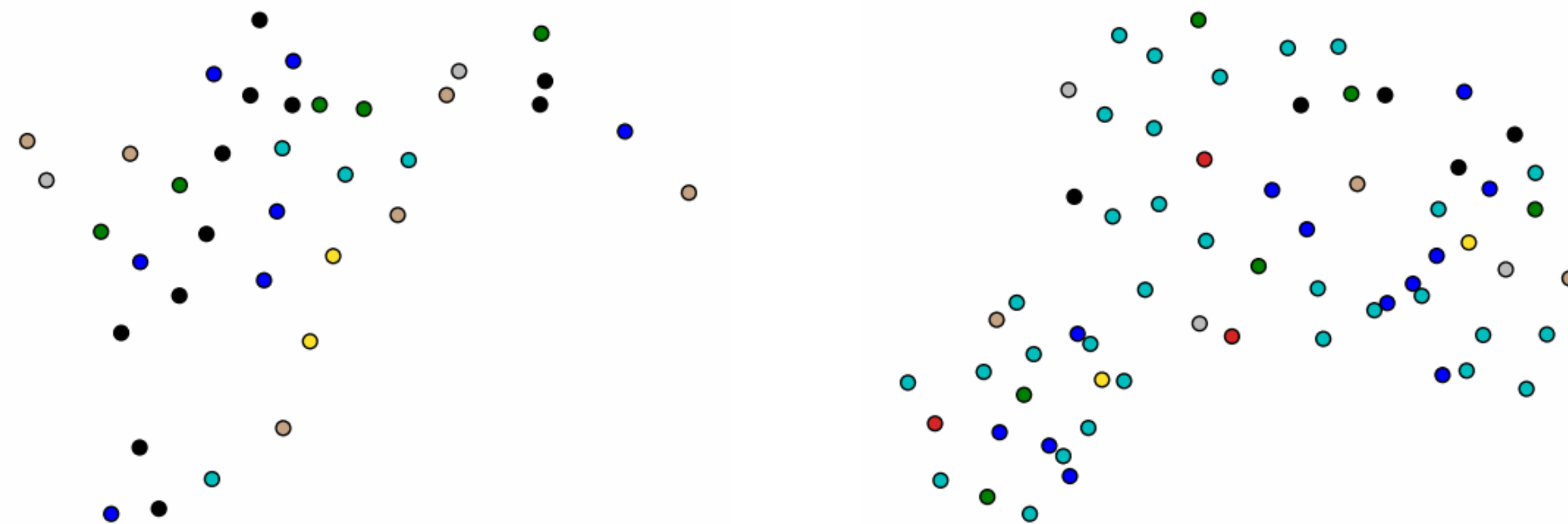
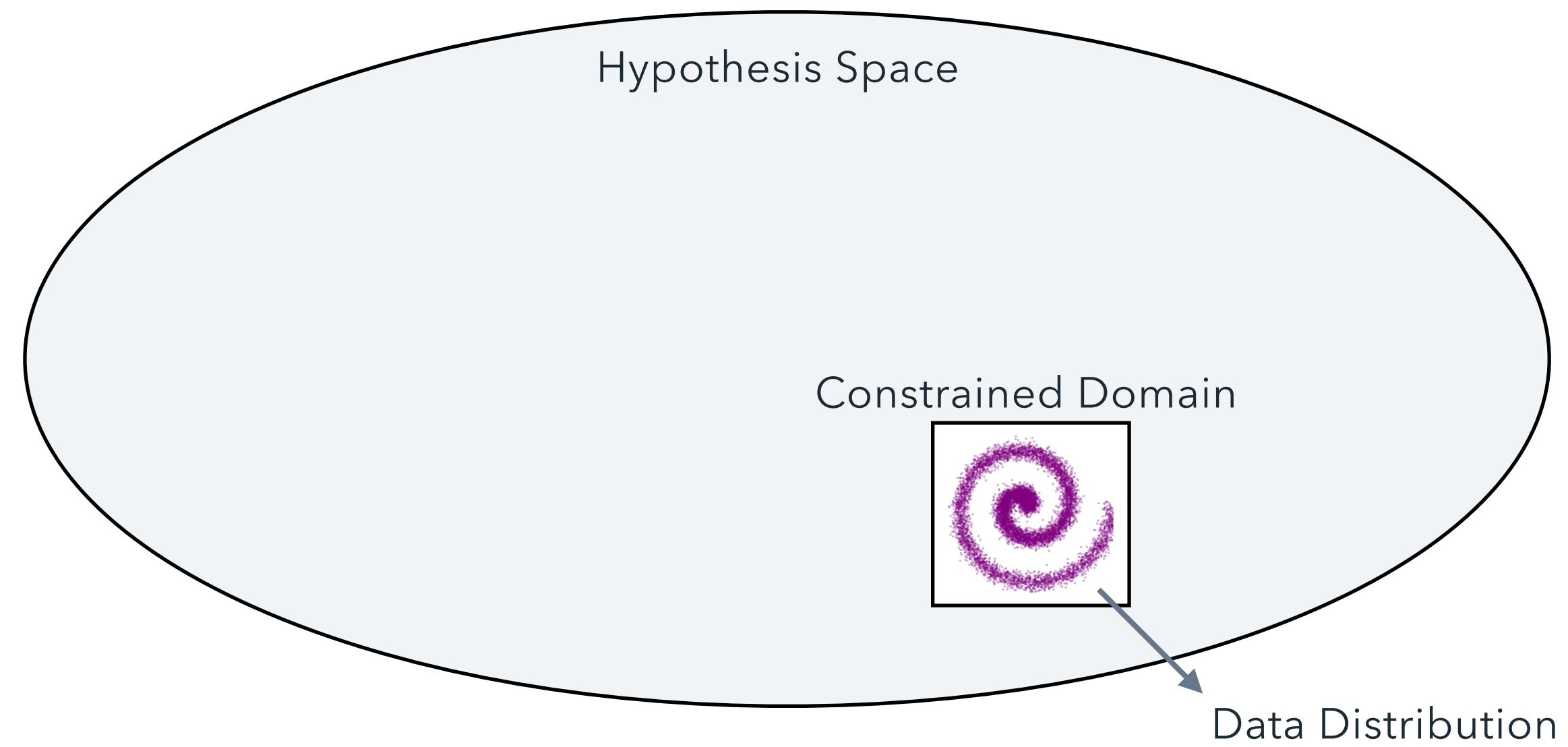


