network science institute

Complexity Society Lab

NetSci maastricht

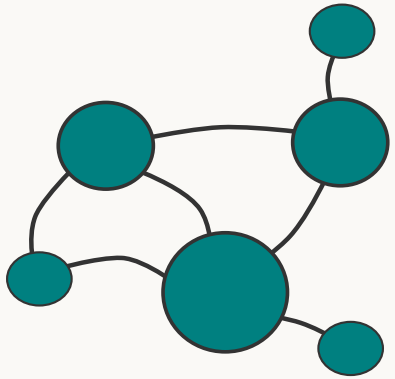
Generalization and Efficiency of Graph Neural ODEs for Network Dynamical Systems

Moritz Laber, Brennan Klein

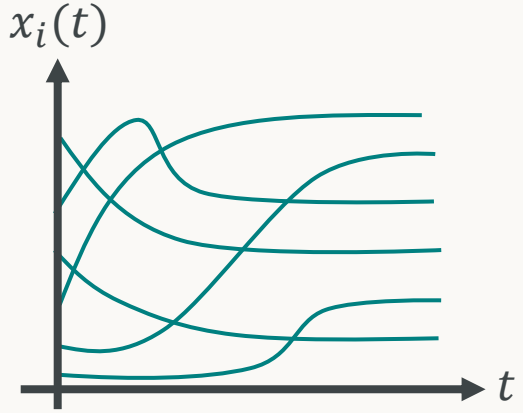
✉ **laber.m@northeastern.edu**

NetSci2025, Maastricht, 2025-06-05

Dynamics from Data



network A

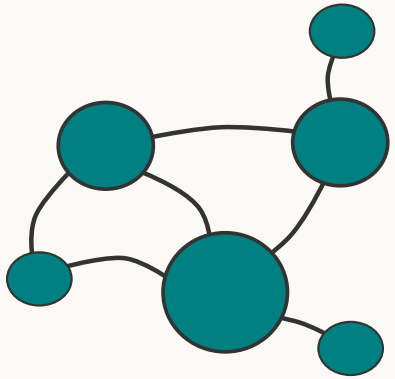


node-wise
timeseries $x_i(t)$

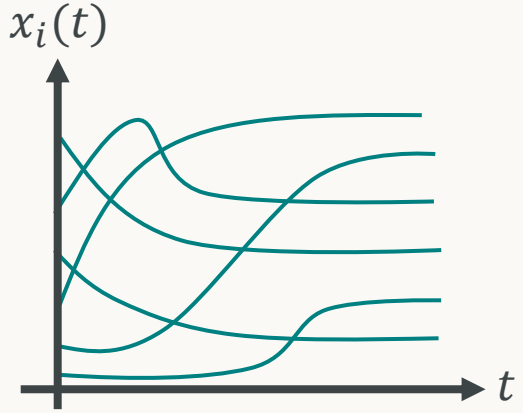
Dynamics from Data

Classical ODEs:
Handcraft differential equations using domain knowledge.

$$\frac{dx_i}{dt} = f(x_1, \dots, x_n)$$



network A

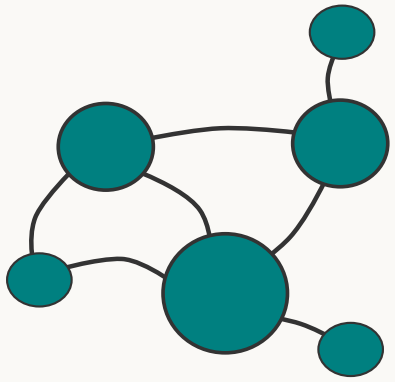


node-wise timeseries $x_i(t)$

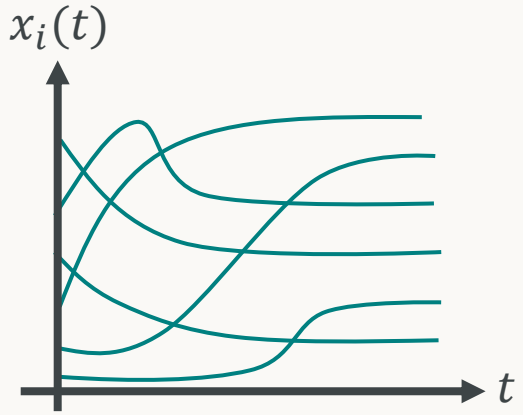
Dynamics from Data

Classical ODEs:
Handcraft differential equations using domain knowledge.

$$\frac{dx_i}{dt} = f(x_1, \dots, x_n)$$



network A



node-wise timeseries $x_i(t)$

Neural ODEs [1,2,3]**:**
Learn differential equations from timeseries data.

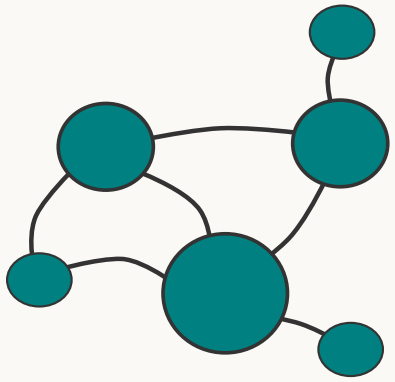
$$\frac{dx_i}{dt} = f_\omega(x_1, \dots, x_n)$$

[1] *Chen et al.* NeurIPS (2018), [2] *Poli et al.* AAAI (2022), [3] *Kidger* PhD Thesis (2022)

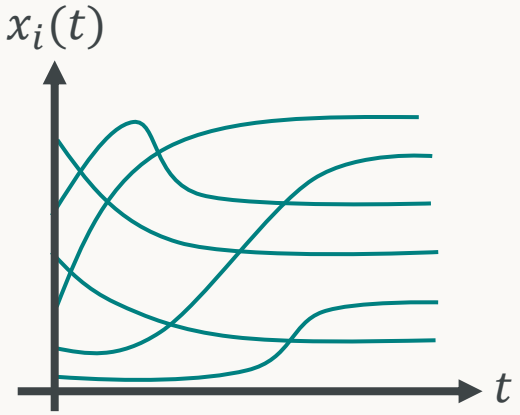
Dynamics from Data

Classical ODEs:
Handcraft differential equations using domain knowledge.

$$\frac{dx_i}{dt} = f(x_1, \dots, x_n)$$



network A



node-wise timeseries $x_i(t)$

Neural ODEs [1,2,3]**:**
Learn differential equations from timeseries data.

$$\frac{dx_i}{dt} = f_\omega(x_1, \dots, x_n)$$

Questions: How do the performance and robustness of neural ODEs depend on the network? [4]

[1] *Chen et al.* NeurIPS (2018), [2] *Poli et al.* AAAI (2022), [3] *Kidger* PhD Thesis (2022)

[4] *Vasiliauskaite & Antulov-Fantulin* Comms Phys. 7 1 (2024)

Contributions

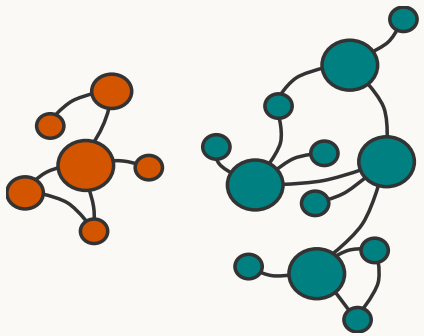
We train neural ODEs on **different dynamical systems** and **graphs from the $\mathbb{S}^1$ model** with different structural characteristics

Contributions

We train neural ODEs on **different dynamical systems** and **graphs from the $\mathbb{S}^1$ model** with different structural characteristics

We evaluate their **performance and robustness** in terms of four criteria:

