UNIVERSITÀ DI PISA

Dynamic Graphs: The Long-Range Challenge

Mauriana Pesaresi Seminar Series

Fabrizio De Castelli, 06/03/2026

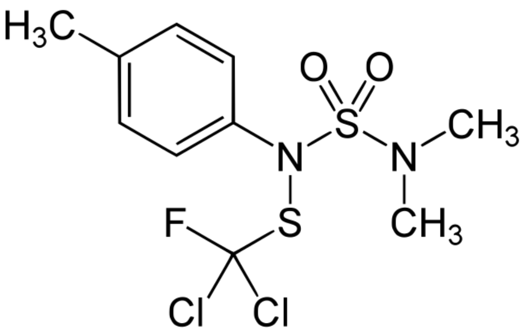
Outline

- Static and Dynamic Graphs
- Graph Neural Networks
- The Long-Range Propagation Problem
- Long-Range Adaptive Memory Retention
- Spatio-Temporal Long-Range Benchmarks

Static and Dynamic Graphs

Fabrizio De Castelli, 06/03/2026

Graphs are **everywhere!**



Molecules

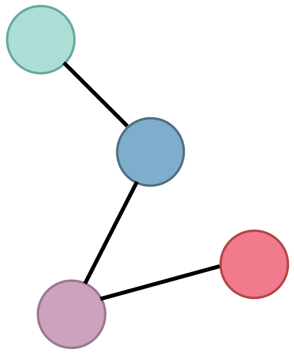
Social Networks

Transportation Networks

And many others…

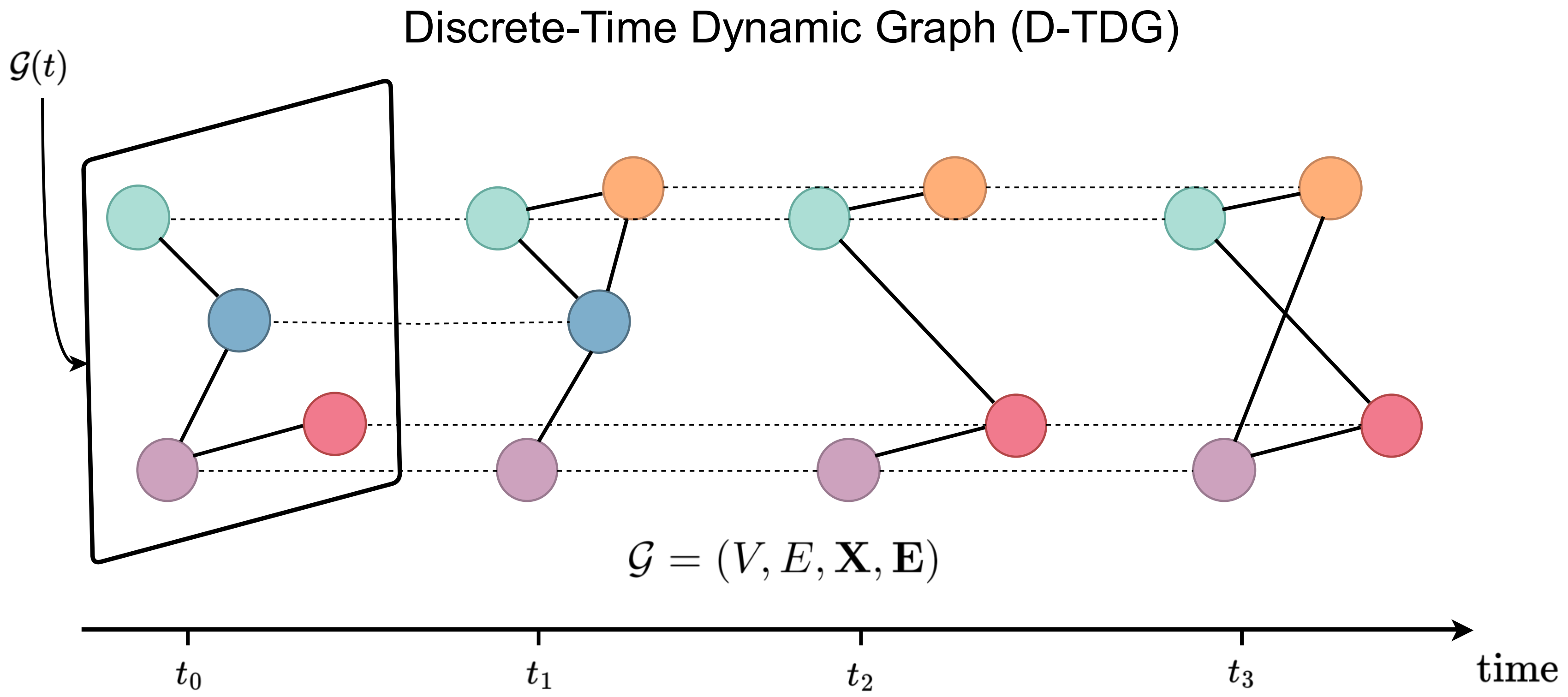
From Static to Dynamic Graphs

Static Graph



$$\mathcal{G} = (V, E, \mathbf{X}, \mathbf{E})$$