# Generative Modelling of Structurally Constrained Graphs



Manuel Madeira, Clément Vignac, Dorina Thanou & Pascal Frossard

## PROBLEM: CONSTRAINED GENERATION

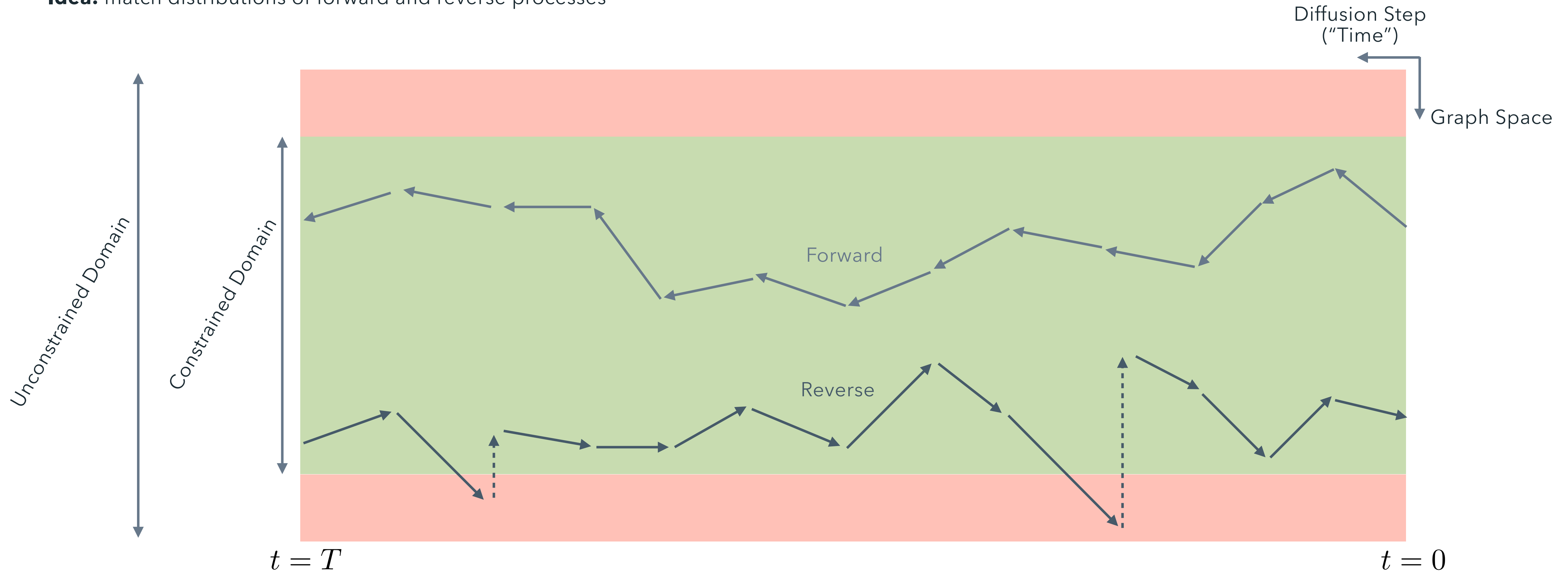


Examples:

- **Low data regimes**  
e.g. biomedical applications
- **Guarantee critical property verification:**  
e.g. real-world problems with  
| physical constraints  
| chemical  
| biological  
| social

## CONSTRUCT: IDEA

**Idea:** match distributions of forward and reverse processes



**Forward:** not learnable → design it to preserve constraint