Generalization:
Network Size



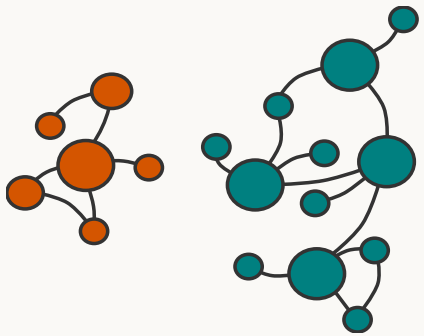
$n^{\text{train}} \ll n^{\text{test}}$

Contributions

We train neural ODEs on **different dynamical systems** and **graphs from the $\mathbb{S}^1$ model** with different structural characteristics

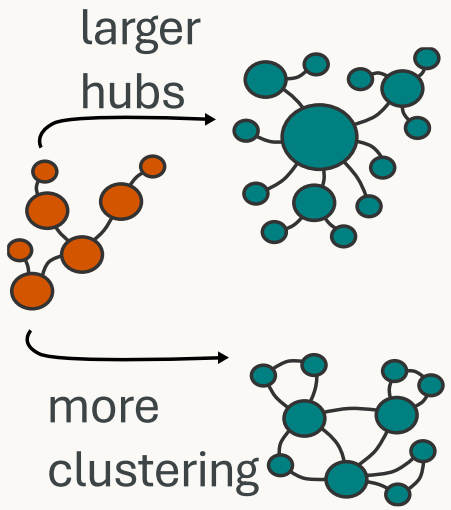
We evaluate their **performance and robustness** in terms of four criteria:

Generalization: Network Size



$n^{\text{train}} \ll n^{\text{test}}$

Generalization: Network Properties

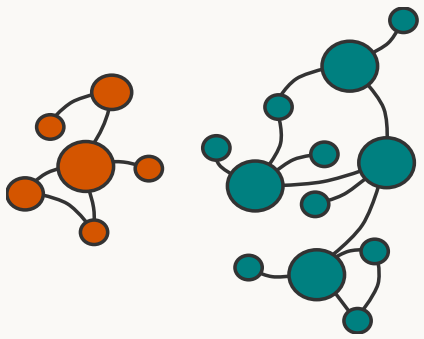


Contributions

We train neural ODEs on **different dynamical systems** and **graphs from the $\mathbb{S}^1$ model** with different structural characteristics

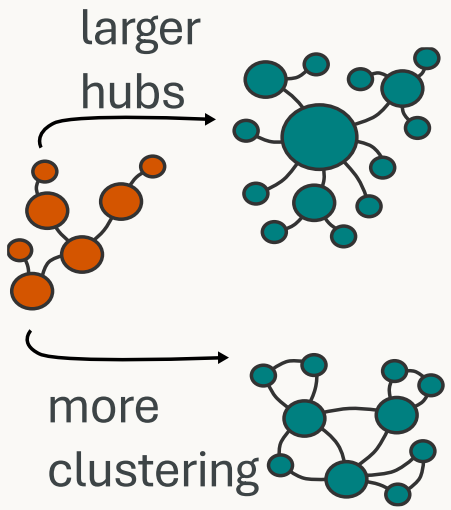
We evaluate their **performance and robustness** in terms of four criteria:

Generalization: Network Size

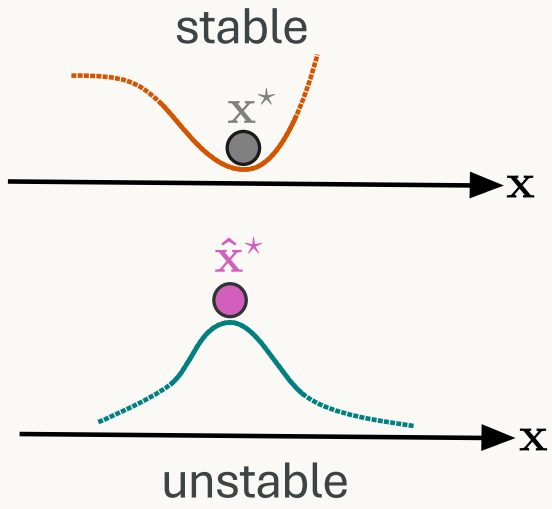


$n^{\text{train}} \ll n^{\text{test}}$

Generalization: Network Properties



Robustness: Local Stability

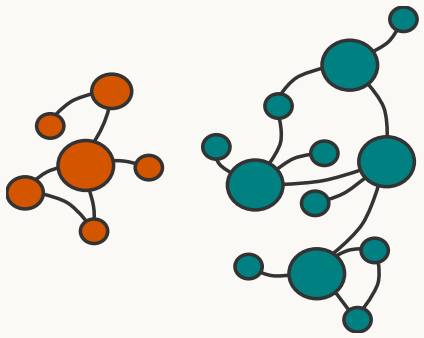


Contributions

We train neural ODEs on **different dynamical systems** and **graphs from the $\mathbb{S}^1$ model** with different structural characteristics

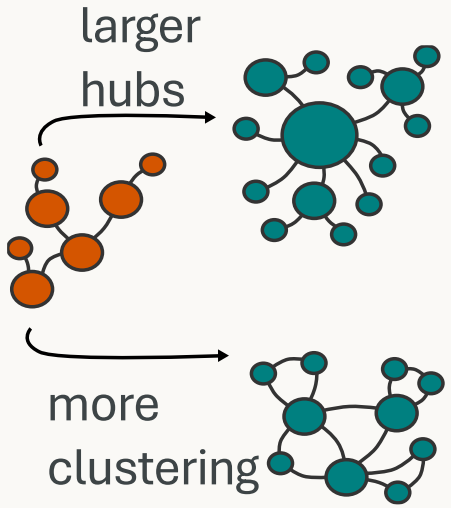
We evaluate their **performance and robustness** in terms of four criteria:

