Workshop on Diffusion Models for Molecule Generation

Chin-Wei Huang, Victor Garcia Satorras

AI4Science - Microsoft Research Amsterdam

Contents

Equivariant Diffusion

For Molecule Generation in 3D

Emiel Hoogeboom*, Victor Garcia Satorras*, Clément Vignac* & Max Welling

***Equal contribution**

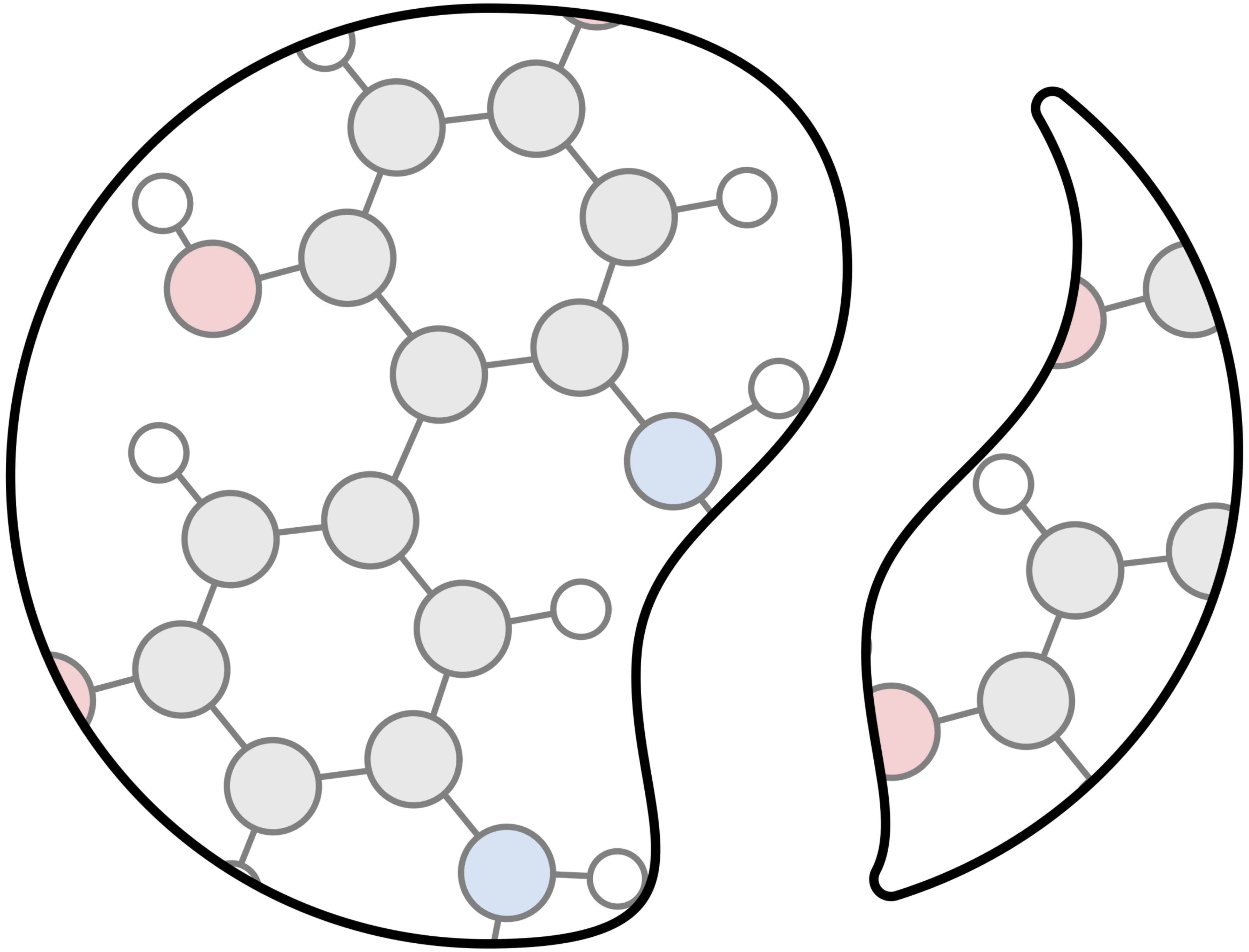
UNIVERSITY OF AMSTERDAM

EPFL

Work done while at University of Amsterdam – Delta Bosch Lab

Why Generate Molecules in 3D?

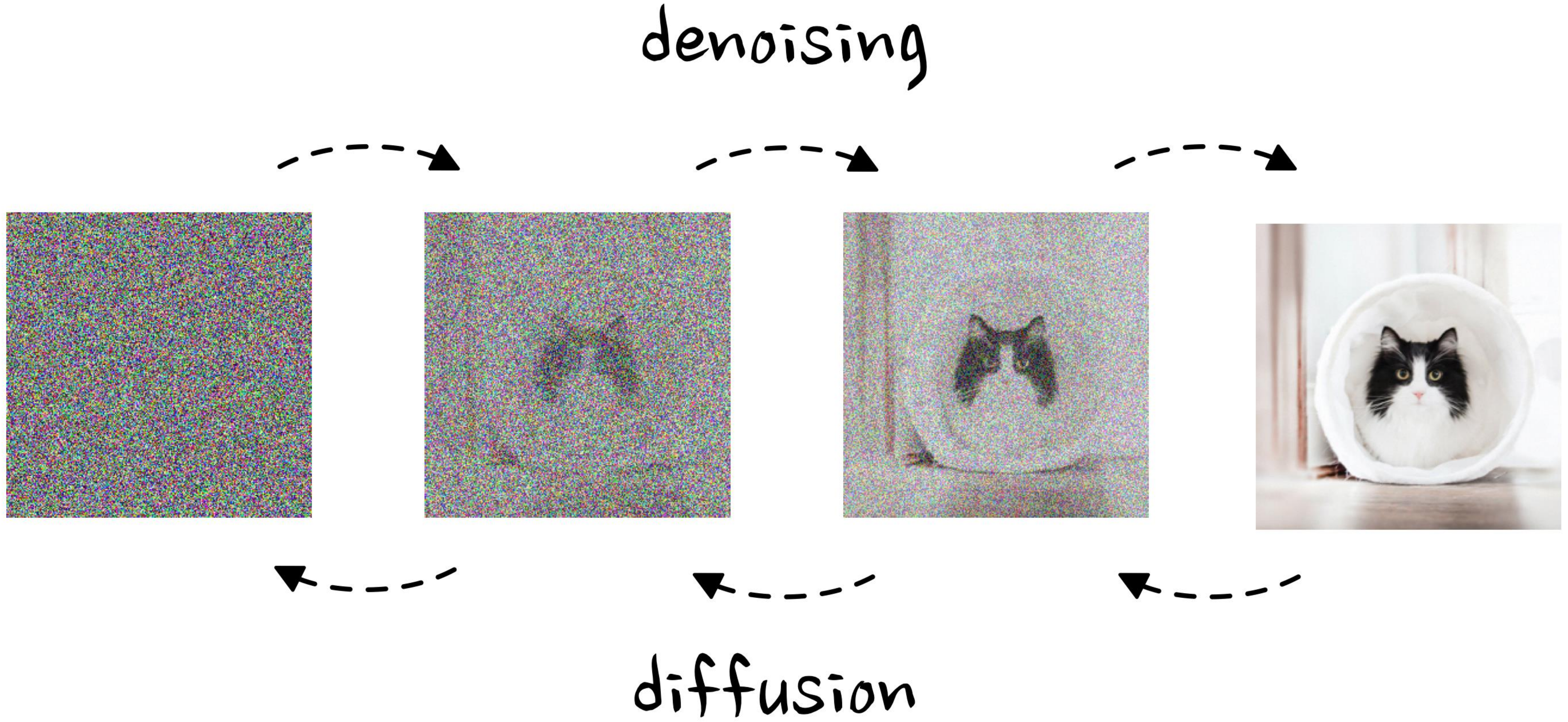
- Drug Discovery
- Catalyst Design
- Material Discovery
- Docking Problems



Denoising Diffusion

Background

- Define a diffusion process that destroys signal towards a standard normal
- Learn denoising process to generate