**Reverse:** includes a **projector** to correct trajectory whenever needed

## CONSTRAINT PRESERVING FORWARD

**Edge-Deletion Invariant** Property:

If  $P(G) = \text{True}$  and  $G' = (X, E \setminus \tilde{E})$ , then  $P(G') = \text{True}$ .



Subset of edges

Examples: graph planarity, acyclicity, maximum degree, ...

**Forward:**

- Nodes: marginal noise (as DiGress)
- Edges: **absorbing** noise → **preserves edge-deletion invariance**

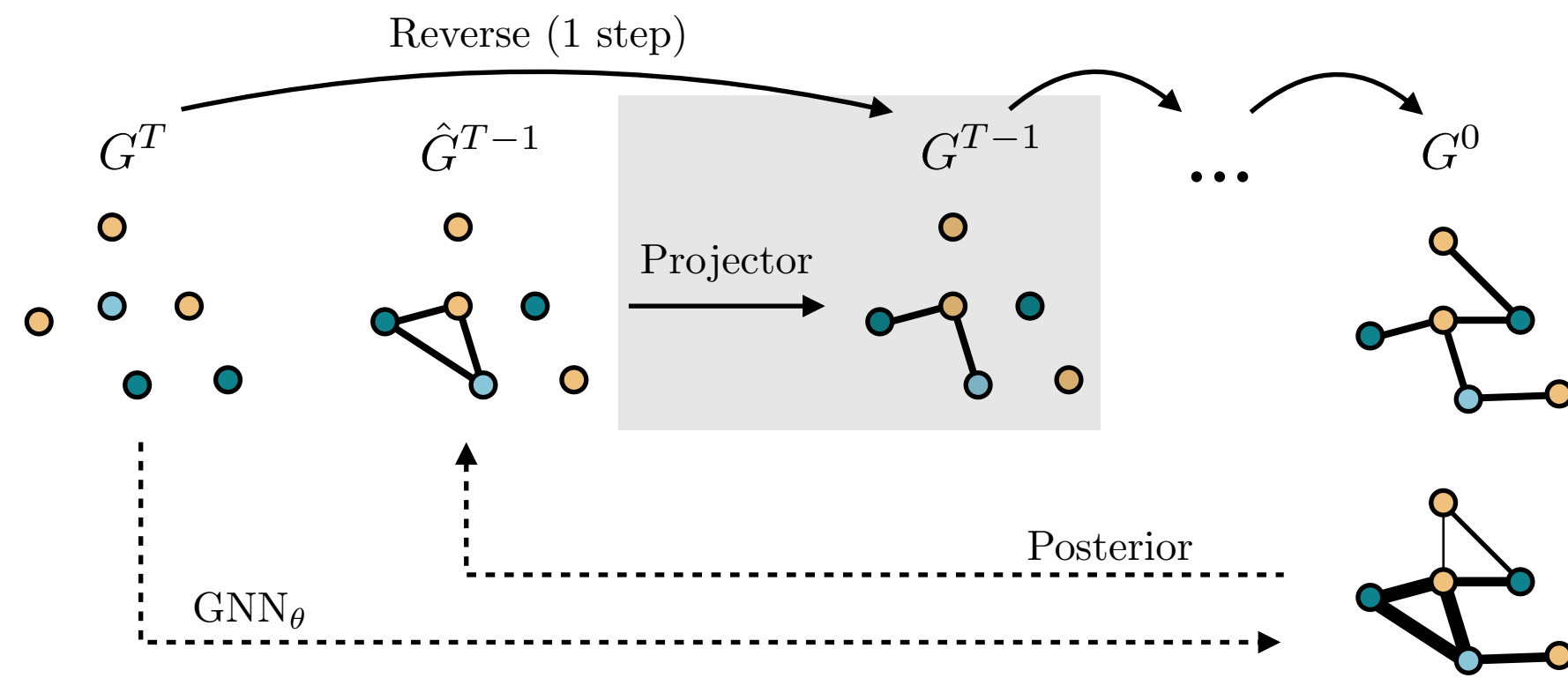
$$\mathbf{Q}_X^t = \alpha^t \mathbf{I} + (1 - \alpha^t) \mathbf{1}_b \mathbf{m}'_X$$