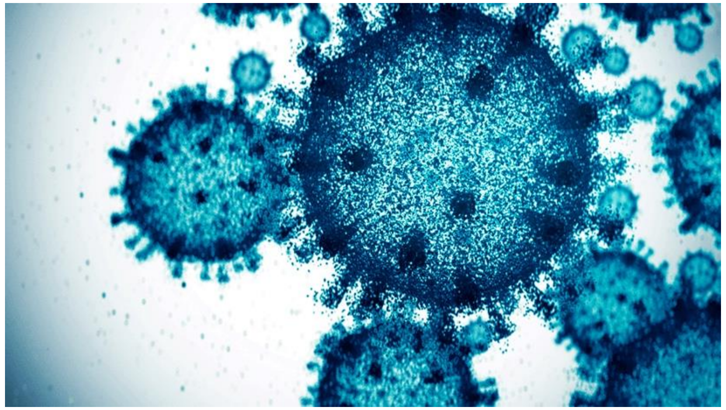
From Static to Dynamic Graphs



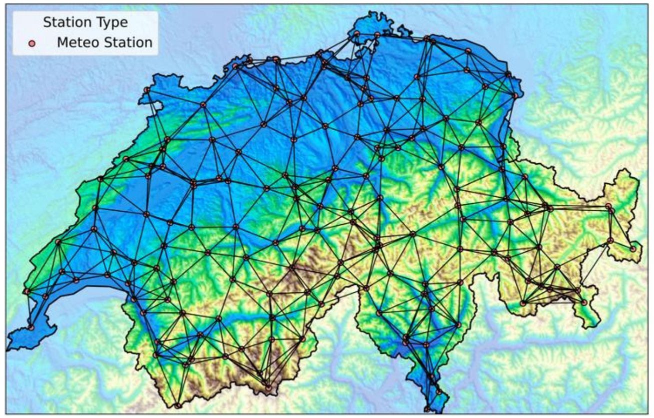
$$\mathcal{G}(t) = (V(t), E(t), \mathbf{X}(t), \mathbf{E}(t))$$

The World is Dynamic

Transportation Networks
“Will a road be congested?”



Infections
“Who will get infected?”



Climate Forecasting
“Will it be windy?”

Research Questions

- How do we handle time to learn good graph representations?
- Can we predict when and how the graph will change in the future?

Zambon et al., “*PeakWeather: MeteoSwiss Weather Station Measurements for Spatiotemporal Deep Learning*”, 2025

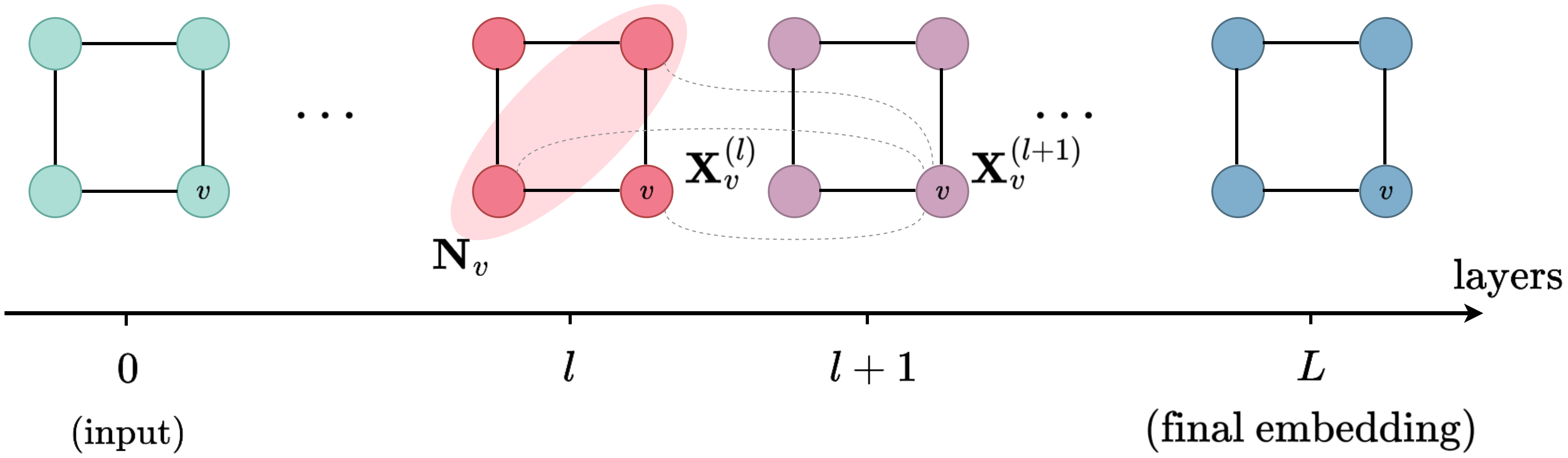
Graph Neural Networks

Fabrizio De Castelli, 06/03/2026

Graph Neural Networks (GNNs)