Denoising Diffusion

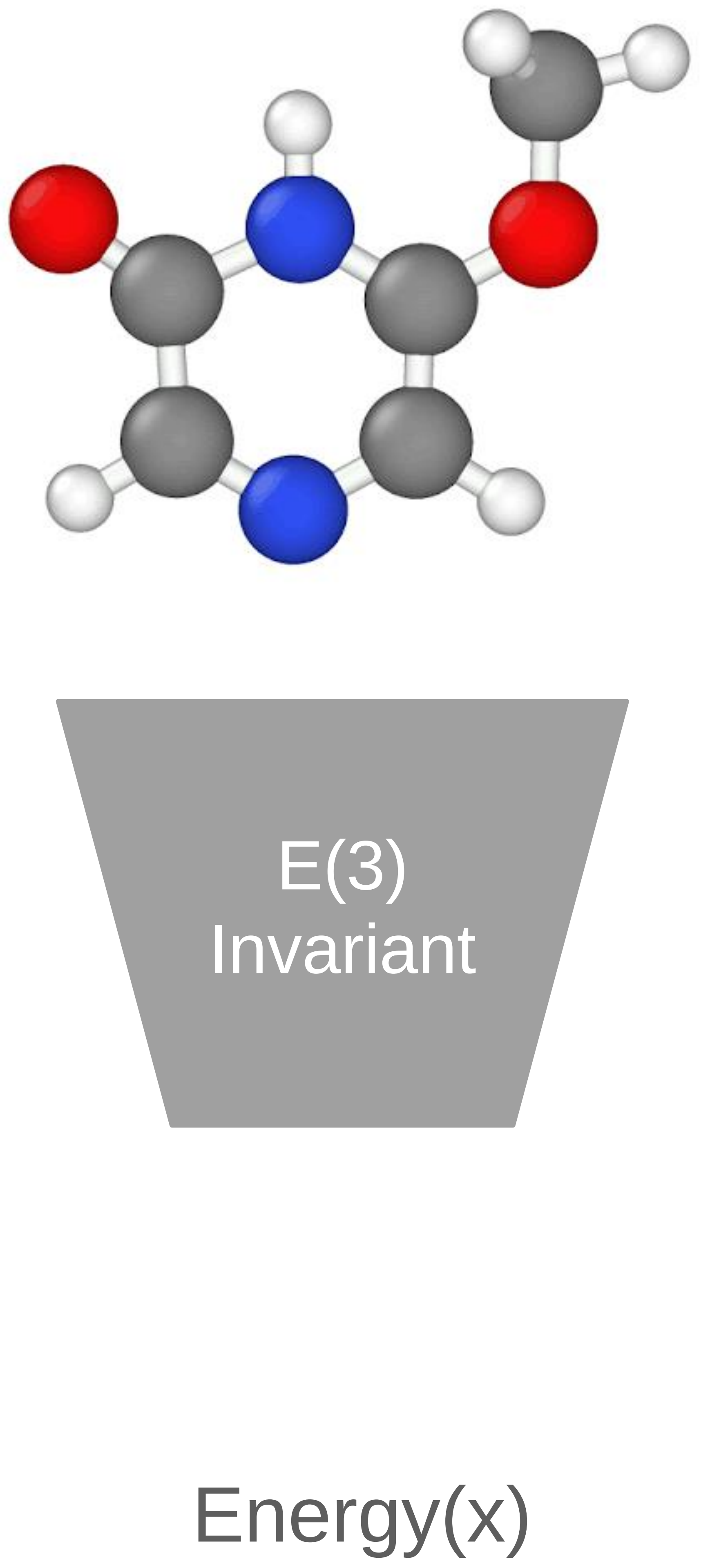
Background

DALL-E (OpenAI)

Stable Diffusion (Stability Ai)

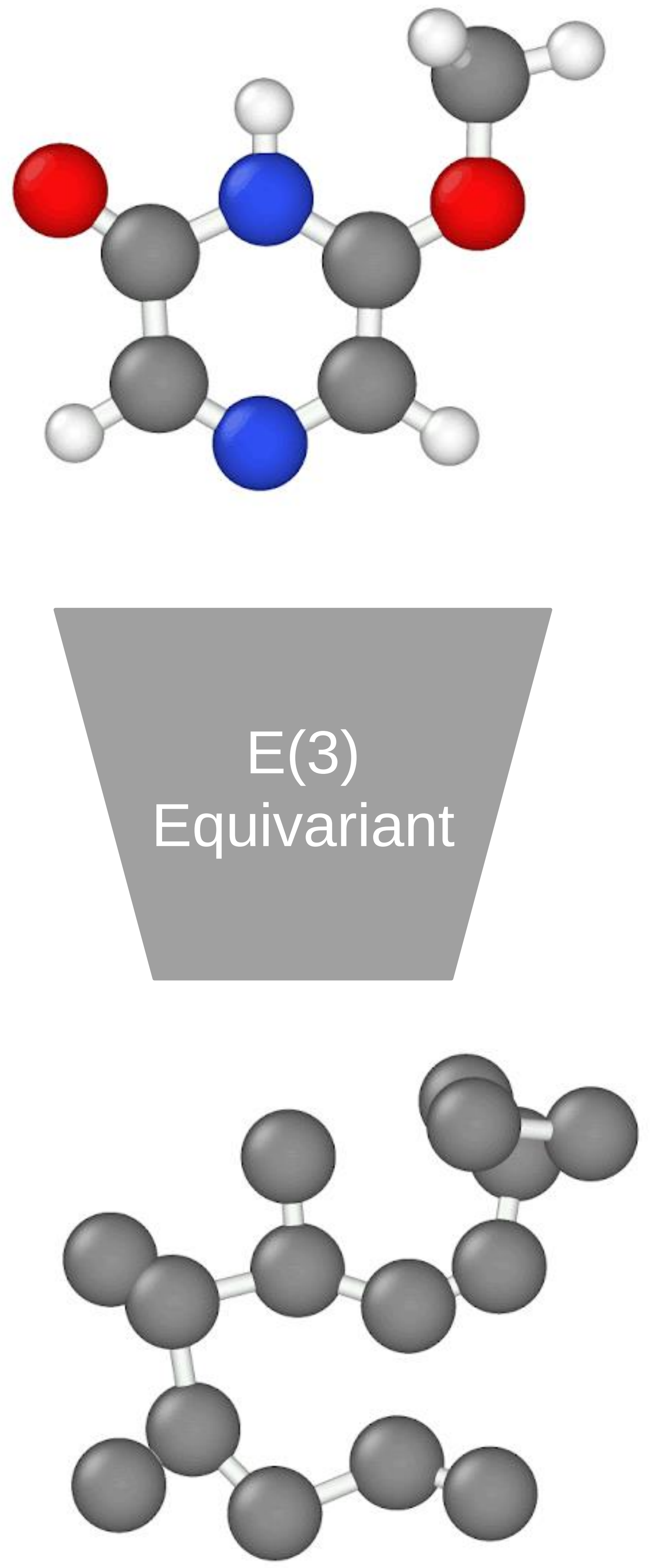
Equivariance / Invariance

$$f(\boldsymbol{x}) = f(\mathbf{R}\boldsymbol{x})$$



Equivariance / Invariance

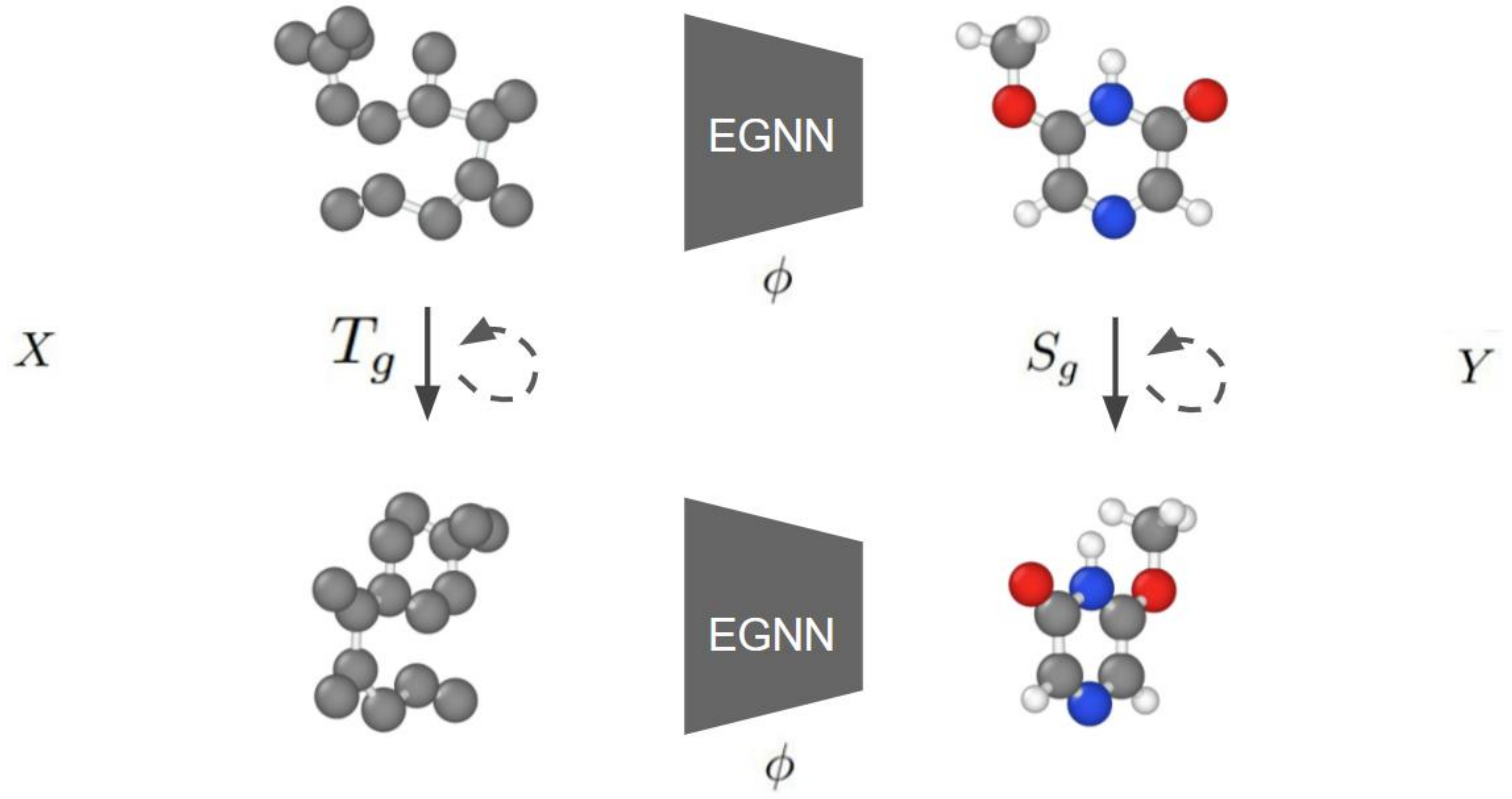
$$\mathbf{R}f(\boldsymbol{x}) = f(\mathbf{R}\boldsymbol{x})$$