$$\mathbf{Q}_E^t = \alpha_{\text{ABS}}^t \mathbf{I} + (1 - \alpha_{\text{ABS}}^t) \mathbf{1}_c \mathbf{e}'_E$$



Limit: no edges!

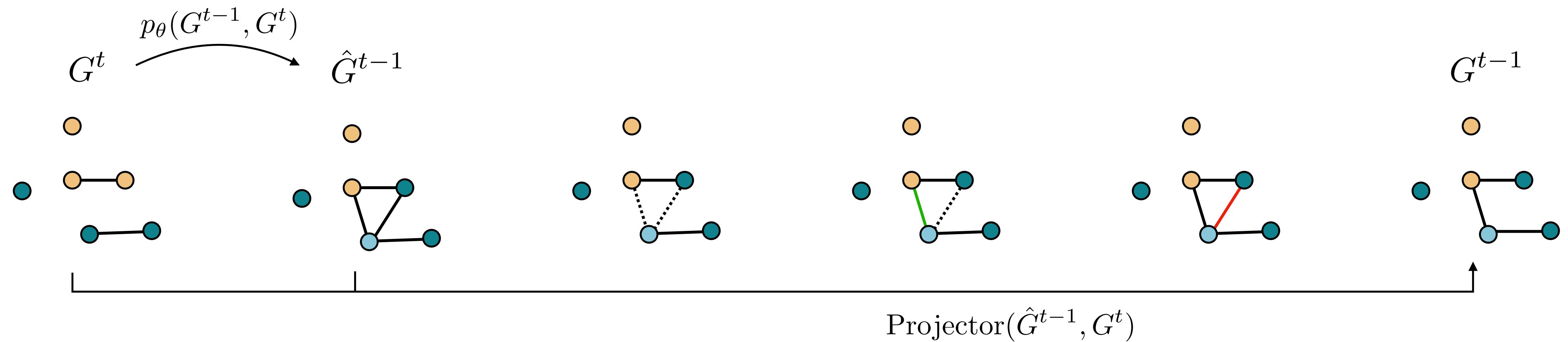
## CONTROLLED REVERSE



**Reverse:** Directly "project"  $\hat{G}^{t-1}$  can be arbitrarily hard (e.g. maximum planar subgraph is NP-hard)

**Idea:** use  $G^t$ , which verifies property, as (good) initialisation point

## PROJECTOR



**Nodes:** directly accept the candidate

**Edges:** iteratively and randomly add candidate edges that preserve property

### Efficiency of Projector

- Only affects sampling
- **Blocking edge hash table:** only perform edge check once
- Reverse is edge addition process - can leverage **incremental property checkers**
  - e.g. planarity:  $O(n) \rightarrow O(\alpha(q, n))$