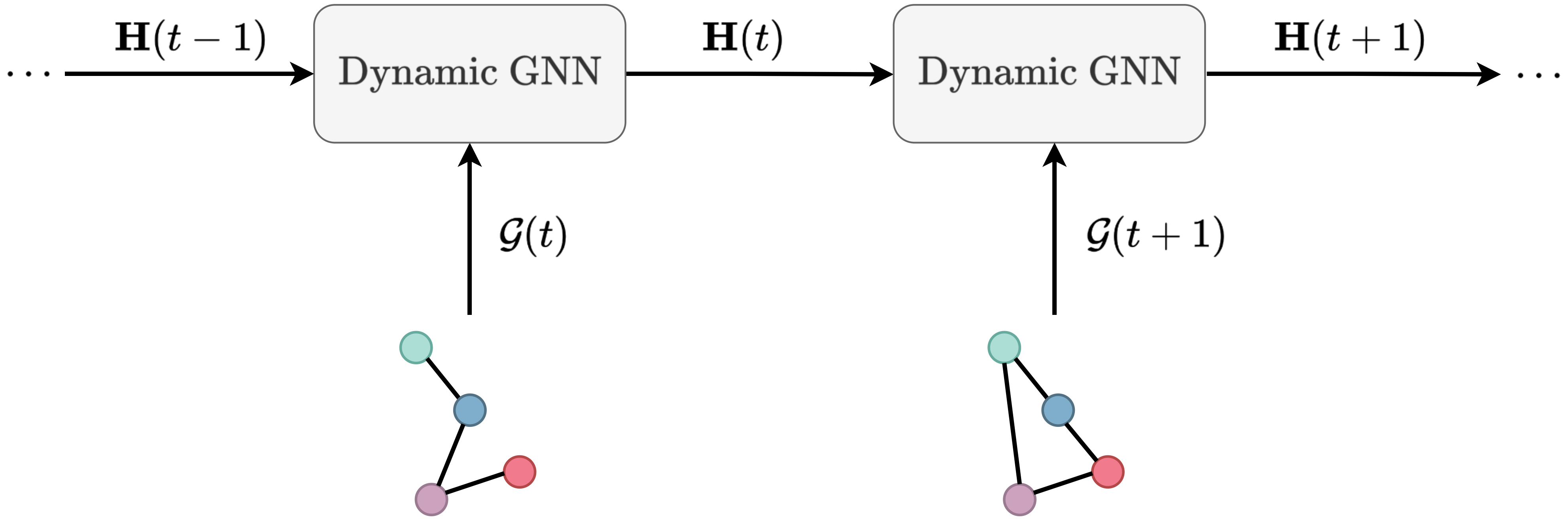
Neural Message Passing Framework

Update the representation of a node aggregating neighbors' messages



Gilmer et al., "*Neural Message Passing for Quantum Chemistry*", 2017

Dynamic Graph Neural Networks

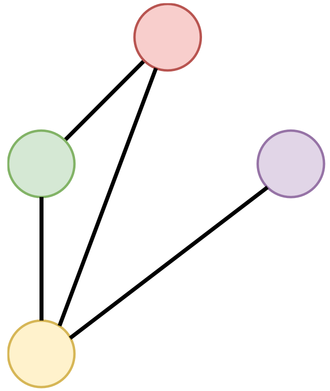


Possible implementations

- **Time and space** (T&S)
- **Time then space** (TTS)

Long-Range Propagation

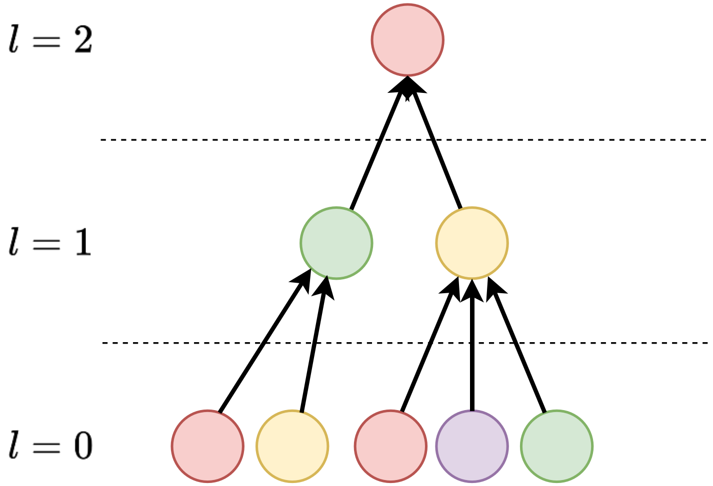
Long-Range Propagation



- **Problem**: learning on graphs typically needs to consider a specific range of node interactions
  - Multiple rounds of message passing widens the receptive field
  - But... The number of involved node representations **grows exponentially** with rounds (layers)

- For temporal graphs, this issue repeats in the sequence

Computational Graph



Di Giovanni et al., "*On Over-Squashing in Message Passing Neural Networks: The Impact of Width, Depth and Topology*", 2017

Towards Adaptive Memory Retention

- **Problem**
  - Dissipative models *forget* knowledge
  - Non-dissipative models *keep building up* information without discarding irrelevant signals
- Long-range modeling requires **filtering irrelevant data**, especially on sequences
- **Solution**
  - Take a non-dissipative architecture
  - Given the input sequence, **learn the optimal degree of dissipation**