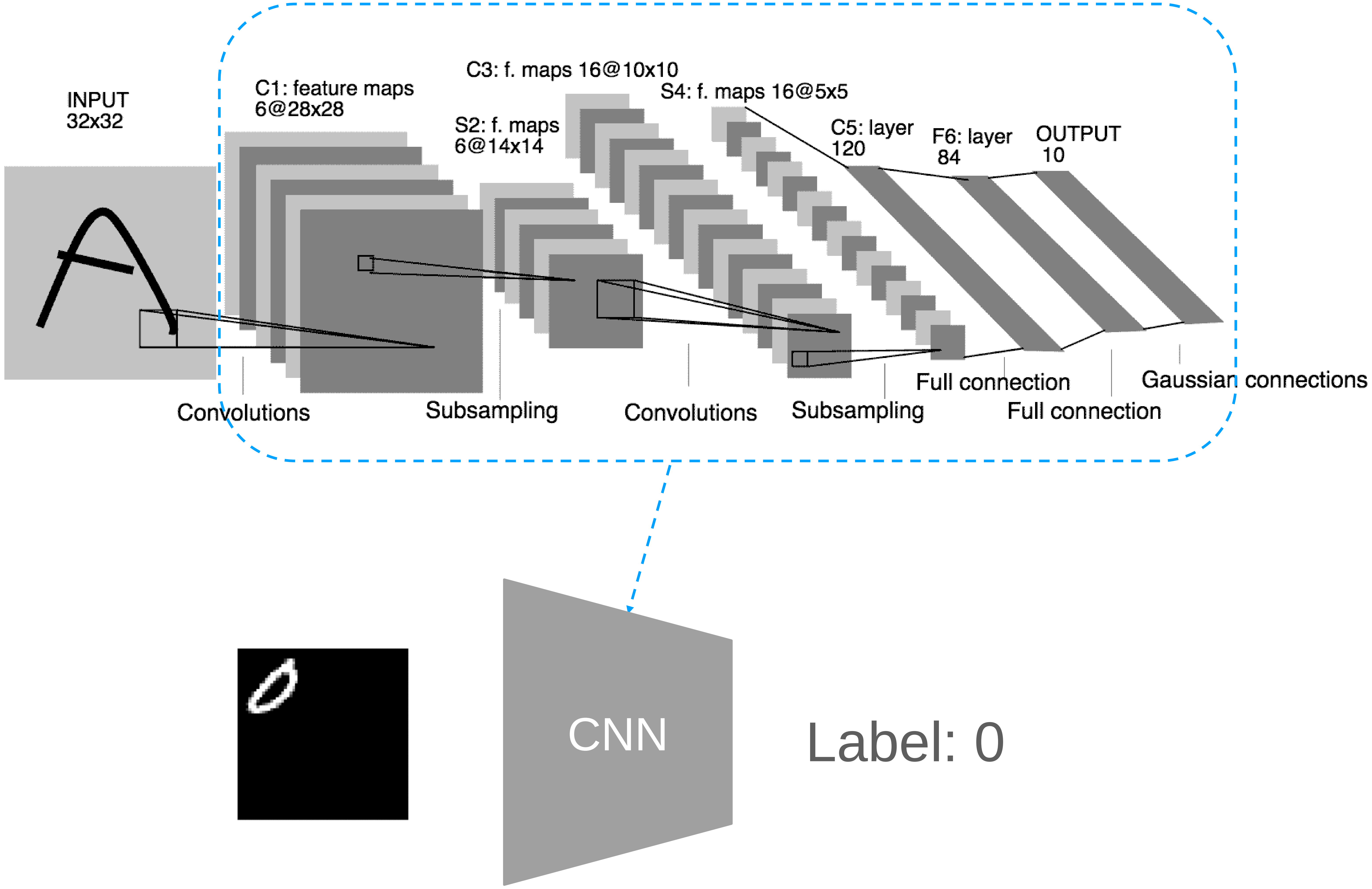
Equivariance / Invariance

Let $T_g : X \to X$ be a set of transformations on X for the abstract group $g \in G$. We say a function $\phi : X \to Y$ is equivariant to g if there exists an equivalent transformation on its output space $S_g : Y \to Y$ such that:

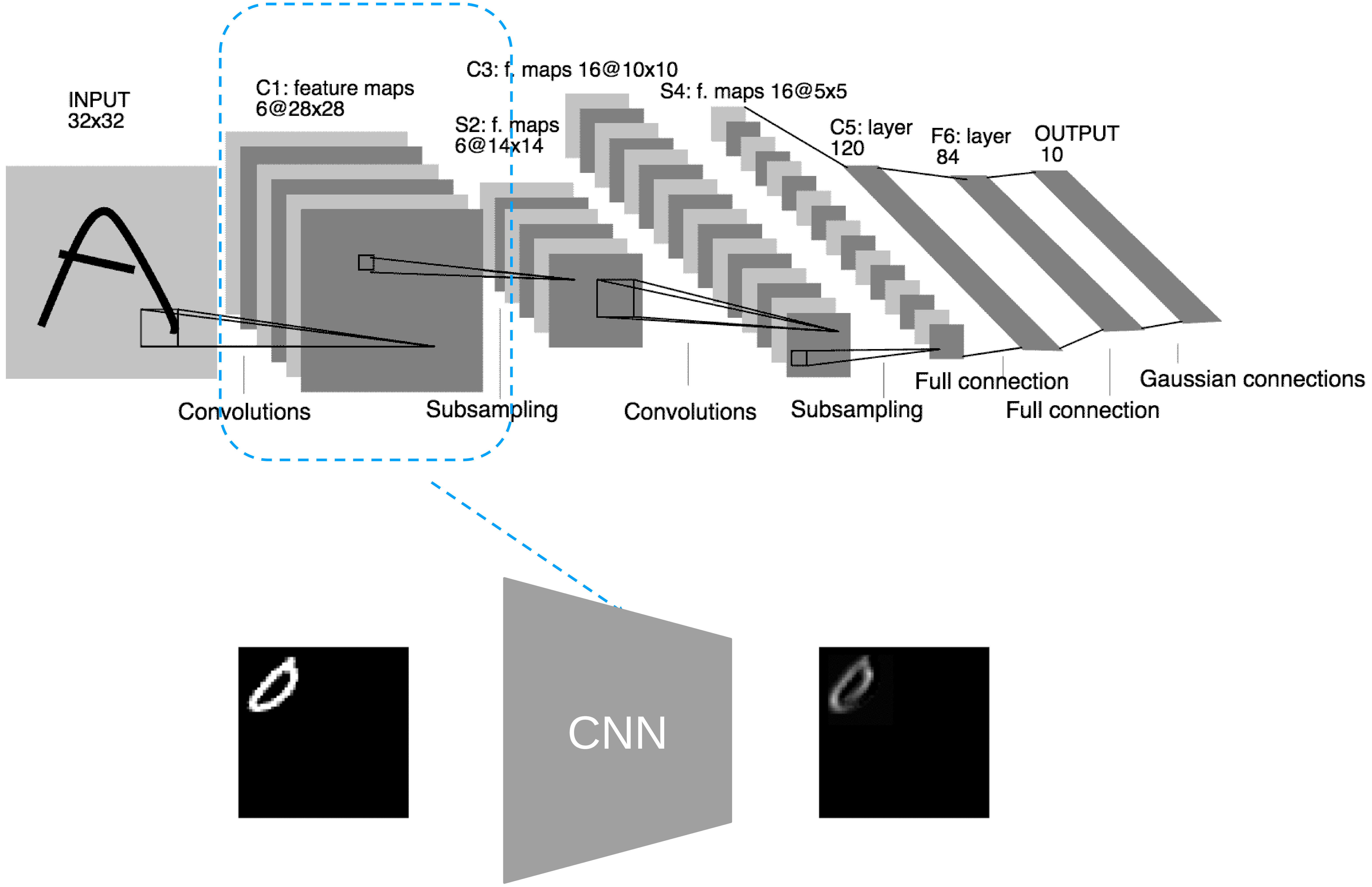
$$\phi(T_g(\mathbf{x})) = S_g(\phi(\mathbf{x}))$$



Equivariance / Invariance



Equivariance / Invariance



Equivariance / Invariance

In molecular modelling we are interested in equivariance w.r.t.:

- Translations 3D
- Rotations 3D (possibly reflections)
- Permutations

Equivariance / Invariance

In molecular modelling we are interested in equivariance w.r.t.:

- Translations 3D
- Rotations 3D (possibly reflections)
- Permutations

E(3)
Equivariance

Equivariance / Invariance

In molecular modelling we are interested in equivariance w.r.t.:

- Translations 3D
- Rotations 3D (possibly reflections)
- Permutations → Graph Neural Networks