Generalization: Network Size

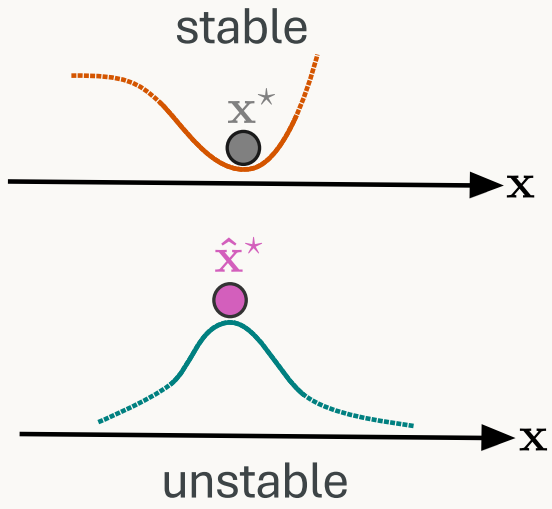


$n^{\text{train}} \ll n^{\text{test}}$

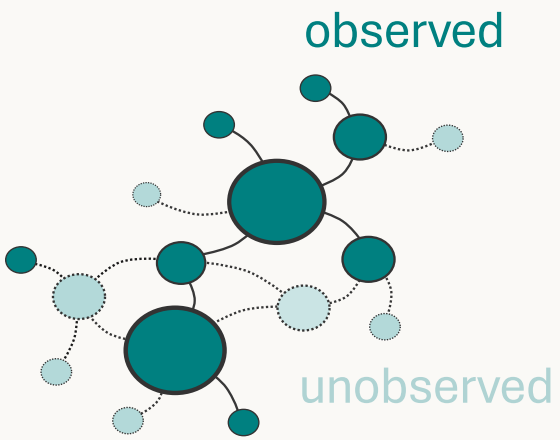
Generalization: Network Properties



Robustness: Local Stability



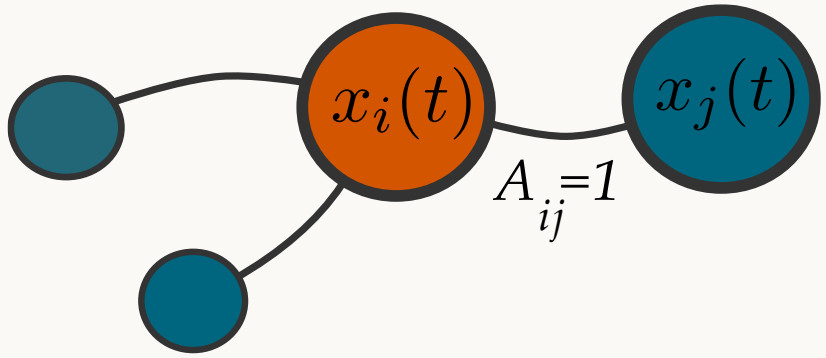
Robustness: Missing Nodes



(Neural) ODEs on Networks

Many dynamical systems on networks are well described by [1]

$$\frac{dx_i}{dt} = f\left(x_i(t)\right) + \sum_{j \in \mathcal{V}} A_{ij} h^{\text{ego}}\left(x_i(t)\right) h^{\text{alt}}\left(x_j(t)\right)$$

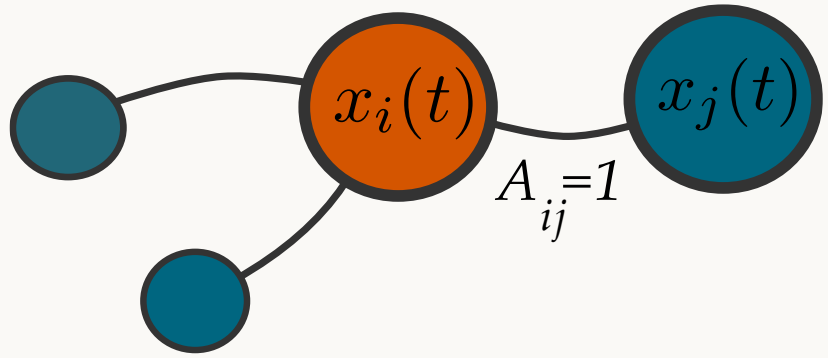


[1] *Barzel & Barabási* Nat. Phys. 9 673-681 (2013)

(Neural) ODEs on Networks

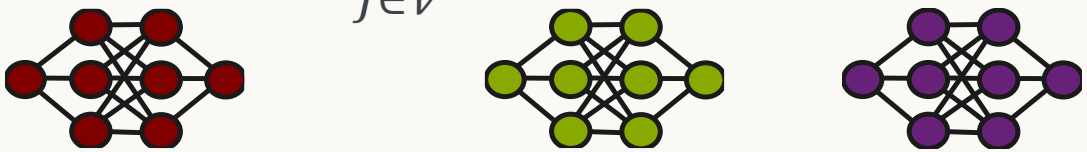
Many dynamical systems on networks are well described by [1]

$$\frac{dx_i}{dt} = f\left(x_i(t)\right) + \sum_{j \in \mathcal{V}} A_{ij} h^{\text{ego}}\left(x_i(t)\right) h^{\text{alt}}\left(x_j(t)\right)$$



These equations form also the basis for our neural ODE architecture

$$\frac{dx_i}{dt} = f_\omega\left(x_i(t)\right) + \sum_{j \in \mathcal{V}} A_{ij}\, h_\omega^{\text{ego}}\left(x_i(t)\right) h_\omega^{\text{alt}}\left(x_j(t)\right)$$

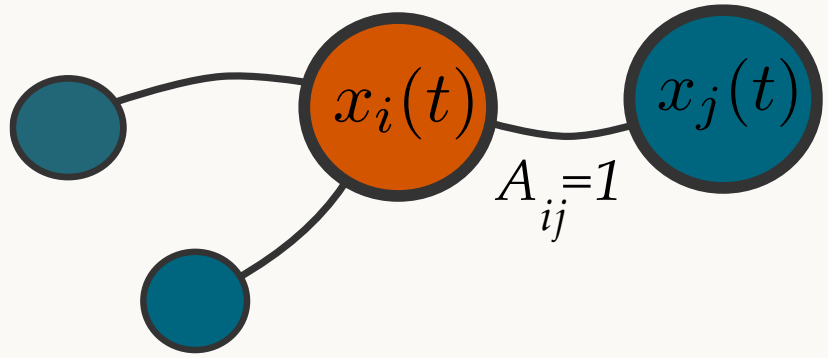


[1] *Barzel & Barabási* Nat. Phys. 9 673-681 (2013)

(Neural) ODEs on Networks

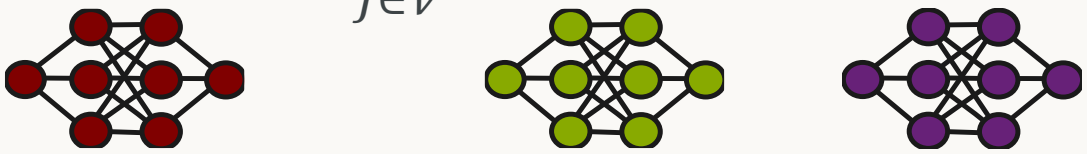
Many dynamical systems on networks are well described by [1]

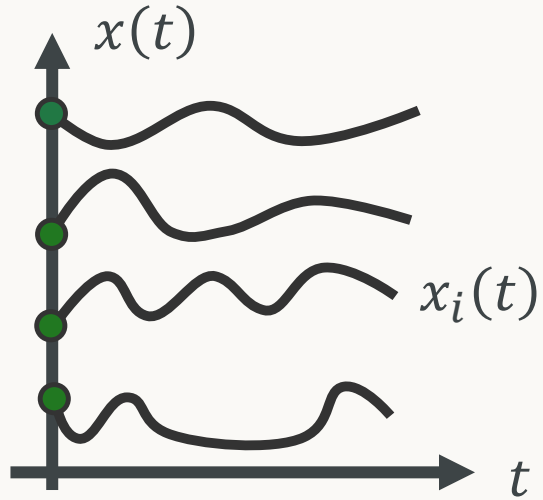
$$\frac{dx_i}{dt} = f\left(x_i(t)\right) + \sum_{j \in \mathcal{V}} A_{ij} h^{\text{ego}}\left(x_i(t)\right) h^{\text{alt}}\left(x_j(t)\right)$$



These equations form also the basis for our neural ODE architecture

$$\frac{dx_i}{dt} = f_\omega\left(x_i(t)\right) + \sum_{j \in \mathcal{V}} A_{ij}\, h_\omega^{\text{ego}}\left(x_i(t)\right) h_\omega^{\text{alt}}\left(x_j(t)\right)$$



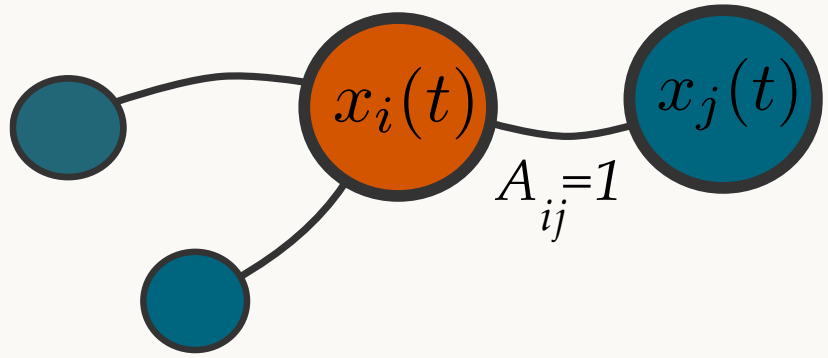


[1] *Barzel & Barabási* Nat. Phys. 9 673-681 (2013)