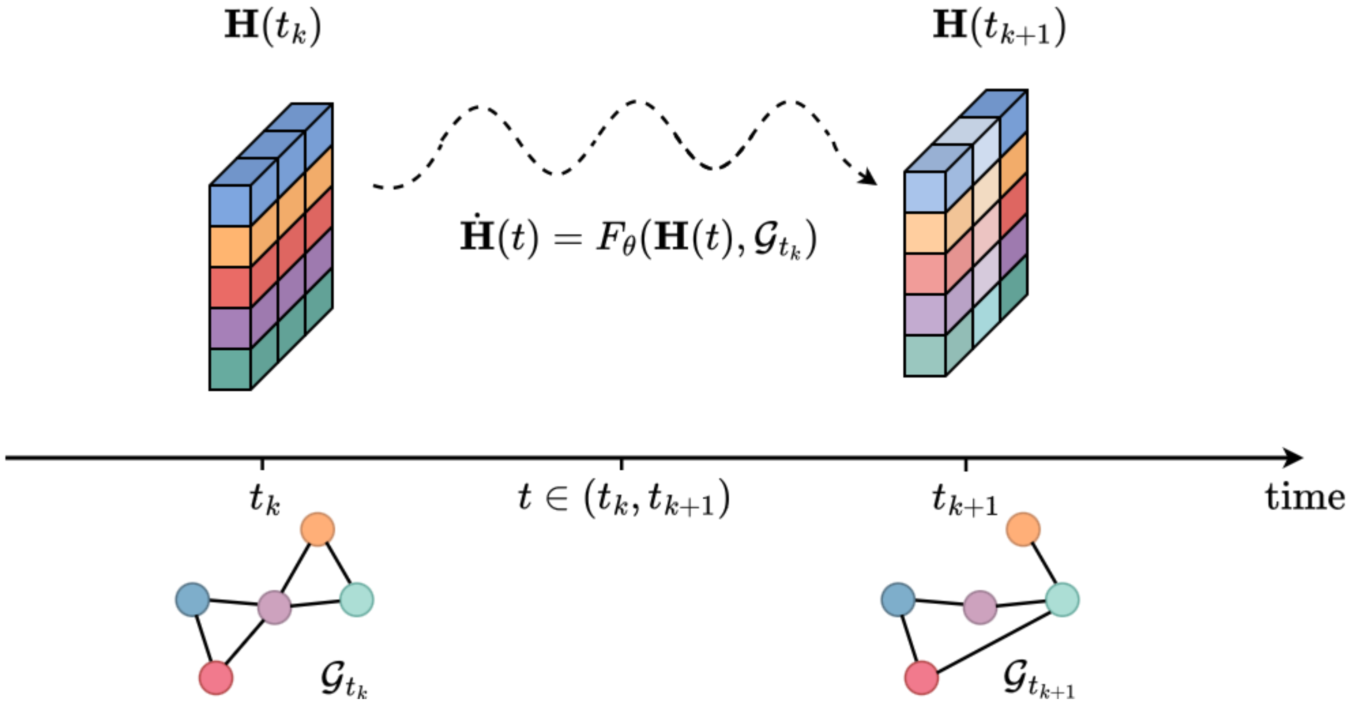
Long-Range Adaptive Memory Retention

Fabrizio De Castelli, 06/03/2026

The Dynamical System Perspective

GNNs ≡ discretization of a continuous process

$$\begin{cases} \dot{\mathbf{H}}(t) = F_\theta(\mathbf{H}(t); \mathcal{G}_{t_k}) & t \in (t_k, t_{k+1}) \\ \mathbf{H}(t^+) = G_\phi(\mathbf{H}(t^-), \mathbf{X}(t)) & t \in \mathcal{T} \\ \mathbf{H}(0) = \mathbf{H}_0 & t = t_0 \end{cases}$$



Long-range Adaptive Memory Retention

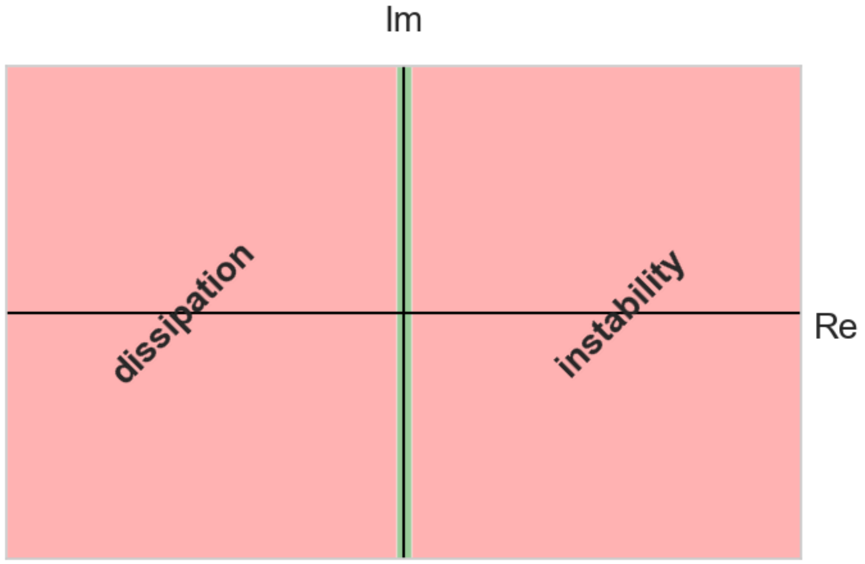
Notation

- For any **square** matrix M
  - Subtracting its transpose yields an **antisymmetric** matrix $\mathbf{M}_{\text{sk}} = \mathbf{M} - \mathbf{M}^\top$
  - Summing its transpose yields a symmetric matrix $\mathbf{M}_{\text{sym}} = \mathbf{M} + \mathbf{M}^\top$
- **Idea**
  - Parametrize the model's space diffusion with **antisymmetric** weight matrices
  - **Relax** the antisymmetry via a learned diagonal damping term
  - Integrate external inputs in the system with a **controlled** linear combination

The importance of the Parametrization

Relaxed antisymmetry ensures **stablility** and **controlled dissipation**

- $Re(\lambda_i(\mathbf{J}(t))) > 0$ ⟶ unstable system
- $Re(\lambda_i(\mathbf{J}(t))) \ll 0$ ⟶ long-range dependencies are forgotten
- $Re(\lambda_i(\mathbf{J}(t))) \approx 0$ ⟶ **non-dissipative** system
  - No gradient vanishing/explosion during training



Long-range Adaptive Memory Propagation

- **Space:** information diffusion via message passing and adaptive dissipativity

$$F_\theta = \sigma\Big(\mathbf{H}(t)(\mathbf{W}_{\text{sk}} - \mathbf{\Gamma}) + \mathcal{S}(\mathbf{A}(t_k), \mathbf{H}(t); \mathbf{V}_{\text{sk}}) + \mathcal{K}(\mathbf{A}(t_k), \mathbf{H}(t); \mathbf{Z}_{\text{sym}})\Big)$$

Adaptive memory retention

Example of a 3x3 matrix

$$\begin{pmatrix} 0 & w_{12} & w_{13} \\ -w_{12} & 0 & w_{23} \\ -w_{13} & -w_{23} & 0 \end{pmatrix} - \begin{pmatrix} s(\gamma_1) & 0 & 0 \\ 0 & s(\gamma_2) & 0 \\ 0 & 0 & s(\gamma_3) \end{pmatrix} = \begin{pmatrix} -s(\gamma_1) & w_{12} & w_{13} \\ -w_{12} & -s(\gamma_2) & w_{23} \\ -w_{13} & -w_{23} & -s(\gamma_3) \end{pmatrix}$$