E(3)
Equivariance

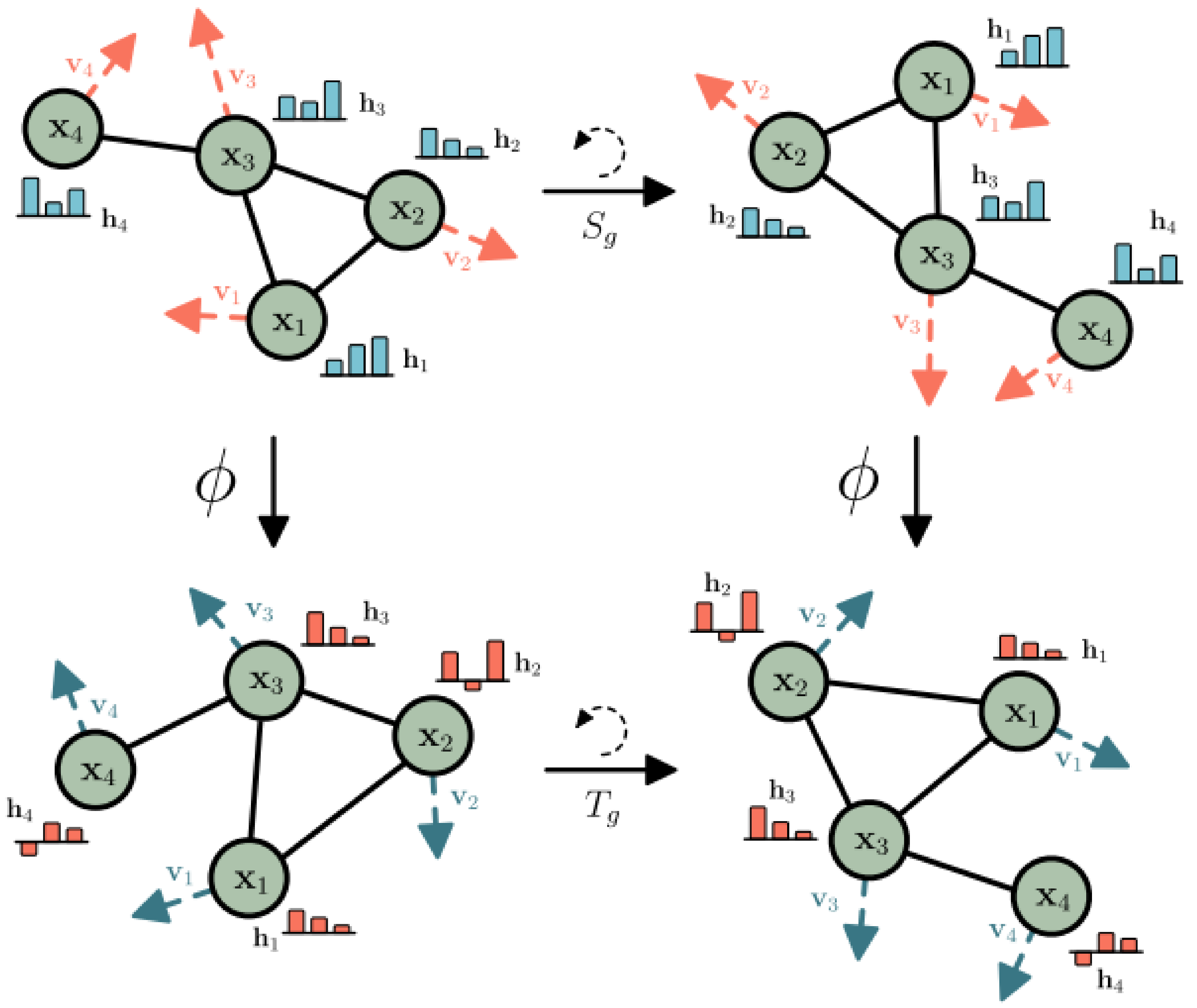
E(n) Equivariant Graph Neural Networks

E(n) Equivariant Graph Neural Networks

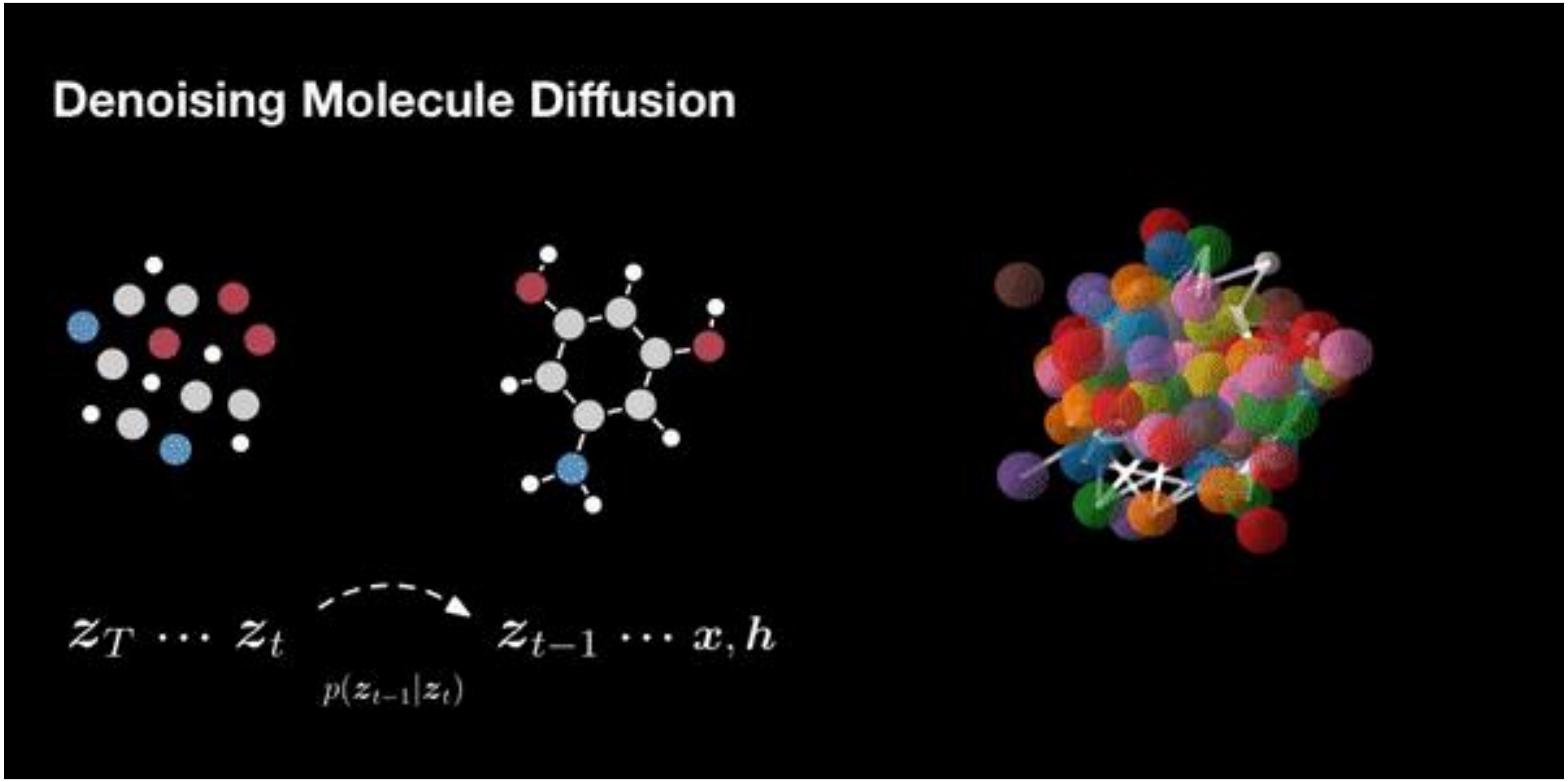
Victor Garcia Satorras [1] Emiel Hoogeboom [1] Max Welling [1]

Abstract

This paper introduces a new model to learn graph neural networks equivariant to rotations, translations, reflections and permutations called E(n)-Equivariant Graph Neural Networks (EGNNs). In contrast with existing methods, our work does not require computationally expensive higher-order representations in intermediate layers while it still achieves competitive or better performance. In addition, whereas existing methods are limited to equivariance on 3 dimensional spaces, our model is easily scaled to higher-dimensional spaces. We demonstrate the effectiveness of our method on dynamical systems modelling, representation learning in graph autoencoders and predicting molecular properties.



Equivariant Diffusion for Molecule Generation in 3D



Closely Related Work

Background

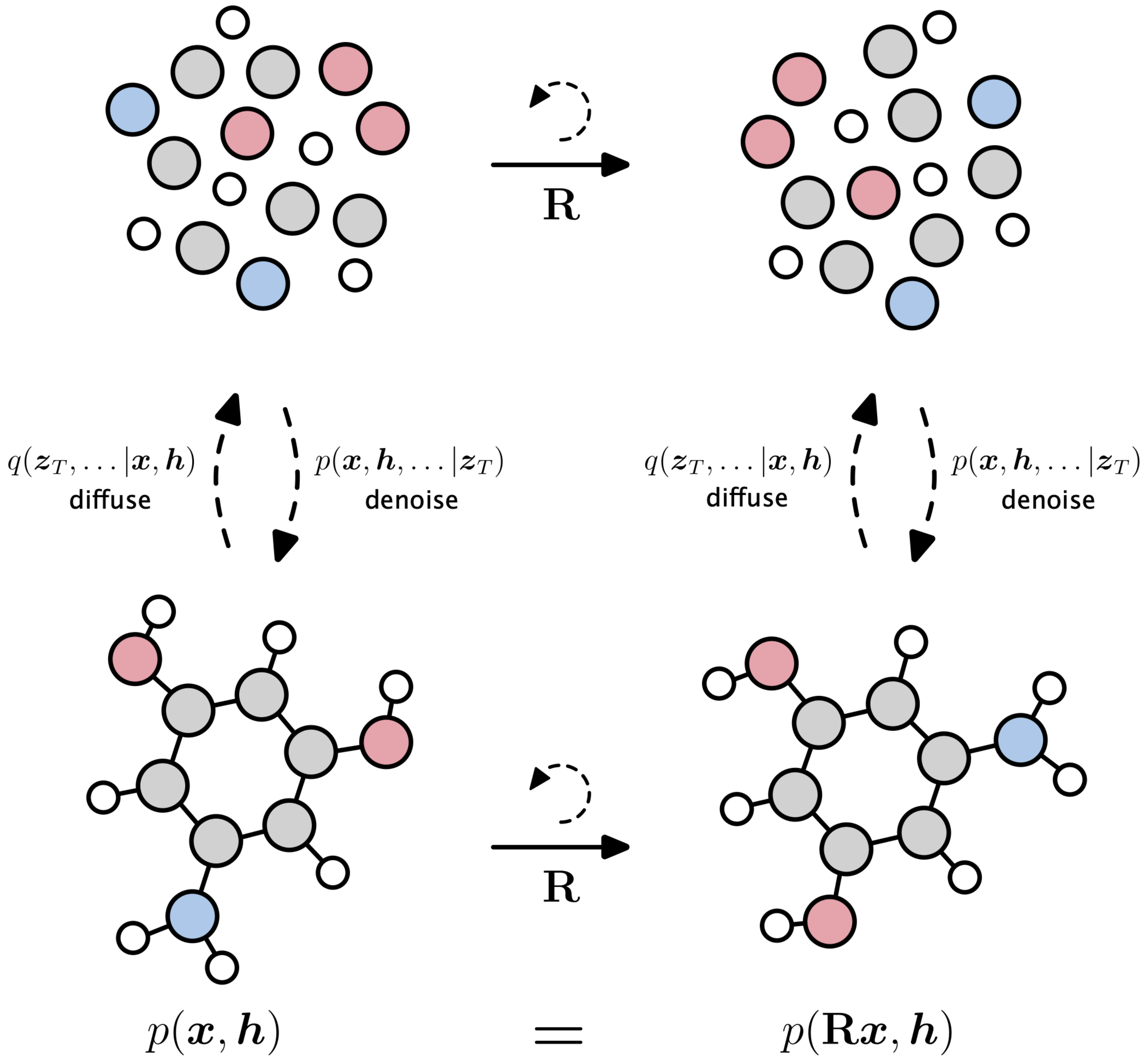
- Geodiff, equivariant diffusion to generate positional data of molecules

- E-NF, an equivariant normalizing flow to generate molecules

Xu, Minkai, et al. "Geodiff: A geometric diffusion model for molecular conformation generation."
Satorras, Victor Garcia, et al. "E(n) equivariant normalizing flows."