(Neural) ODEs on Networks

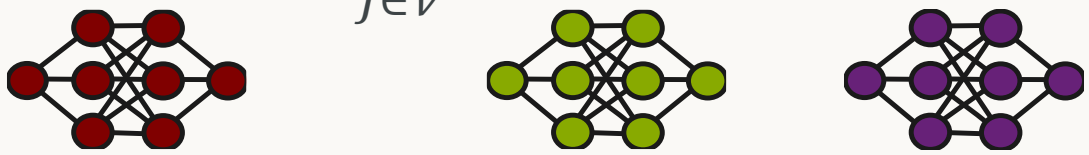
Many dynamical systems on networks are well described by [1]

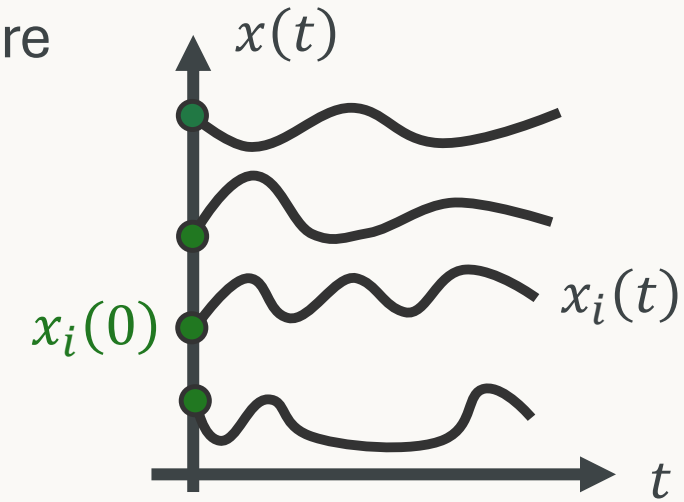
$$\frac{dx_i}{dt} = f\left(x_i(t)\right) + \sum_{j \in \mathcal{V}} A_{ij} h^{\text{ego}}\left(x_i(t)\right) h^{\text{alt}}\left(x_j(t)\right)$$



These equations form also the basis for our neural ODE architecture

$$\frac{dx_i}{dt} = f_\omega\left(x_i(t)\right) + \sum_{j \in \mathcal{V}} A_{ij}\, h_\omega^{\text{ego}}\left(x_i(t)\right) h_\omega^{\text{alt}}\left(x_j(t)\right)$$



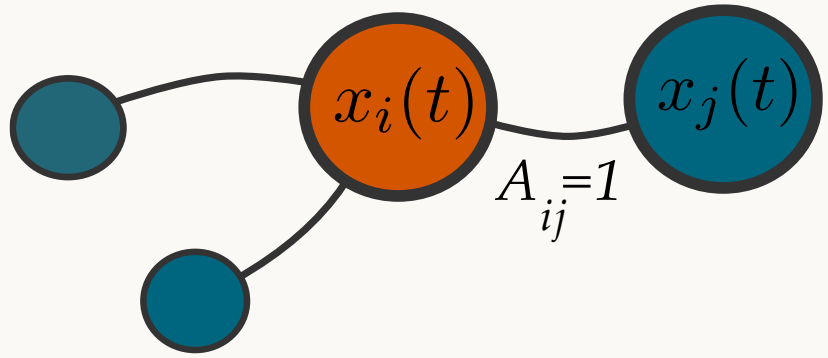


[1] *Barzel & Barabási* Nat. Phys. 9 673-681 (2013)

(Neural) ODEs on Networks

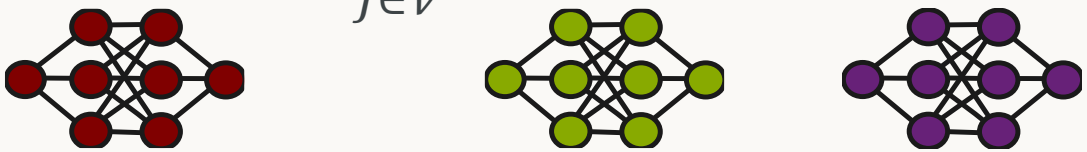
Many dynamical systems on networks are well described by [1]

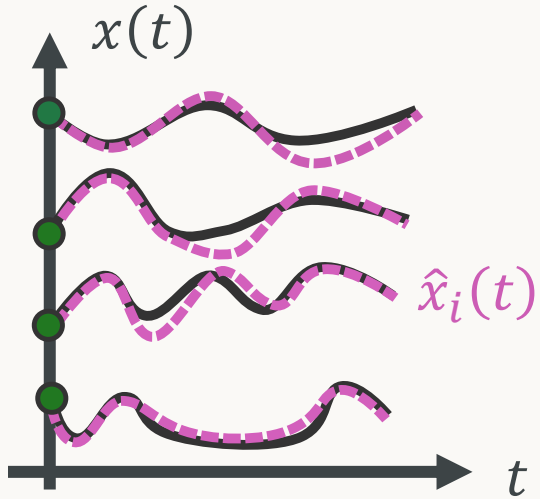
$$\frac{dx_i}{dt} = f\left(x_i(t)\right) + \sum_{j \in \mathcal{V}} A_{ij} h^{\text{ego}}\left(x_i(t)\right) h^{\text{alt}}\left(x_j(t)\right)$$



These equations form also the basis for our neural ODE architecture

$$\frac{dx_i}{dt} = f_\omega\left(x_i(t)\right) + \sum_{j \in \mathcal{V}} A_{ij}\, h_\omega^{\text{ego}}\left(x_i(t)\right) h_\omega^{\text{alt}}\left(x_j(t)\right)$$

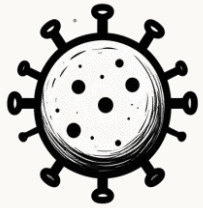




[1] *Barzel & Barabási* Nat. Phys. 9 673-681 (2013)

Setup: Dynamics

We use synthetic data from **five dynamical systems** from different domains [1,2].



Susceptible-Infected-Susceptible (SIS)

Mass-Action Kinetics (MAK)

Michaelis-Menten Model (MM)

Birth-Death Process (BD)

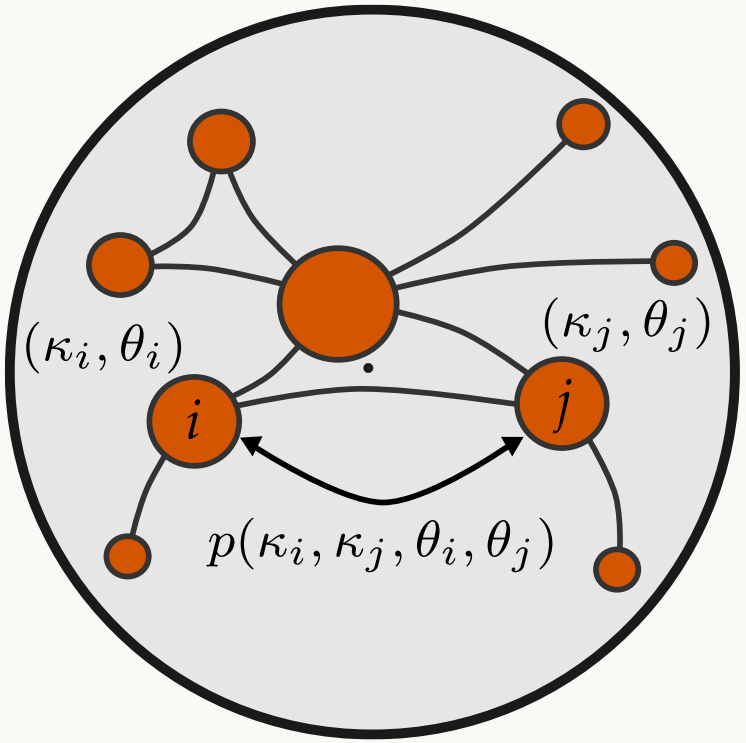
Neuronal Dynamics (ND)