θ: Model Parameters

Long-range Adaptive Memory Propagation

- **Space:** information diffusion via message passing and adaptive dissipativity

$$F_\theta = \sigma\Big(\mathbf{H}(t)(\mathbf{W}_{\text{sk}} - \mathbf{\Gamma}) + \mathcal{S}(\mathbf{A}(t_k), \mathbf{H}(t); \mathbf{V}_{\text{sk}}) + \mathcal{K}(\mathbf{A}(t_k), \mathbf{H}(t); \mathbf{Z}_{\text{sym}})\Big)$$

Weight-antisymmetric neighborhood aggregation

Space-antisymmetric neighborhood aggregation

- **Time:** external impulses (inputs) integration in the system

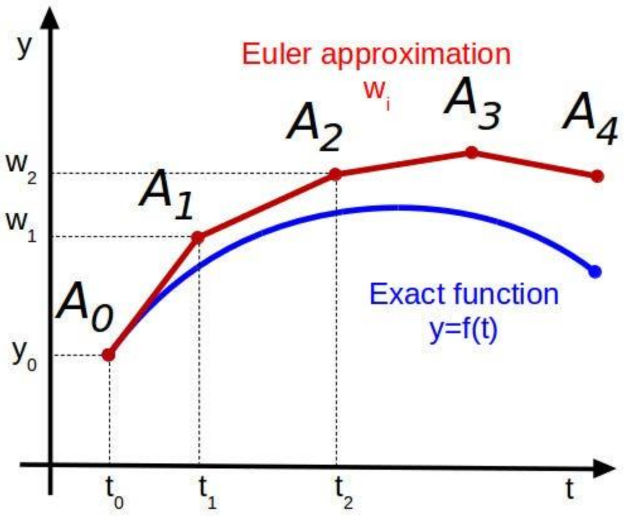
$$G_\phi = \frac{1}{\xi}\Big(\mathbf{H}(t_k^-) + \mu \mathbf{X}(t_k)\mathbf{U}\Big),$$

θ, ϕ: Model Parameters

Practical Implementation

- **Space:** Forward Euler discretization of the ODE for L steps, with step size ϵ

$$\mathbf{H}^{(l+1)}(t) = \mathbf{H}^{(l)}(t) + \epsilon\, F_\theta\big(\mathbf{H}^{(l)}(t); \mathcal{G}_t\big)$$



Practical Implementation

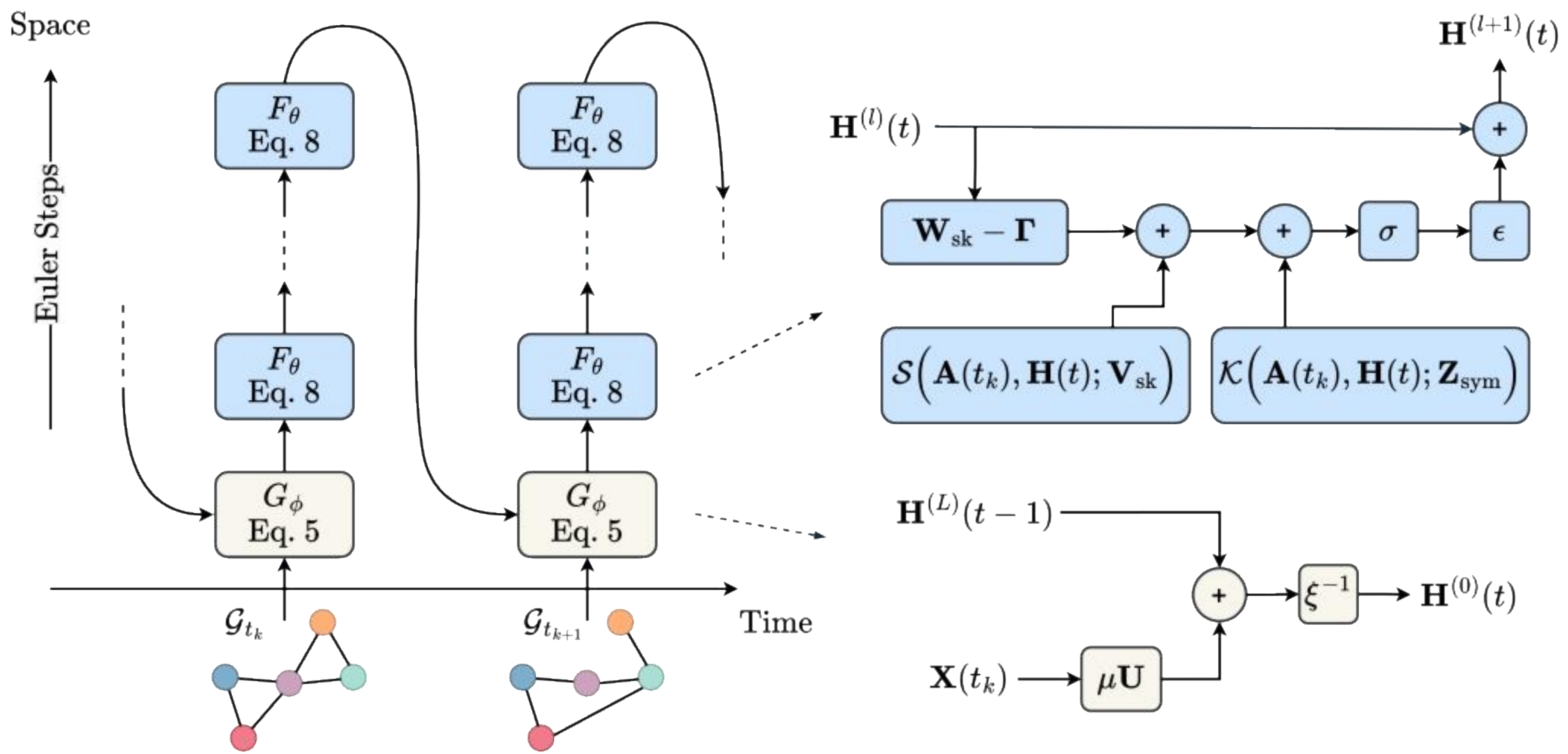
- **Space:** Forward Euler discretization of the ODE for L steps, with step size ϵ

$$\mathbf{H}^{(l+1)}(t) = \mathbf{H}^{(l)}(t) + \epsilon\, F_\theta\big(\mathbf{H}^{(l)}(t); \mathcal{G}_t\big)$$

