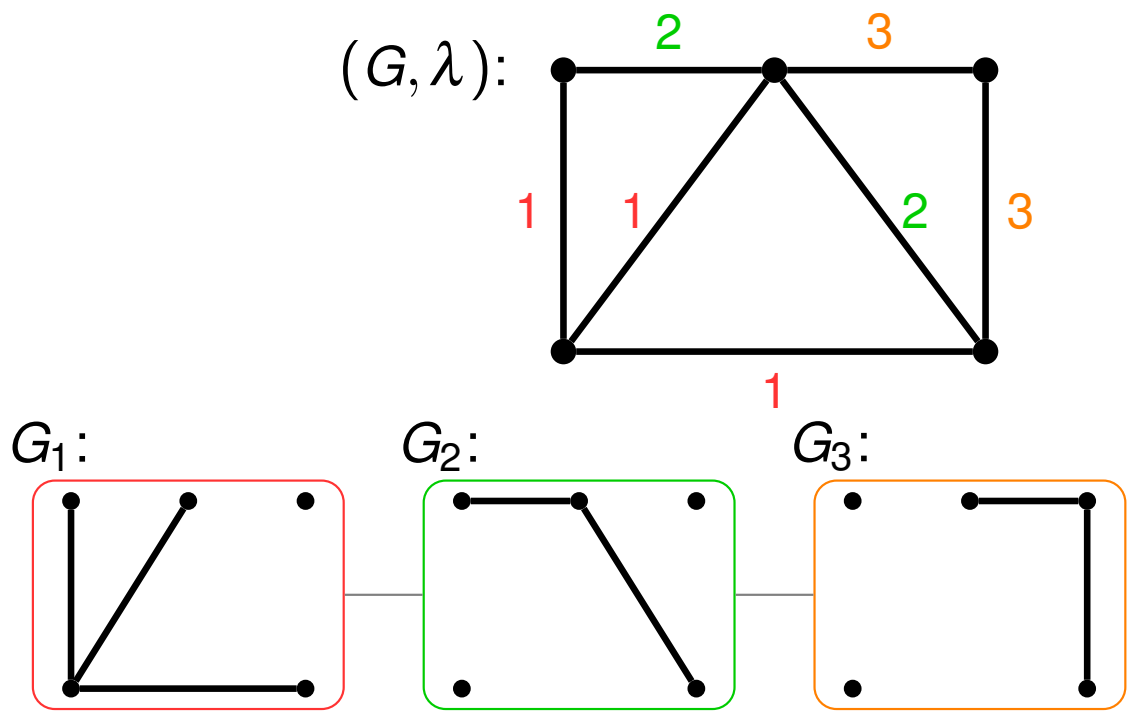
Temporal Graph

A **temporal graph** $(G = (V, E), \lambda)$ is defined as a graph $G = (V, E)$ with a labeling function $\lambda : E \to 2^{\mathbb{N}}$ that assigns time labels to edges.