[1] *Meena* et al. Nat. Phys. 19 1033 (2023) [2] *Hens et al.* Nat. Phys. 15 403-412 (2019)

Setup: Networks

We use the $\mathbb{S}^1$ model [1,2] to generate graphs with different properties:

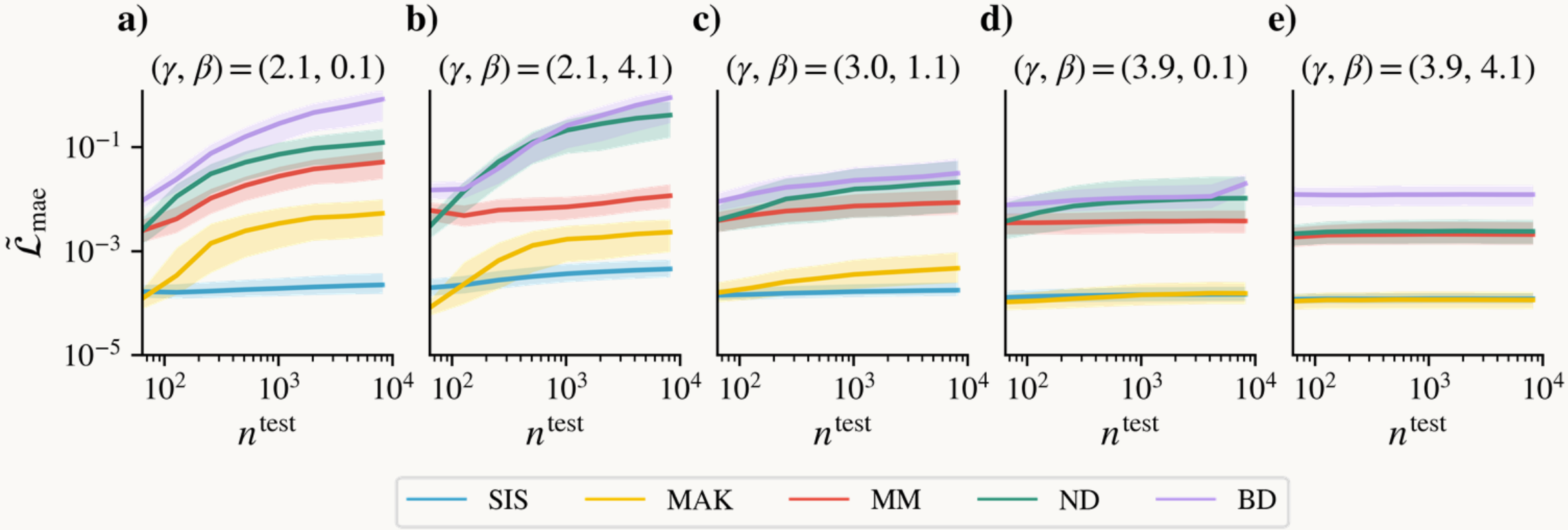
- number of nodes: $n \in \{64, \dots, 8192\}$
- average degree $\bar{k} = 10$
- exponent of the degree distribution $\gamma \in [2.1, \dots, 3.9]$.
  - **lower γ ➔ larger hubs.**
- inverse temperature $\beta \in [0.1, \dots, 4.1]$.
  - **larger β ➔ more clustering.**



[1] *Serrano* et al. PRL 100, 078701 (2008) [2] *Krioukov* et al. PRE 82 036106 (2010)

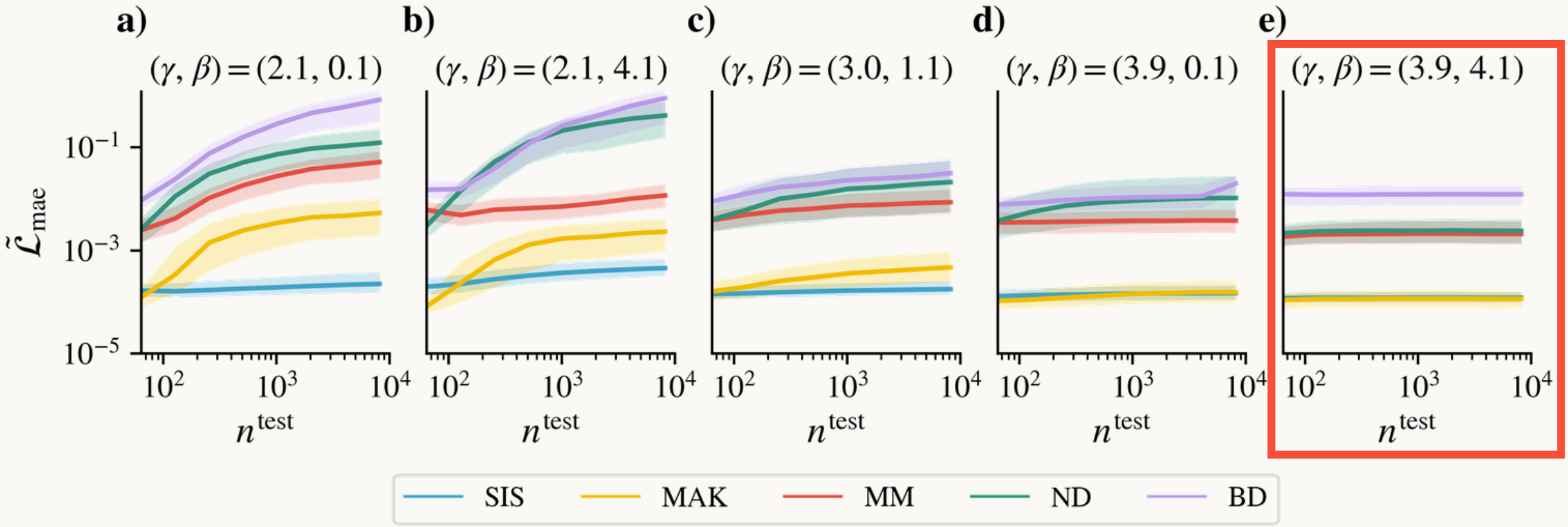
Generalization: Larger networks

Generalization to larger networks is **possible for low degree heterogeneity.**
Clustering has limited influence on size generalization.