- **Time:** Take last Forward Euler step and combine it with the input

$$\mathbf{H}^{(0)}(t+1) = \frac{1}{\xi}\Big(\mathbf{H}^{(L)}(t) + \mu\mathbf{X}(t_k)\mathbf{U}\Big),$$

LAMP Architecture

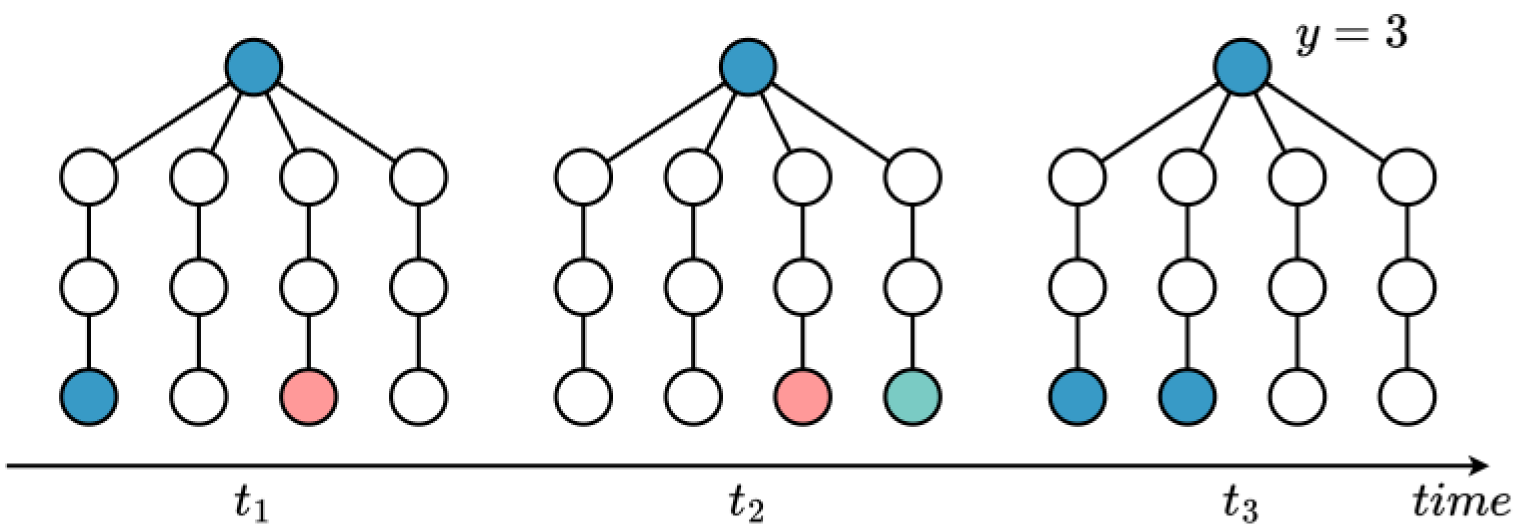


Spatio-Temporal Long-Range Benchmark

Fabrizio De Castelli, 06/03/2026

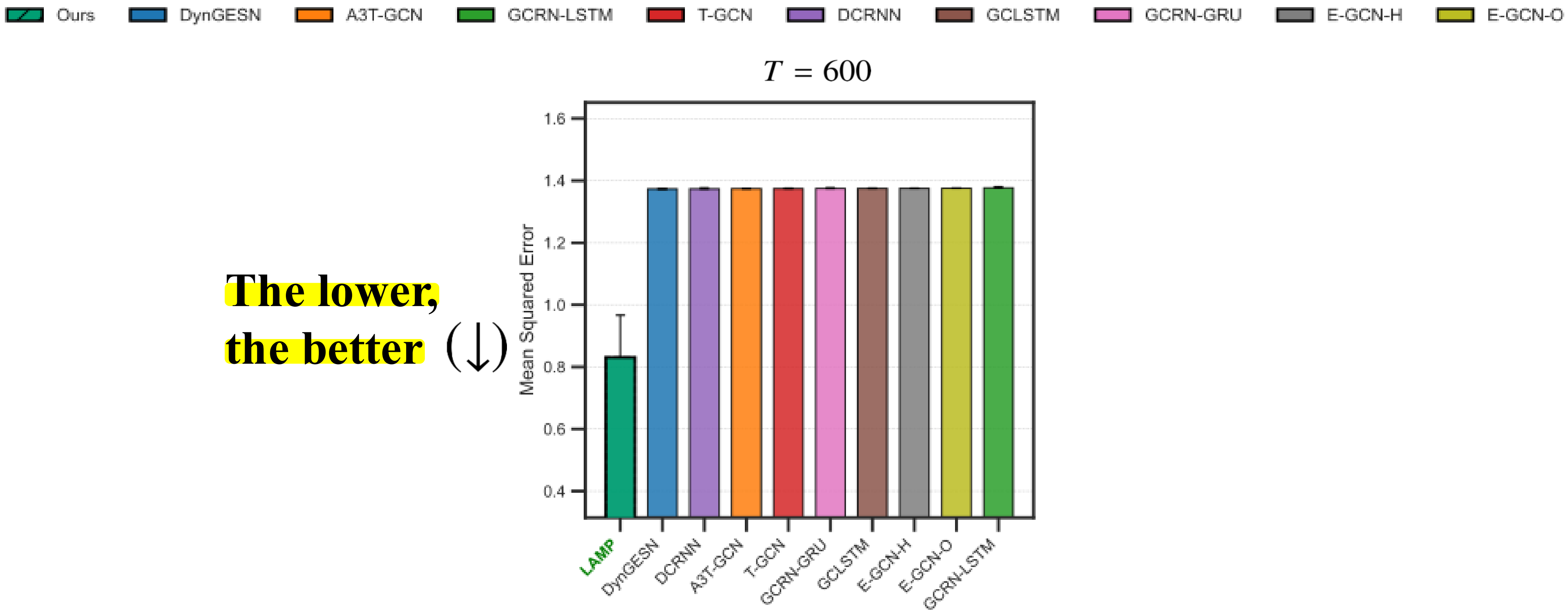
Colored Leaf Counting