EDMs: Equivariant Diffusion Models

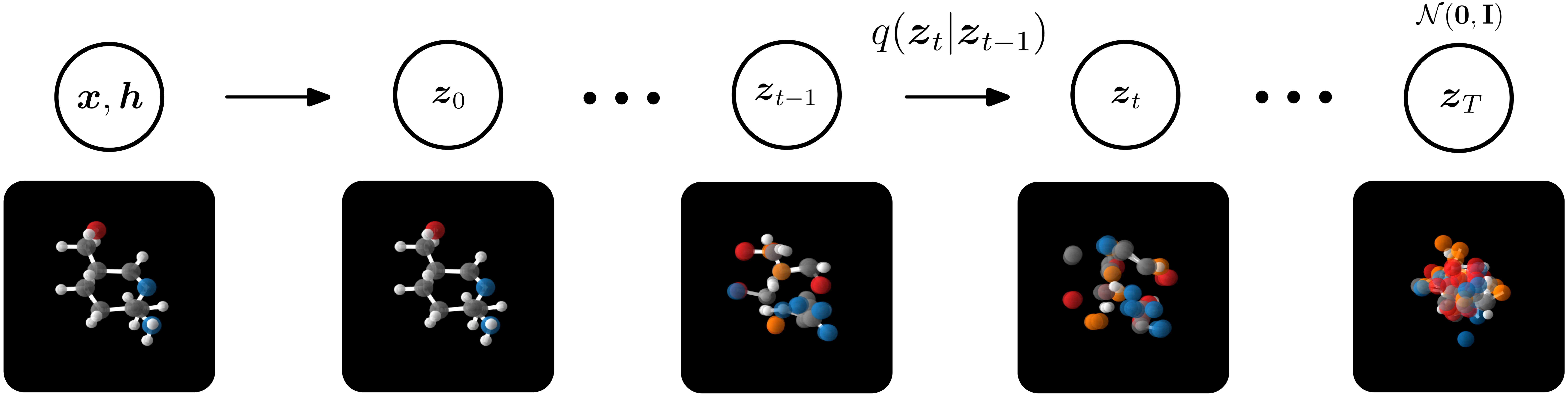
- Diffusion process to destroy info
- Learn denoising process to generate
- Handles continuous and discrete data



Diffusion Process

Equivariant Diffusion Models

- Adds Gaussian noise over time steps t = 0, …, T
- Is equivariant to rotations and translations

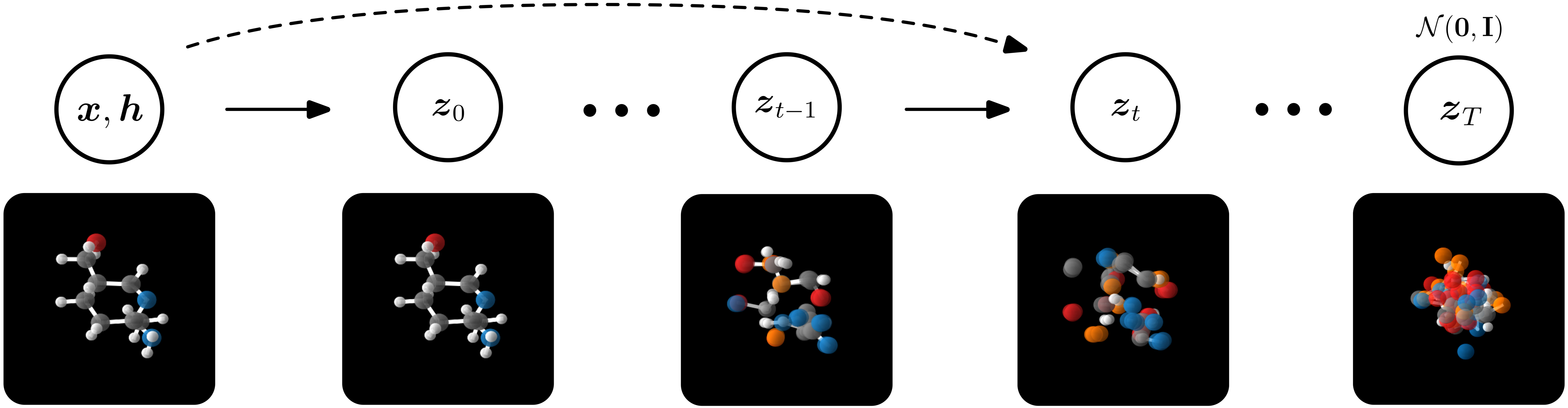


Diffusion Process

Equivariant Diffusion Models

- Jump to time step t with

$$q(\boldsymbol{z}_t|\boldsymbol{x},\boldsymbol{h}) = \mathcal{N}_{xh}(\boldsymbol{z}_t|\alpha_t[\boldsymbol{x},\boldsymbol{h}], \sigma_t^2\mathbf{I})$$



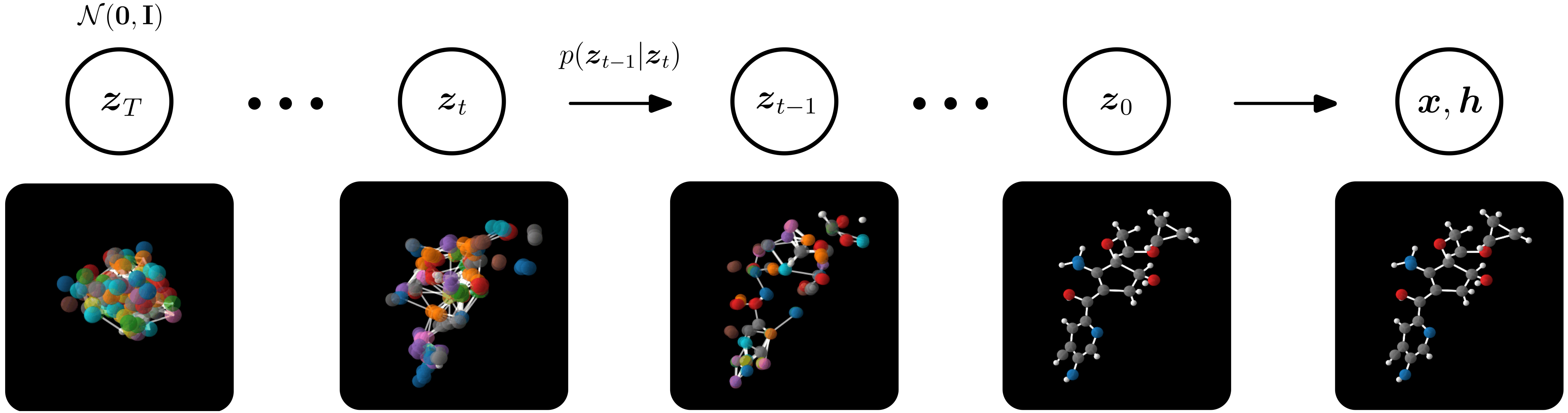
To generate molecules

Learn the *reverse* denoising process

Learnable Denoising Process

Equivariant Diffusion Models

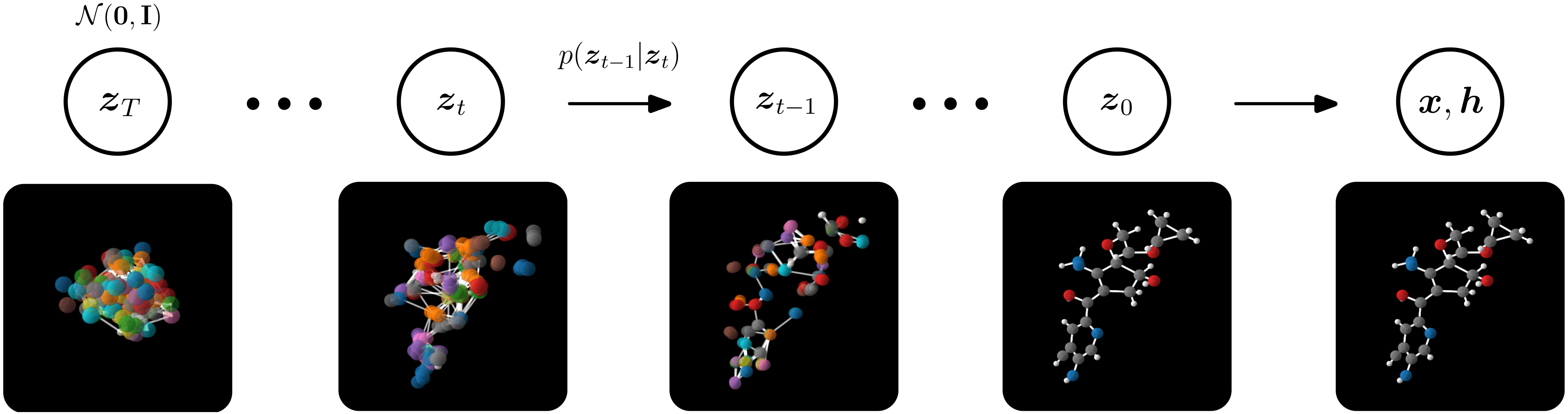
- Denoise with distributions $p\left(\boldsymbol{z}_{t-1} \mid \boldsymbol{z}_t\right)=\mathcal{N}\left(\boldsymbol{z}_{t-1} \mid \boldsymbol{\mu}_{t \rightarrow t-1}\left(\hat{\boldsymbol{x}}, \boldsymbol{z}_t\right), \sigma_{t \rightarrow t-1}^2 \mathbf{I}\right)$
- Chosen to have same form as the *true* denoising process