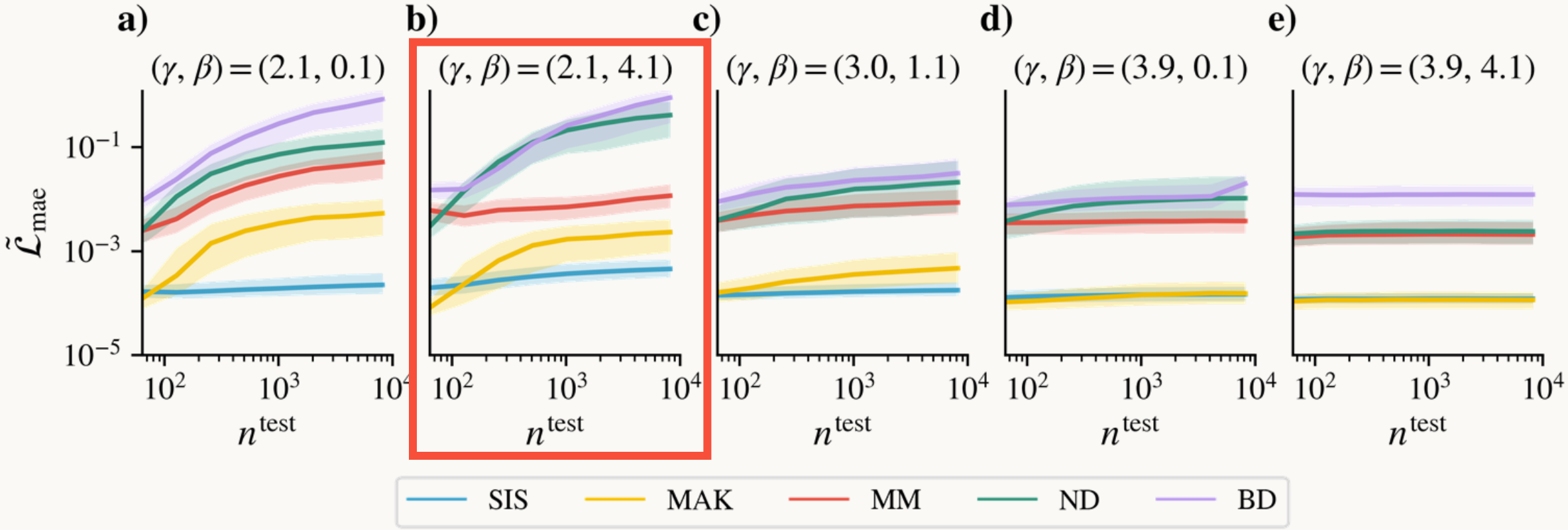
Generalization: Larger networks

Generalization to larger networks is **possible for low degree heterogeneity.**
Clustering has limited influence on size generalization.



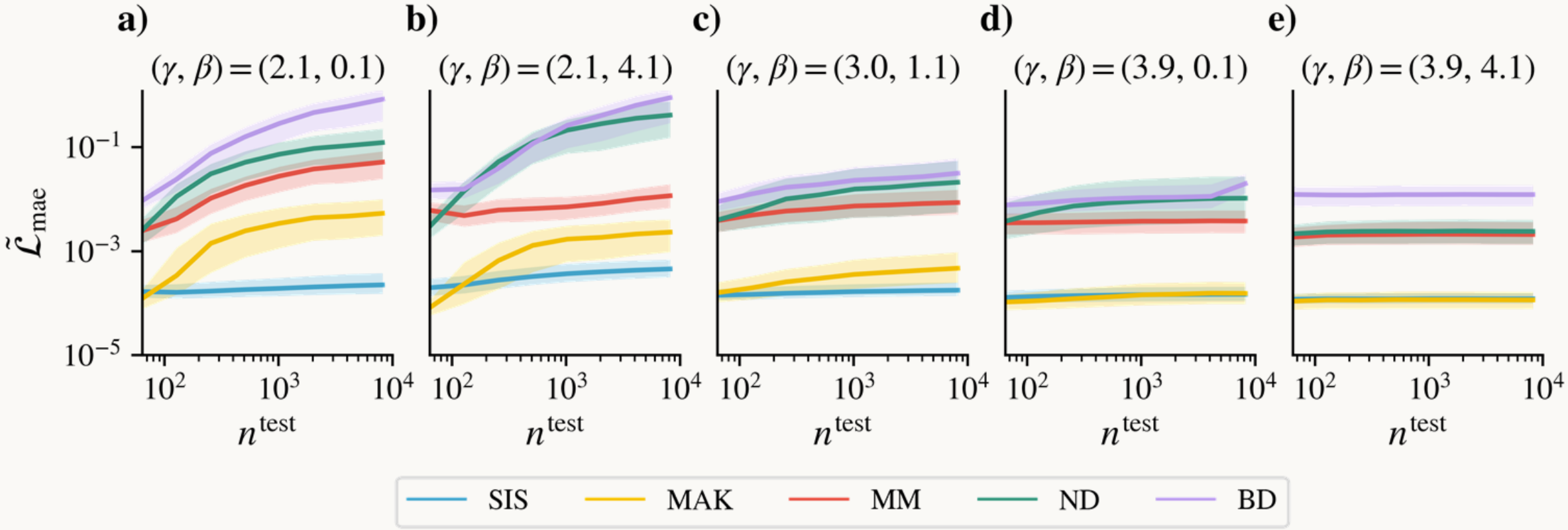
Generalization: Larger networks

Generalization to larger networks is **possible for low degree heterogeneity.**
Clustering has limited influence on size generalization.



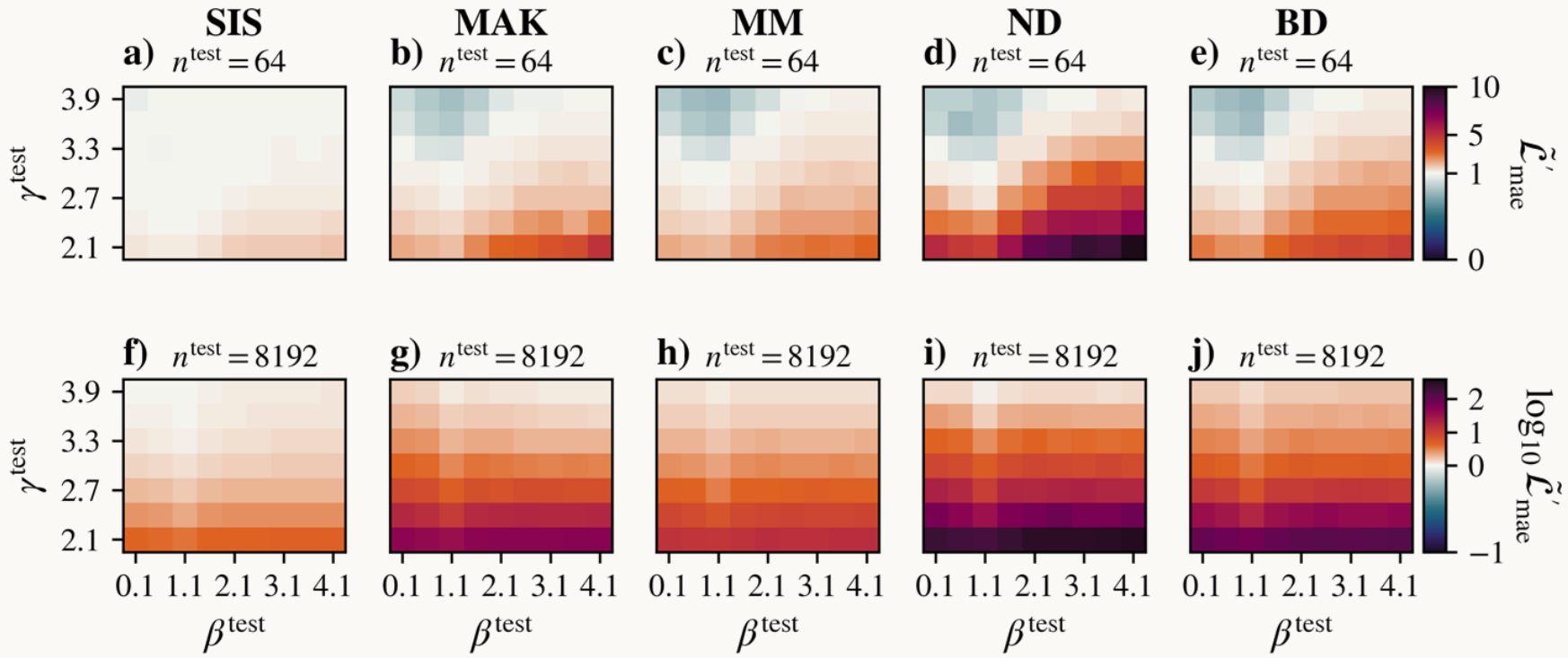
Generalization: Larger networks

Generalization to larger networks is **possible for low degree heterogeneity.**
Clustering has limited influence on size generalization.