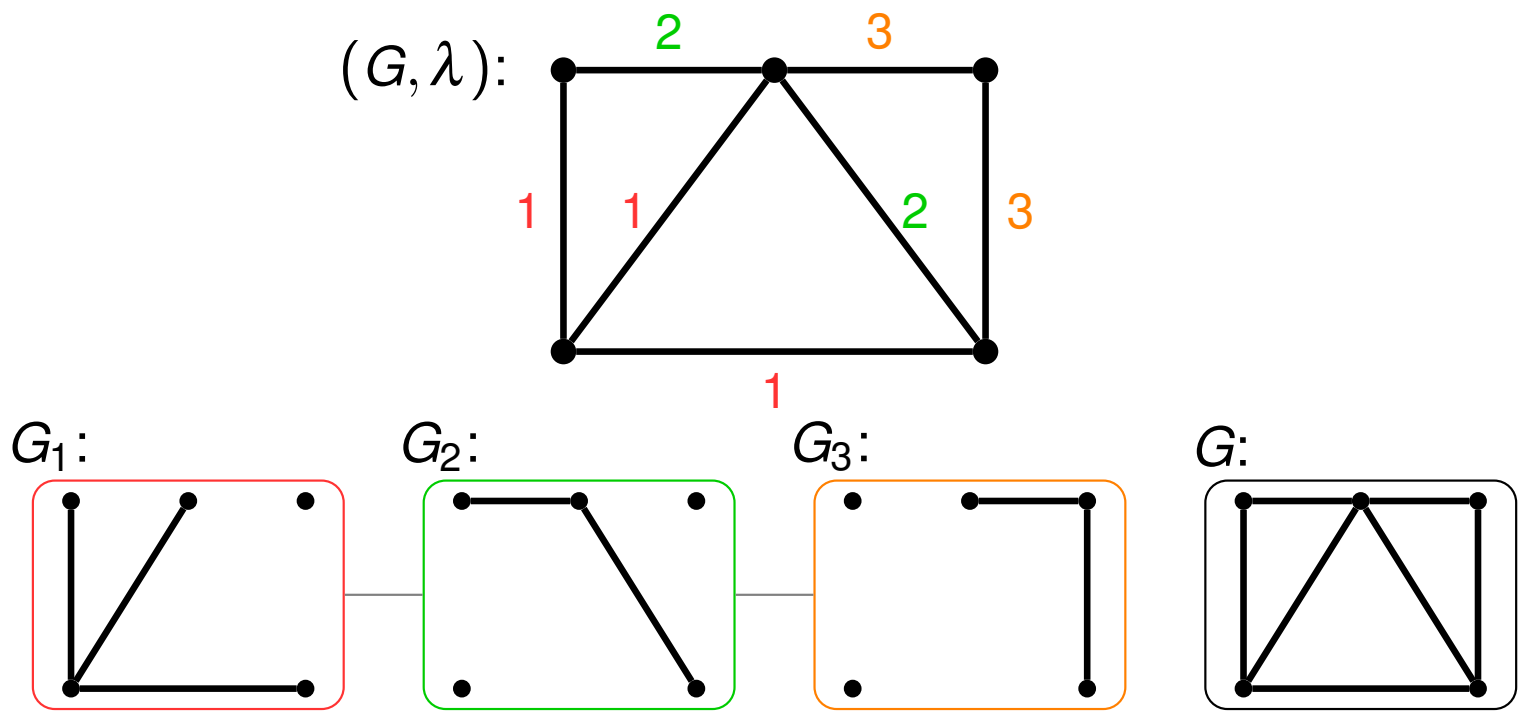
Temporal Graph

A **temporal graph** $(G = (V, E), \lambda)$ is defined as a graph $G = (V, E)$ with a labeling function $\lambda : E \to 2^{\mathbb{N}}$ that assigns time labels to edges.