Learnable Denoising Process

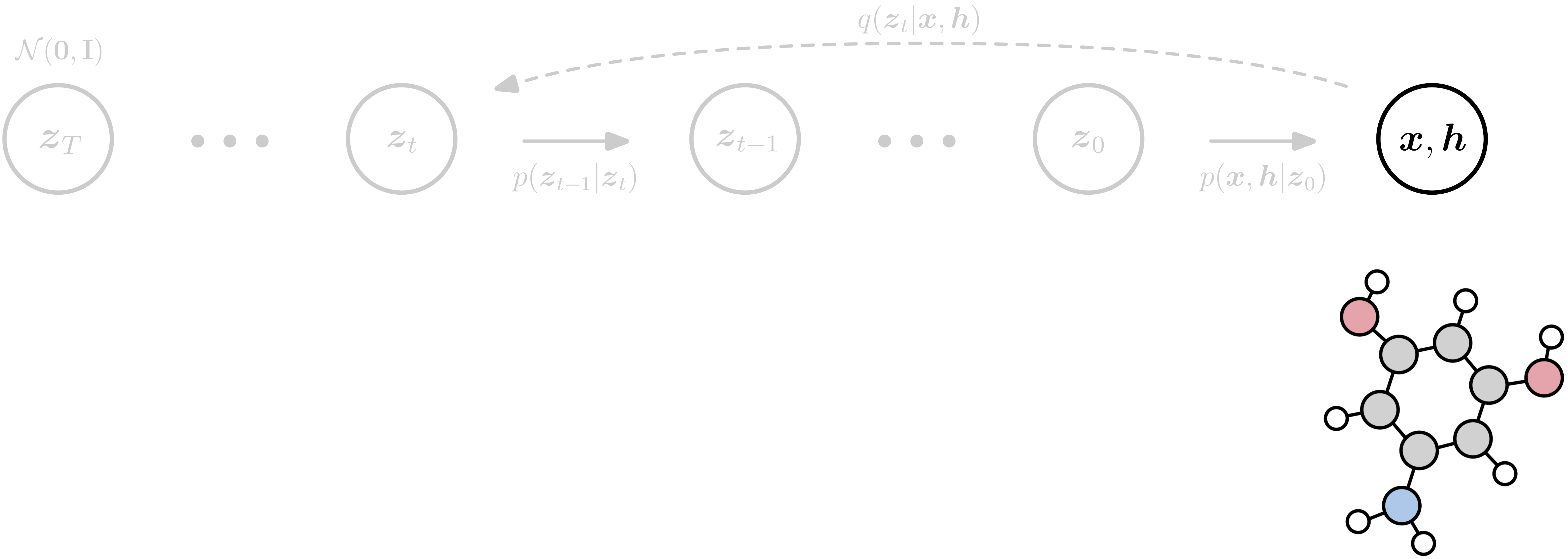
Equivariant Diffusion Models

- Denoise with distributions $p\left(\boldsymbol{z}_{t-1} \mid \boldsymbol{z}_t\right) \approx \mathcal{N}\left(\boldsymbol{z}_{t-1} \mid \underline{\hat{\boldsymbol{\mu}}_{t \rightarrow t-1}\left(\boldsymbol{z}_t\right)}, \sigma_{t \rightarrow t-1}^2 \mathbf{I}\right)$
- Chosen to have same form as the *true* denoising process