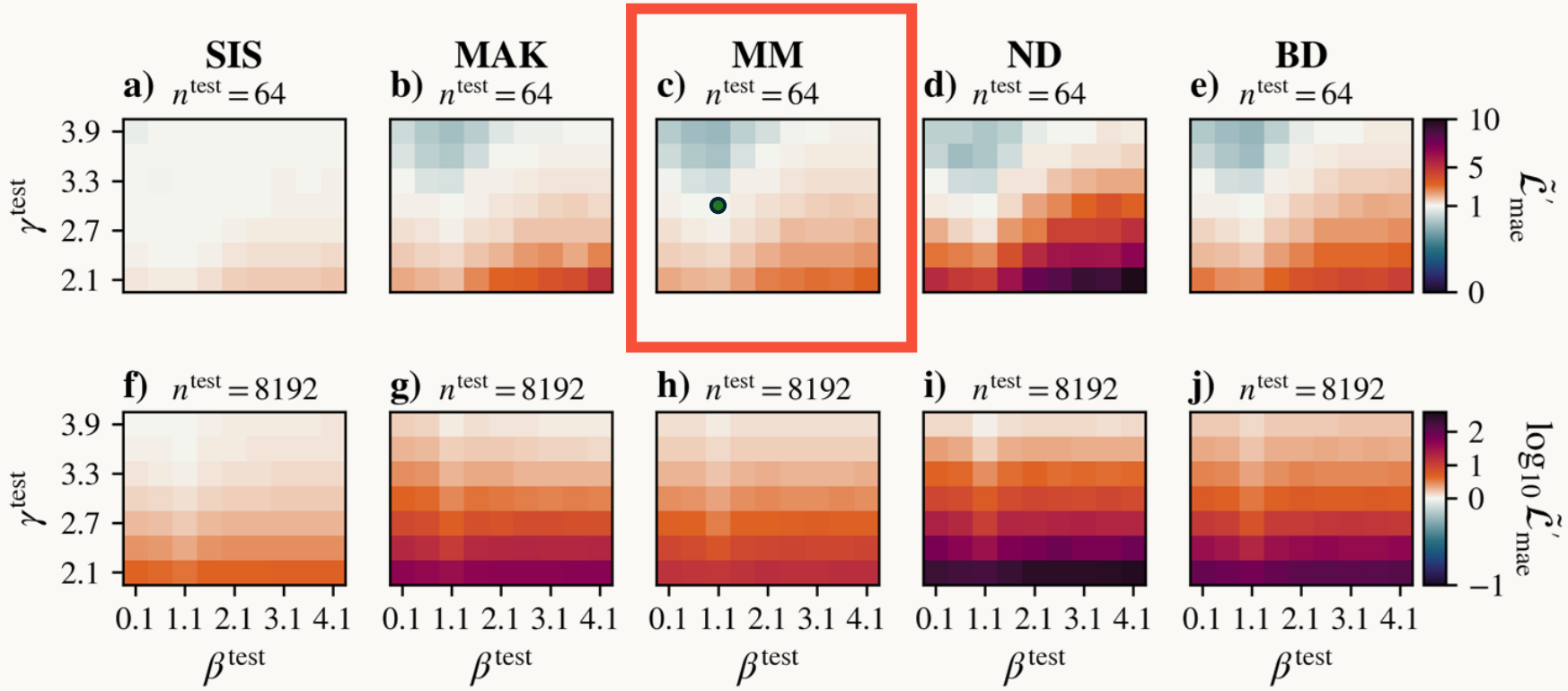
Generalization: Structurally different networks

Performance is **good on less clustered**, and **less degree heterogeneous networks**, but degrades with larger degree heterogeneity.



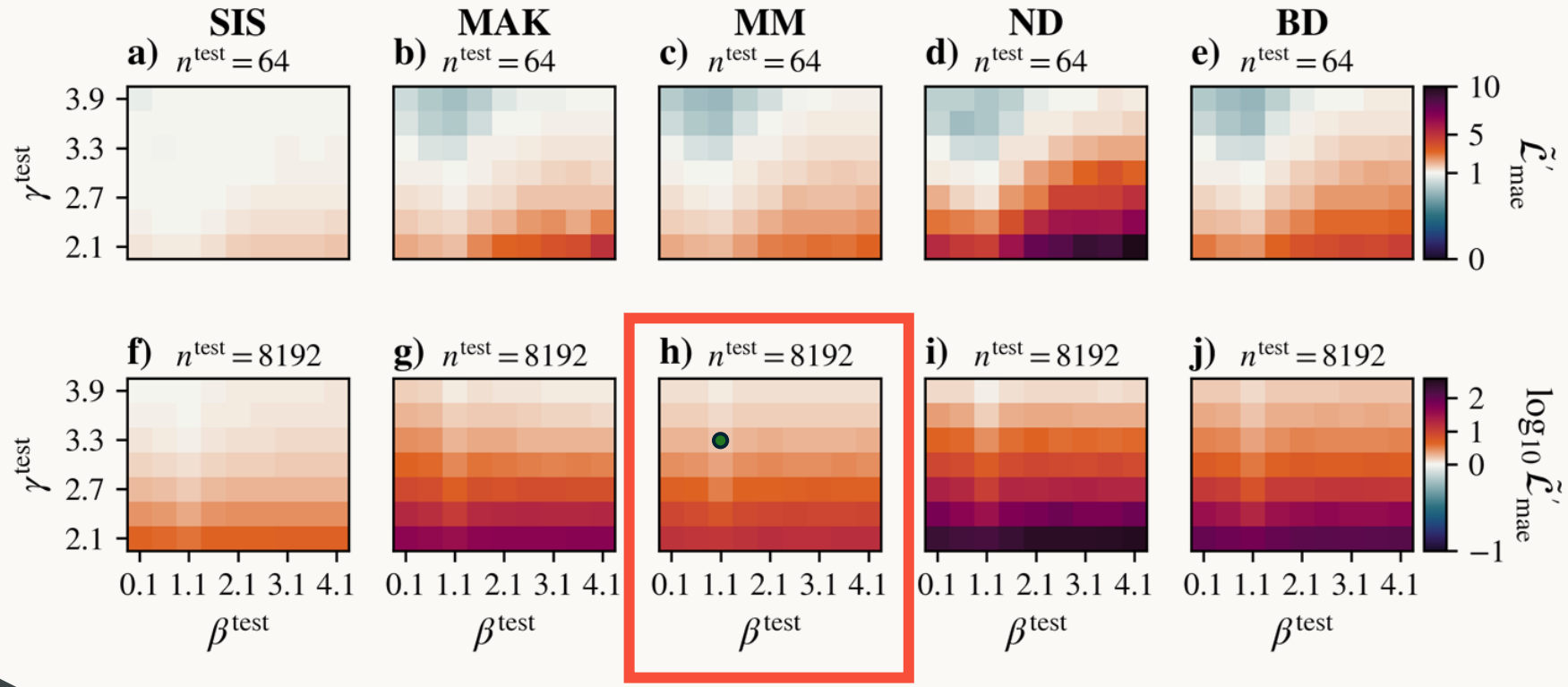
Generalization: Structurally different networks

Performance is **good on less clustered**, and **less degree heterogeneous networks**, but degrades with larger degree heterogeneity.