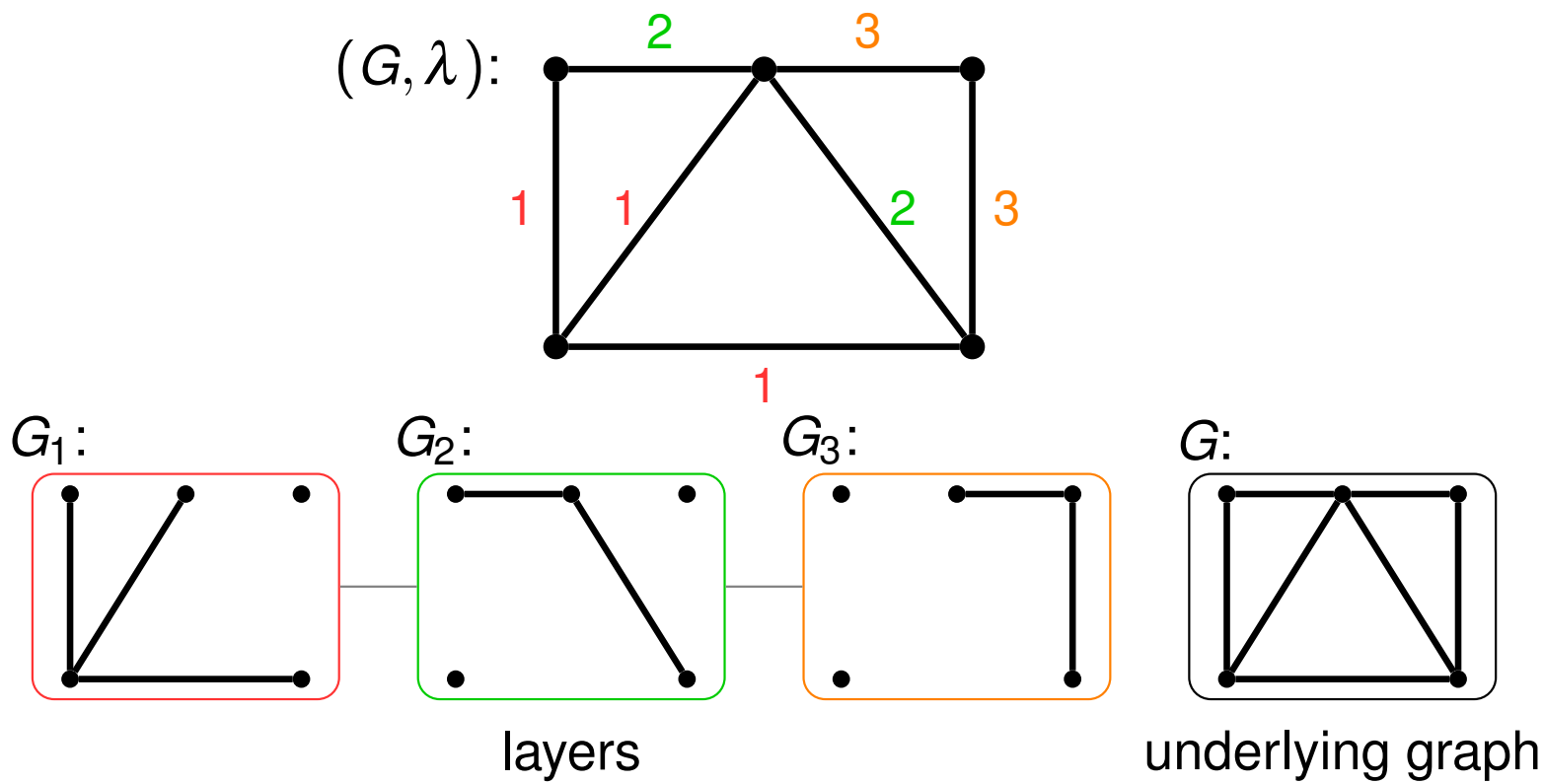
Temporal Graph

A **temporal graph** $(G = (V, E), \lambda)$ is defined as a graph $G = (V, E)$ with a labeling function $\lambda : E \to 2^{\mathbb{N}}$ that assigns time labels to edges.



Temporal Graph

A **temporal graph** $(G = (V, E), \lambda)$ is defined as a graph $G = (V, E)$ with a labeling function $\lambda : E \to 2^{\mathbb{N}}$ that assigns time labels to edges.



Sliding Window Temporal Coloring, Example, Motivation, Definition

Sliding Window Temporal Coloring

Input: A temporal graph (G, λ), and two integers $k \in \mathbb{N}$ and $\Delta \leq T$.
Question: Does there exist a proper sliding Δ-window temporal coloring ϕ of (G, λ) using at most k colors?

Parameterized Tractability

- **FPT** (fixed-parameter tractable): Solvable in $f(k) \cdot n^{O(1)}$ time.

n: instance size

k: parameter

Parameterized Tractability

- **FPT** (fixed-parameter tractable): Solvable in $f(k) \cdot n^{O(1)}$ time.
- **XP**: Solvable in $n^{g(k)}$ time.

n: instance size

k: parameter

Parameterized Tractability

- **FPT** (fixed-parameter tractable): Solvable in $f(k) \cdot n^{O(1)}$ time.
- **XP**: Solvable in $n^{g(k)}$ time.
- **Polynomial Kernel**: Poly-time algorithm transforming an instance (I, k) into an equivalent instance (I', k') s.t. $|(I', k')| \leq k^{O(1)}$.

n: instance size

k: parameter

Parameterized Tractability

- **FPT** (fixed-parameter tractable): Solvable in $f(k) \cdot n^{O(1)}$ time.
- **XP**: Solvable in $n^{g(k)}$ time.
- **Polynomial Kernel**: Poly-time algorithm transforming an instance (I, k) into an equivalent instance (I', k') s.t. $|(I', k')| \leq k^{O(1)}$.

Parameterized Hardness

- **W[1]-hard**: Presumably no FPT algorithm (XP algorithm possible).

n: instance size

k: parameter

Parameterized Tractability

- **FPT** (fixed-parameter tractable): Solvable in $f(k) \cdot n^{O(1)}$ time.
- **XP**: Solvable in $n^{g(k)}$ time.
- **Polynomial Kernel**: Poly-time algorithm transforming an instance (I, k) into an equivalent instance (I', k') s.t. $|(I', k')| \leq k^{O(1)}$.

Parameterized Hardness

- **W[1]-hard**: Presumably no FPT algorithm (XP algorithm possible).
- **para-NP-hard**: NP-hard for constant k (no XP algorithm).

n: instance size k: parameter

Temporal Graph Coloring
Table of Results

Hardness Results:

- Sliding Window Temporal Coloring is NP-hard, even if k, Δ, and T are constant and
  - G is $k+1$-colorable and has $O(k)$ max. degree, and every snapshot has connected components of size $O(k)$.
  - Every snapshot is a cluster graph.
  - Every snapshot has a dominating set of size one.
- Sliding Window Temporal Coloring is NP-hard, even if k and Δ are constant and the vertex cover number of the underlying graph is in $O(k)$.

Algorithmic Results:

- Exponential Time Algorithm that is optimal assuming ETH.
- Extension for small number of agents (FPT Algorithm).
- FPT-Approximation algorithm for parameter “feedback vertex number of G” (additive error of one).

Sliding Window Temporal Graph Coloring
Main Algorithm

Sketch of the main exponential time Algorithm (Thm 4.5)

Sliding Window Temporal Graph Coloring
Main Algorithm II

How to exploit few vertices? $\rightarrow$ Preprocessing Step and FPT algorithm for # of vertices (Thm 4.6) (Motivation + Main Ideas)

Temporal Graph Coloring
Approximation Algorithm

Vertex Cover FPT algorithm (Thm 4.9) (Motivation + Main Ideas)

Outlook
and Future Work