## RESULTS

### Synthetic Datasets: state-of-the-art performance

Model	Planar Dataset										
	Deg. ↓	Clus. ↓	Orbit ↓	Spec. ↓	Wavelet ↓	Ratio ↓	Valid ↑	Unique ↑	Novel ↑	V.U.N. ↑	Property ↑
Train set	0.0002	0.0310	0.0005	0.0038	0.0012	1.0	100	100	0.0	0.0	100
GraphRNN (You et al., 2018)	0.0049	0.2779	1.2543	0.0459	0.1034	527.4	0.0	100	100	0.0	—
GRAN (Liao et al., 2019)	0.0007	0.0426	0.0009	0.0075	0.0019	1.9	97.5	85.0	2.5	0.0	—
SPECTRE (Martinkus et al., 2022)	0.0005	0.0785	0.0012	0.0112	0.0059	2.9	25.0	100	100	25.0	—
DiGress (Vignac et al., 2022)	0.0007	0.0780	0.0079	0.0098	0.0031	5.2	77.5	100	100	77.5	—
EDGE (Chen et al., 2023)	0.0998	0.2672	1.0160	0.0998	0.3888	576.5	0.0	100	100	0.0	—
BwR (EDP-GNN) (Diamant et al., 2023)	0.0231	0.2596	0.5473	0.0444	0.1314	267.3	0.0	100	100	0.0	—
BiGG (Dai et al., 2020)	0.0007	0.0570	0.0367	0.0105	0.0052	17.0	62.5	85.0	42.5	5.0	—
HSpectre (one-shot) (Bergmeister et al., 2023)	0.0003	0.0245	0.0006	0.0104	0.0030	1.6	67.5	100	100	67.5	—
HSpectre (Bergmeister et al., 2023)	0.0005	0.0626	0.0017	0.0075	0.0013	2.1	95.0	100	100	95.0	—
DiGress+	0.0008 ±0.0001	0.0410 ±0.0033	0.0048 ±0.0004	0.0056 ±0.0004	0.0020 ±0.0002	3.6 ±0.2	76.4 ±1.3	100.0 ±0.0	100.0 ±0.0	76.4 ±1.3	76.4 ±1.3
ConStruct	0.0003 ±0.0001	0.0403 ±0.0047	0.0004 ±0.0001	0.0053 ±0.0004	0.0009 ±0.0001	1.1 ±0.1	100.0 ±0.0	100.0 ±0.0	100.0 ±0.0	100.0 ±0.0	100.0 ±0.0

### Digital Pathology: beats unconstrained models by up to ~70 percentage points

Model	Low TLS Dataset										
	Ratio ↓	Conn. ↑	Planar ↑	V.U.N. ↑	$\kappa(0)$ ↓	$\kappa(1)$ ↓	$\kappa(2)$ ↓	$\kappa(3)$ ↓	$\kappa(4)$ ↓	$\kappa(5)$ ↓	TLS Valid ↑
Train set	1.0	100	100	0.0	0.6928	0.0000	0.0000	0.0000	0.0000	0.0000	100
Baseline	194.6 ±3.0	50.3 ±0.7	10.0 ±0.5	0.0 ±0.0	0.6256 ±0.0228	0.2350 ±0.0000	0.2350 ±0.0000	0.0470 ±0.0470	0.0000 ±0.0000	0.0000 ±0.0000	0.0 ±0.0
DiGress+	4.9 ±1.0	96.0 ±0.7	19.8 ±1.8	18.6 ±1.8	0.7306 ±0.0371	0.1410 ±0.0576	0.0000 ±0.0000	0.0000 ±0.0000	0.0000 ±0.0000	0.0000 ±0.0000	18.6 ±1.8
ConStruct	4.4 ±0.3	98.4 ±0.8	100.0 ±0.0	98.4 ±0.8	0.6781 ±0.0795	0.2350 ±0.0000	0.0940 ±0.0576	0.0000 ±0.0000	0.0000 ±0.0000	0.0000 ±0.0000	96.2 ±0.7
High TLS Dataset											
Train set	1.0	100	100	0.0	0.4257	0.4512	0.4745	0.6395	0.7770	0.7663	100
Baseline	353.7 ±2.2	52.2 ±0.4	3.7 ±0.1	0.2 ±0.2	0.3273 ±0.0017	0.3414 ±0.0067	0.3641 ±0.0134	0.5184 ±0.0214	0.6398 ±0.0205	0.6957 ±0.0126	0.0 ±0.0
DiGress+	10.5 ±0.6	97.8 ±0.8	8.4 ±1.1	7.8 ±1.2	0.3194 ±0.0034	0.3308 ±0.0041	0.3598 ±0.0096	0.4878 ±0.0155	0.6234 ±0.0305	0.6887 ±0.0250	6.6 ±0.9
ConStruct	6.4 ±0.6	99.8 ±0.2	100.0 ±0.0	99.8 ±0.2	0.3378 ±0.0048	0.3437 ±0.0104	0.3799 ±0.0112	0.5306 ±0.0150	0.6360 ±0.0177	0.6798 ±0.0436	88.0 ±0.5

**Other results:** using ConStruct for controlled molecular generation, likelihood-based variants, etc.

## CONCLUSION

- ConStruct **guarantees structural constraint satisfaction** in graph discrete diffusion models
- ConStruct allows for **domain knowledge integration** into the generative process → State-of-the-art performance