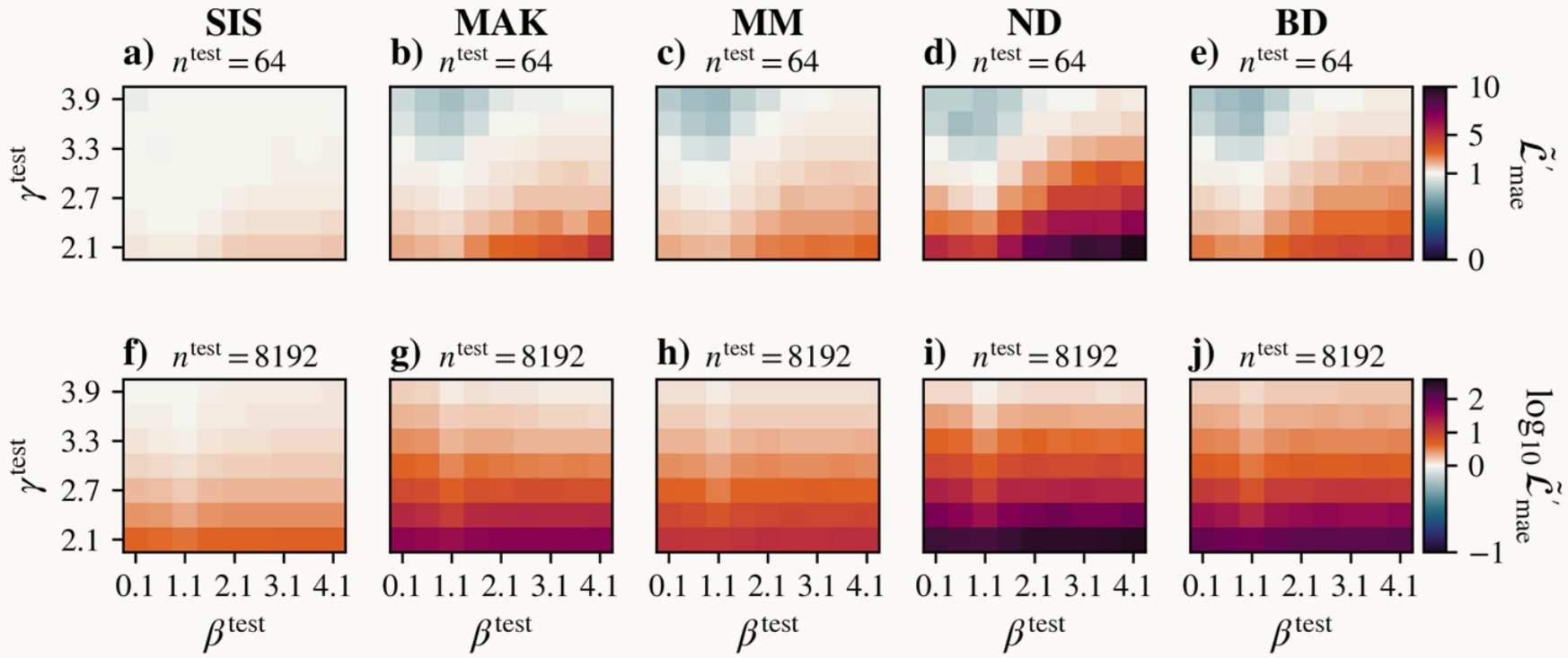
Generalization: Structurally different networks

Performance is **good on less clustered**, and **less degree heterogeneous networks**, but degrades with larger degree heterogeneity.



Generalization: Structurally different networks

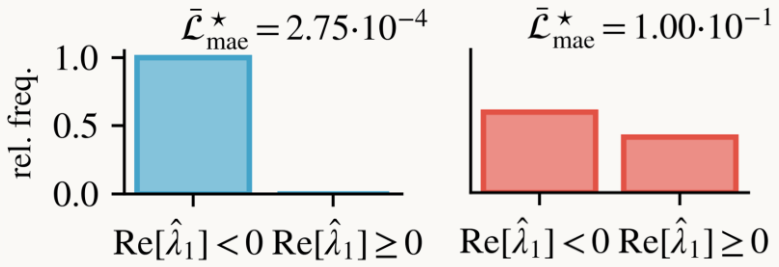
Performance is **good on less clustered**, and **less degree heterogeneous networks**, but degrades with larger degree heterogeneity.



Robustness

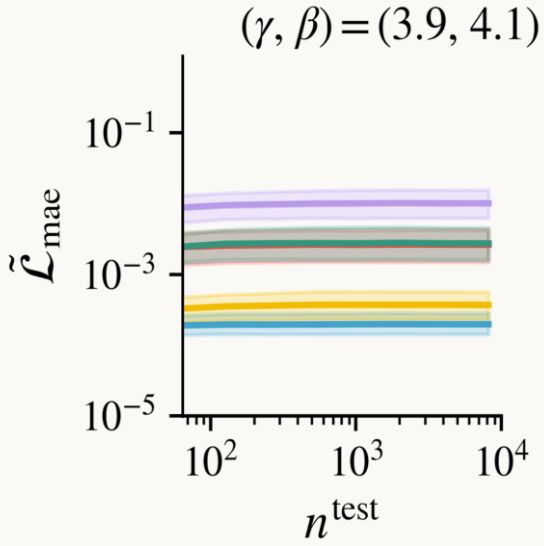
Local Stability

Well performing models **also capture stability**. Models that don't do well find wrong or even unstable fixed points.



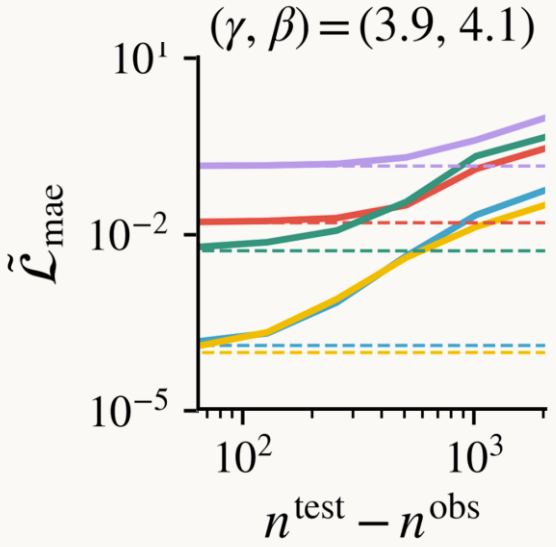
Noisy Training Data

Size generalization results are **robust to moderate noise** in the training data.



Missing Nodes

Performance degrades fast with missing nodes, even only a few are non-observed at prediction time.



Conclusion

Neural ODEs can exhibit excellent size generalization for some dynamical systems and if graphs are not too degree heterogeneous.

However, they struggle to generalize to graphs that are more degree heterogeneous than the training graph and are brittle to missing nodes at time of deployment.

Paper will appear soon!

Thanks to Melanie Weber and Tina Eliassi-Rad.

laber.m@northeastern.edu

Copyright

Laber M., Klein B., Eliassi-Rad T. *When do neural ordinary differential equations generalize on complex networks?* arXiv:2602.08980 (2026)