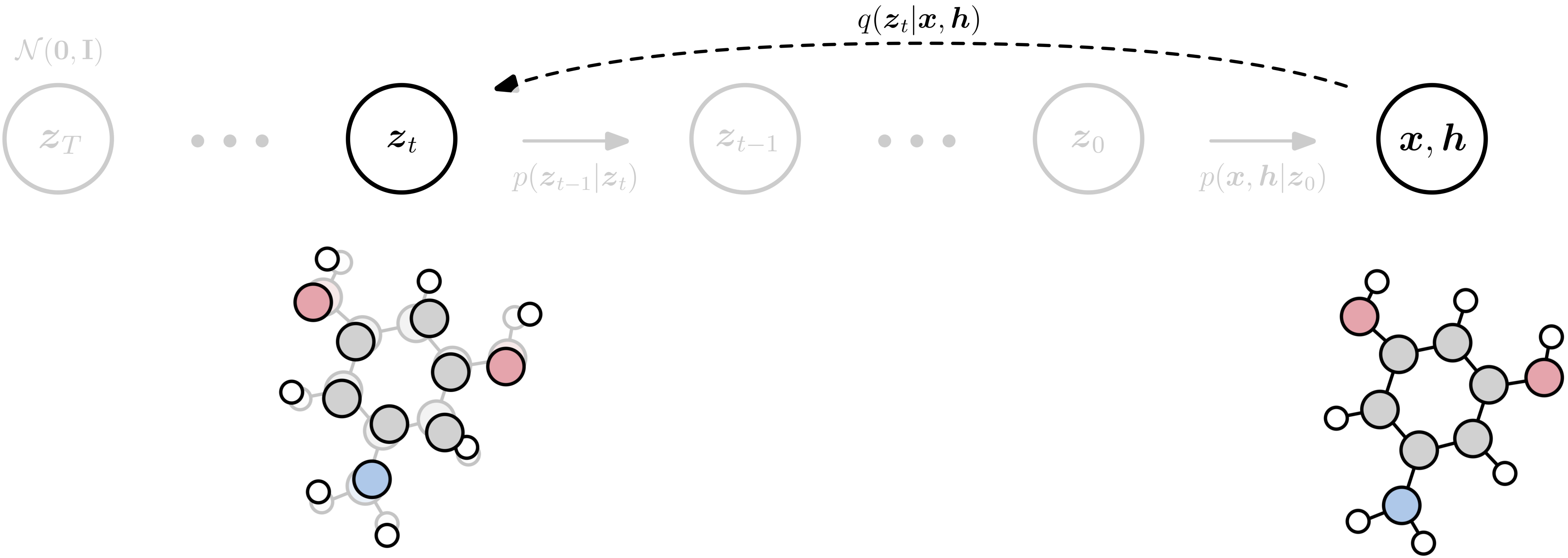
Optimization

Equivariant Diffusion Models



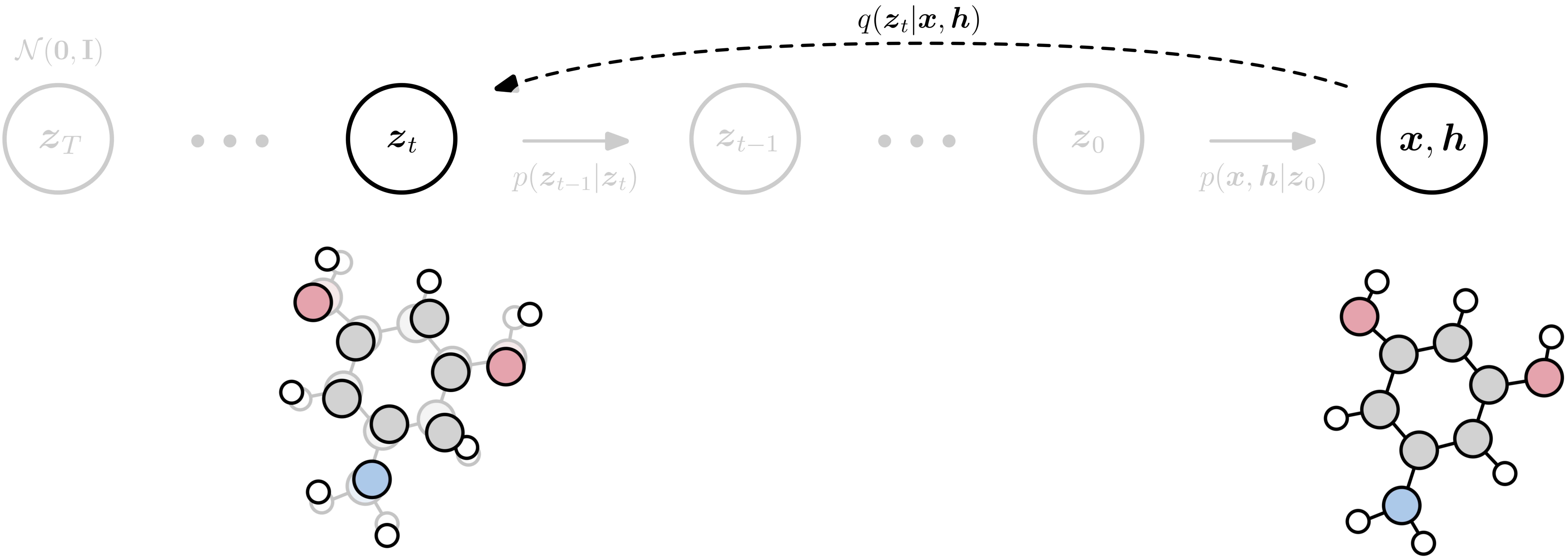
Optimization

Equivariant Diffusion Models



Optimization

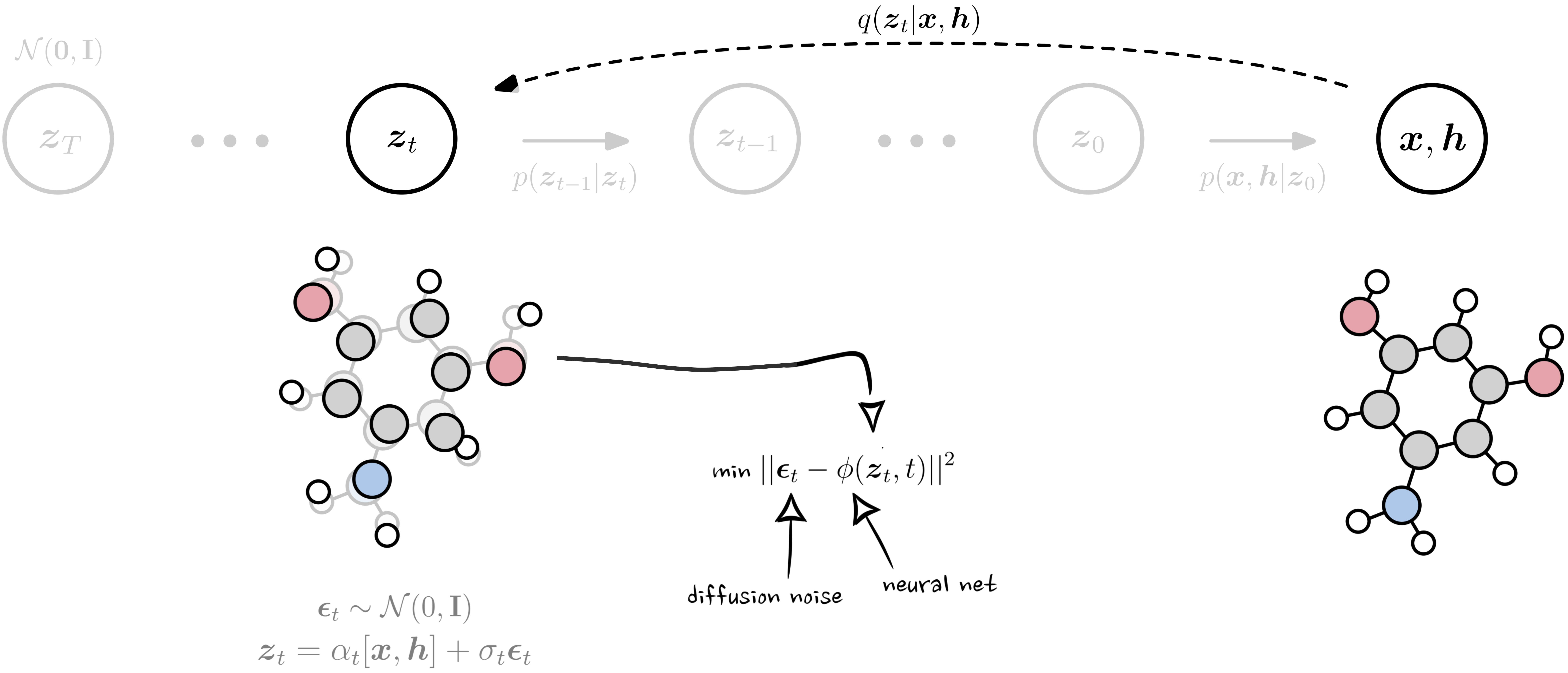
Equivariant Diffusion Models



$$\boldsymbol{\epsilon}_t \sim \mathcal{N}(0, \mathbf{I})$$
$$\boldsymbol{z}_t = \alpha_t[\boldsymbol{x}, \boldsymbol{h}] + \sigma_t \boldsymbol{\epsilon}_t$$

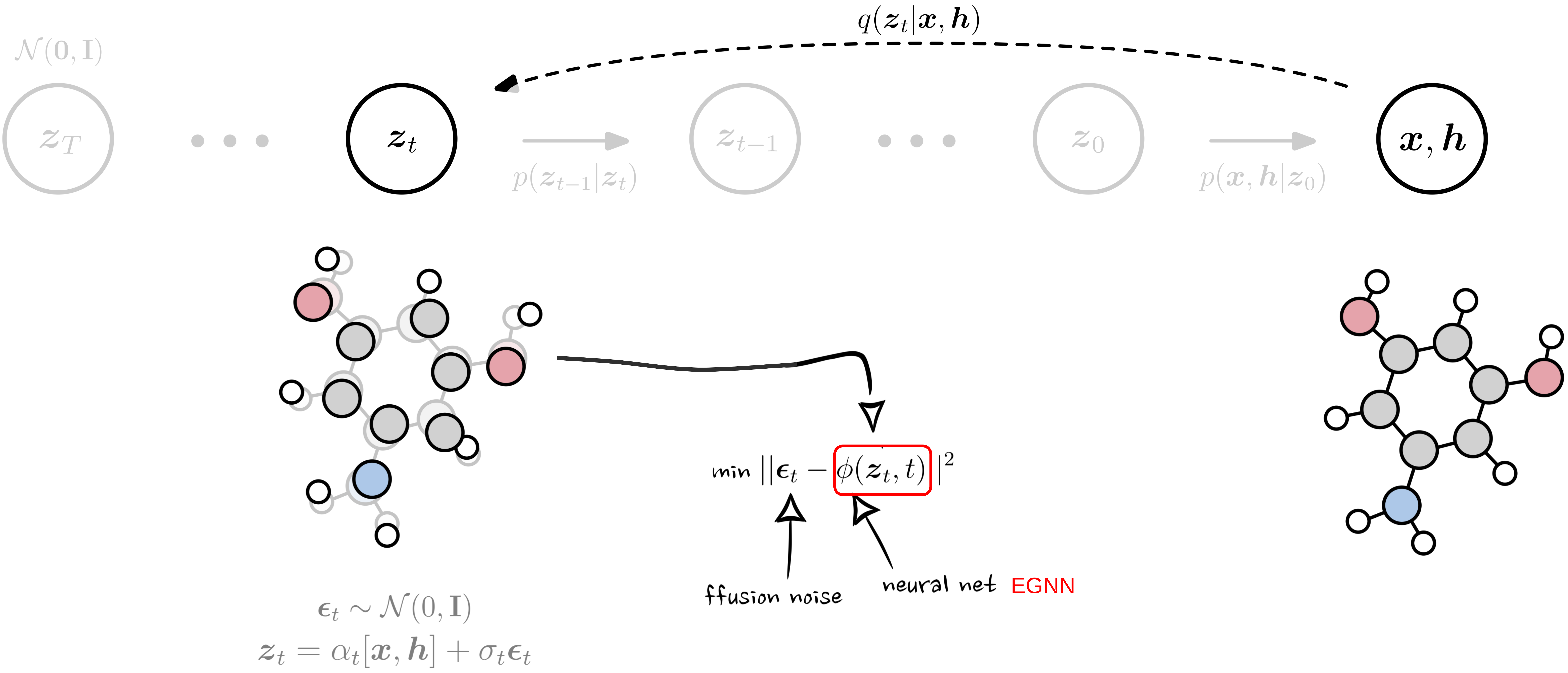
Optimization

Equivariant Diffusion Models



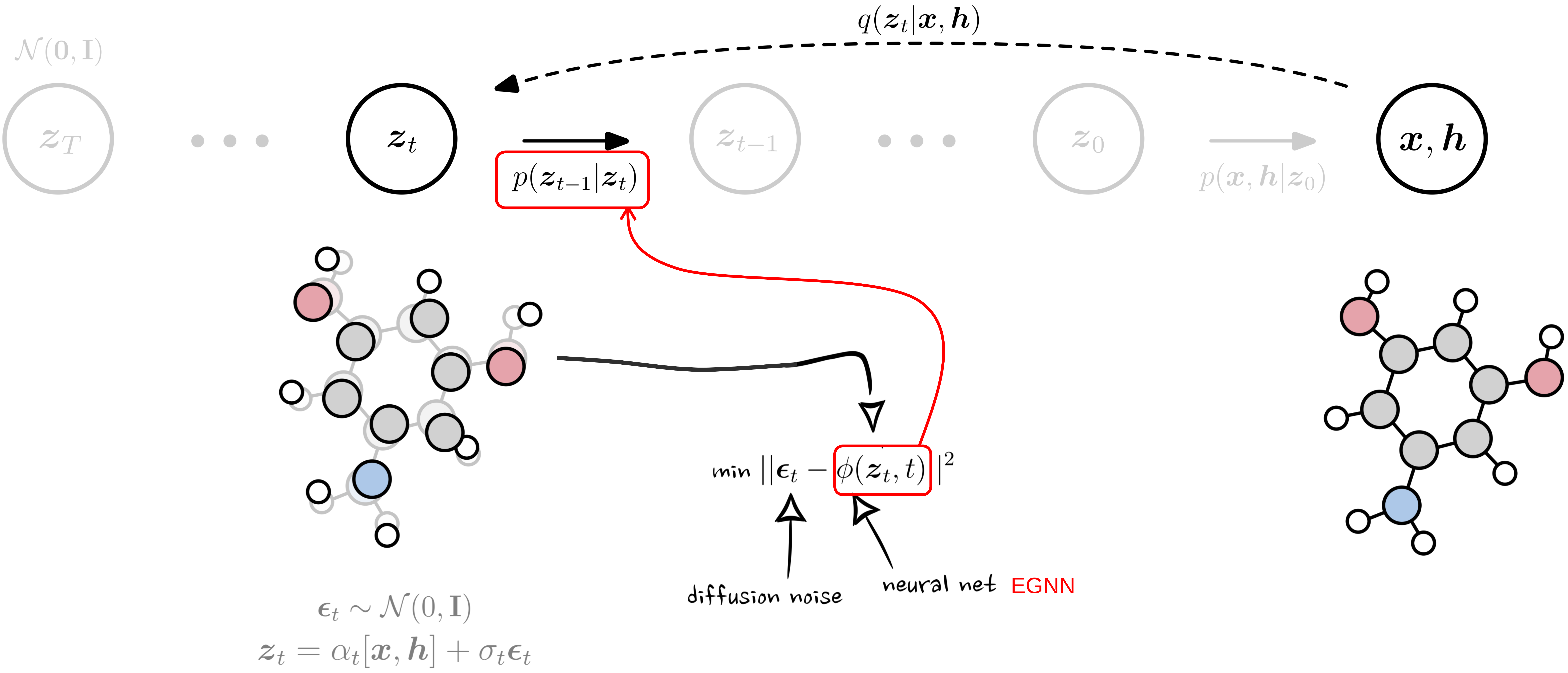
Optimization

Equivariant Diffusion Models



Optimization

Equivariant Diffusion Models



Discrete Data

Equivariant Diffusion Models

- Atom type is onehot, charge is integer
- Concatenate everything together
- Log-likelihood computation in the paper