- **Input**: N sequences of length T, number of colors K, spider graph topology
- **Target**: Count how many times leaves emitted the same color of the root



Colored Leaf Counting – Single Color

Can the model propagate and accumulate sparse temporal signals between distant nodes?

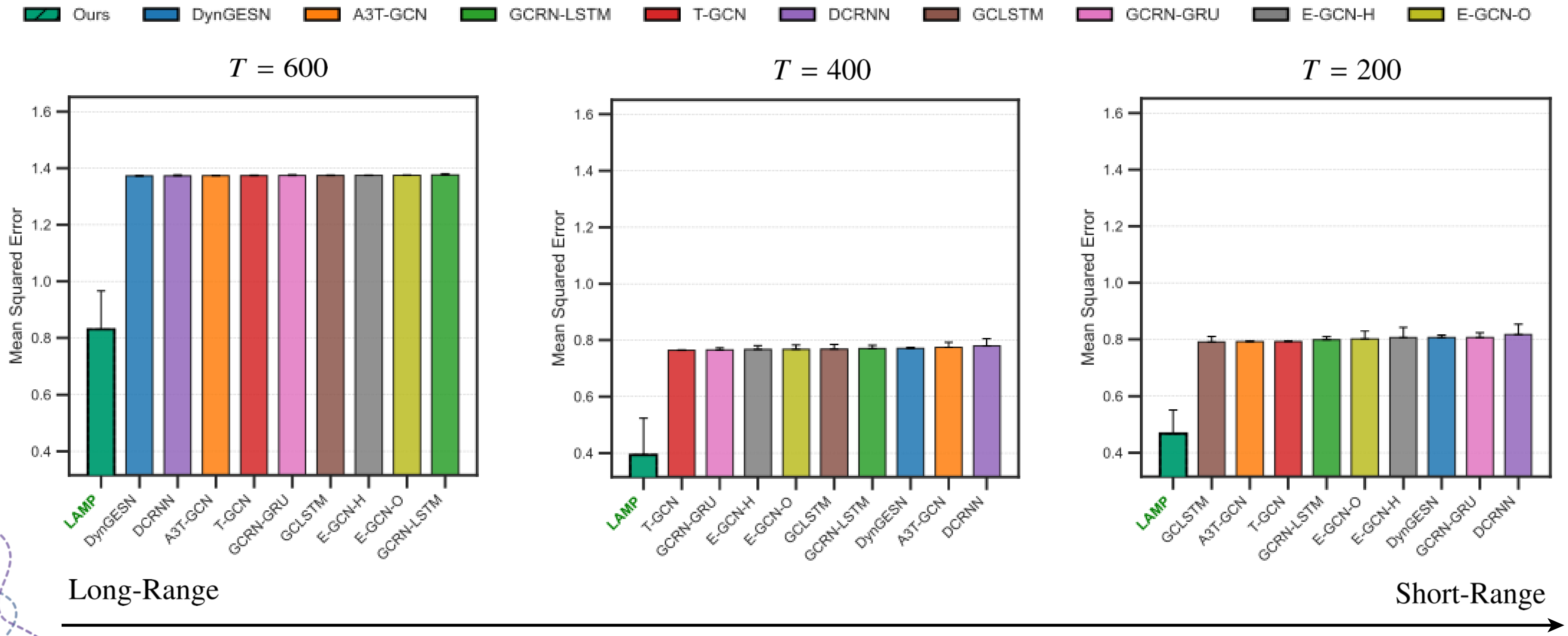


The lower, the better (↓)

Fabrizio De Castelli, 06/03/2026

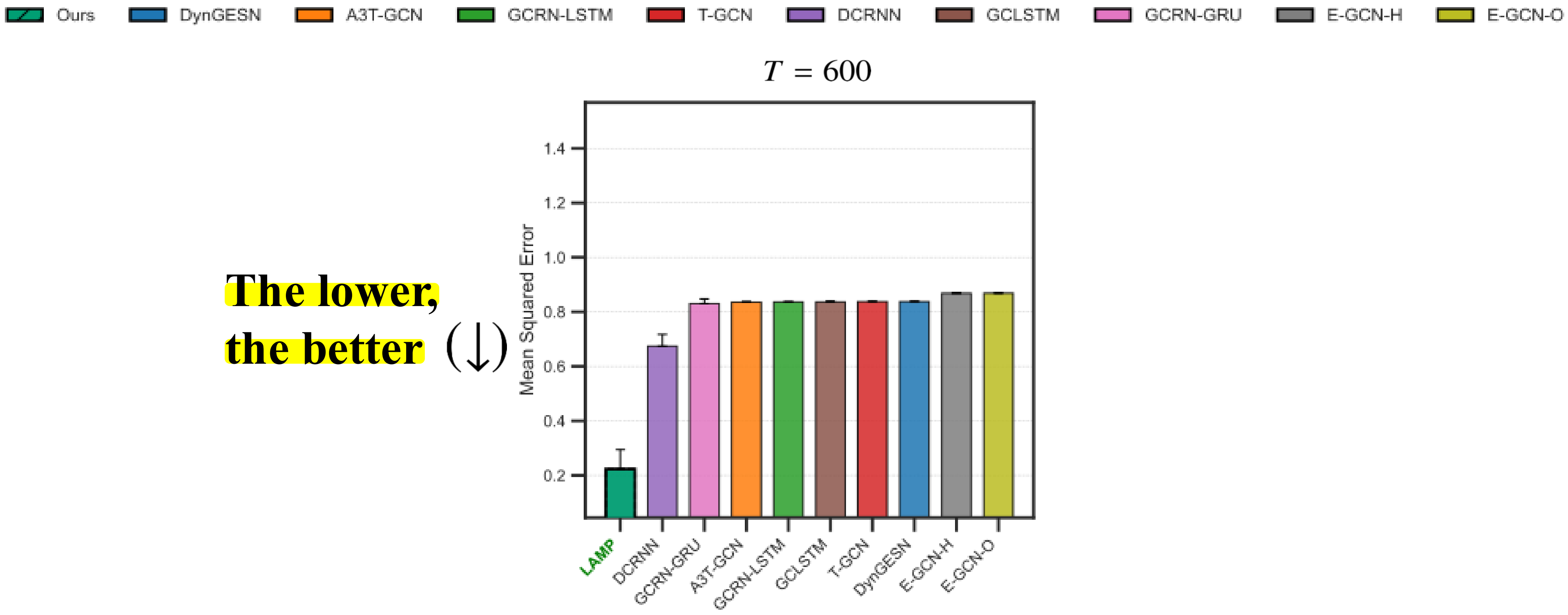
Colored Leaf Counting – Single Color

Can the model propagate and accumulate sparse temporal signals between distant nodes?



Colored Leaf Counting – Three Colors

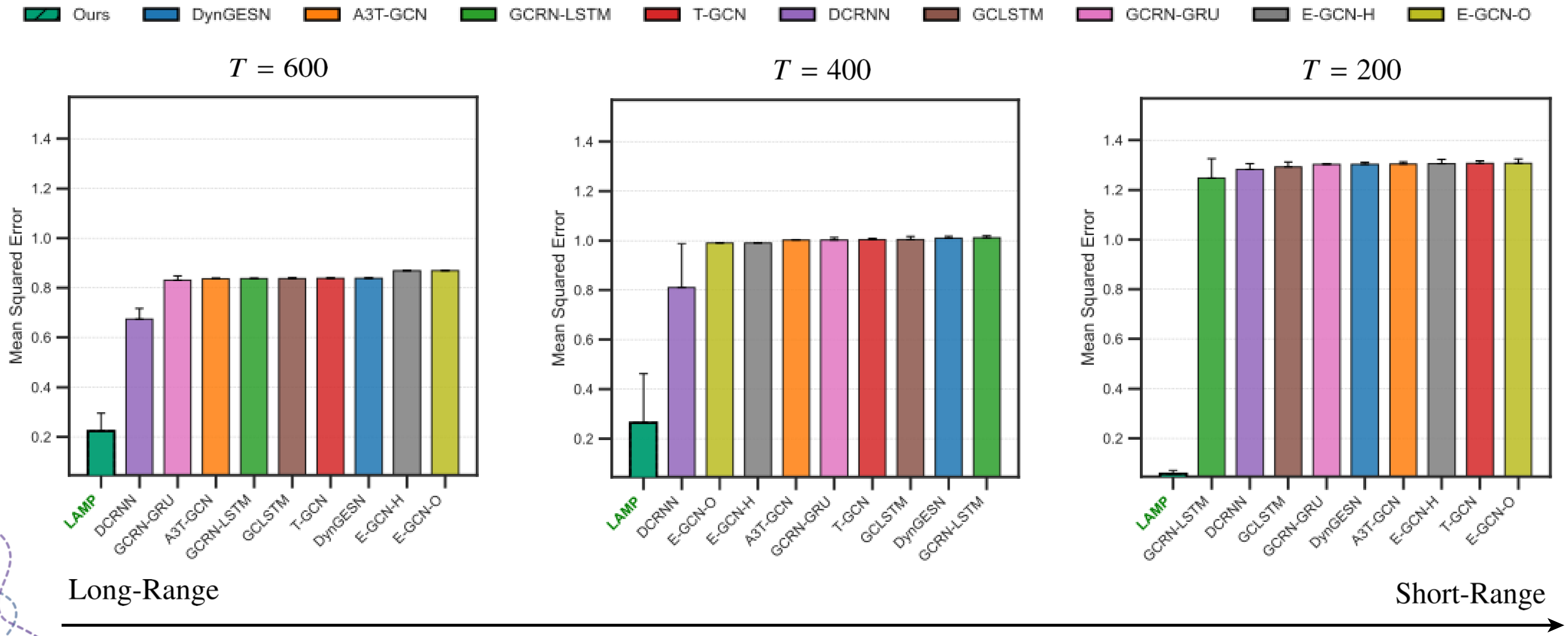
Can the model propagate and accumulate sparse temporal signals between distant nodes?



The lower, the better (↓)

Colored Leaf Counting – Three Colors

Can the model propagate and accumulate sparse temporal signals between distant nodes?



Conclusions

- Long-Range Propagation is an **open problem**
- A clever architectural design helps limiting this problem
- **Questions**
  - What happens if the set of nodes **changes** over time?
  - How to ensure a **node-level** adaptive dissipation rate?
  - Is there a way of having instantaneous propagation between snapshots?