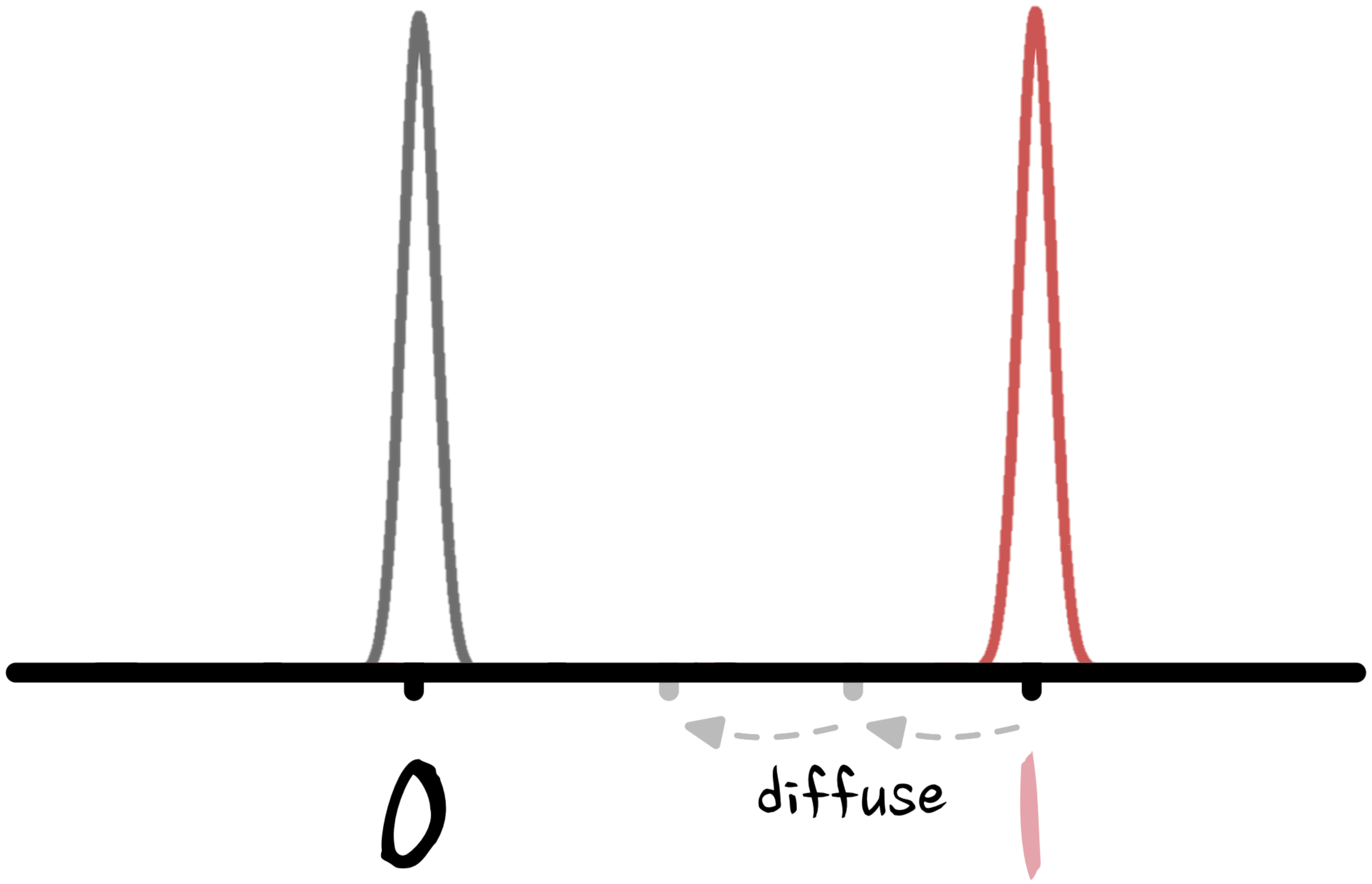
$$[\boldsymbol{x}, \qquad \boldsymbol{h}^{\text{onehot}}]$$

Discrete Data

Equivariant Diffusion Models

Example: Diffusion on binary values

- If more than 1, overlap later
- If less than 1, overlap earlier

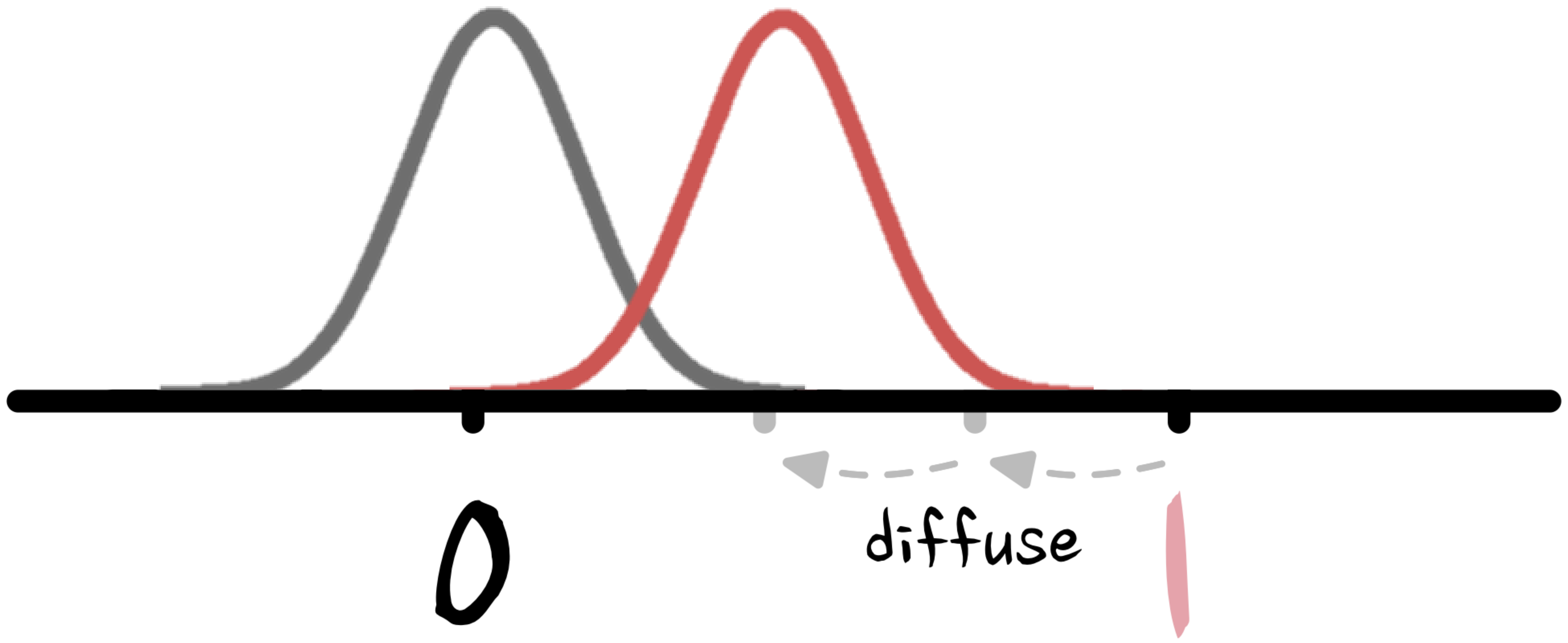


Discrete Data

Equivariant Diffusion Models

Example: Diffusion on binary values

- If more than 1, overlap later
- If less than 1, overlap earlier

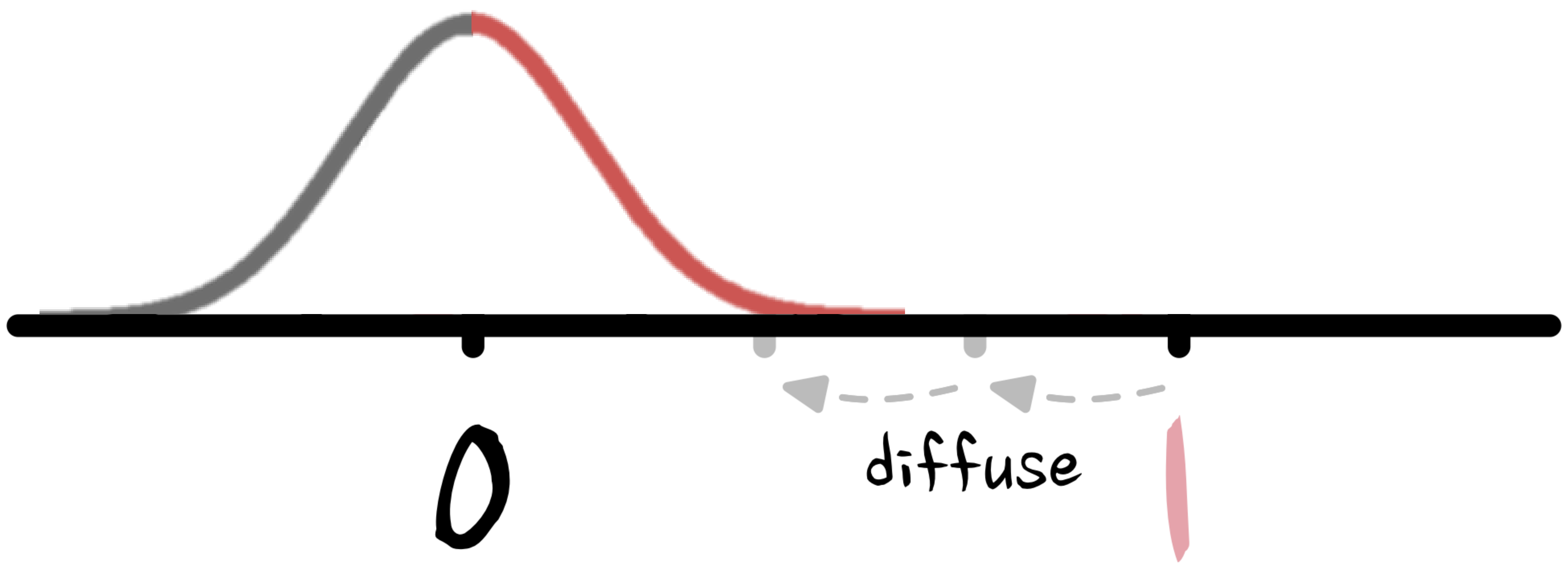


Discrete Data

Equivariant Diffusion Models

Example: Diffusion on binary values

- If more than 1, overlap later
- If less than 1, overlap earlier



Discrete Data

Equivariant Diffusion Models

- Add a weighting of 0.25 for the atom type
- Atom type is diffused *earlier* in the diffusion process
- Therefore it is modelled *later* in the denoising process
- 47% -> 82% on molecule stability metric

$$[\boldsymbol{x}, 0.25\ \boldsymbol{h}^{\text{onehot}}]$$

Results

- Outperforms existing methods
- Scales to GEOM-Drugs

Table Neg. log-likelihood $-\log p(\mathbf{x}, \mathbf{h}, M)$, atom stability and molecule stability with standard deviations across 3 runs on QM9, each drawing 10000 samples from the model.

# Metrics	NLL	Atom stable (%)	Mol stable (%)
E-NF	-59.7	85.0	4.9
G-Schnet	N.A	95.7	68.1
GDM	-94.7	97.0	63.2
GDM-aug	-92.5	97.6	71.6
EDM (ours)	**-110.7**$\pm$1.5	**98.7**$\pm$0.1	**82.0**$\pm$0.4
Data		99.0	95.2

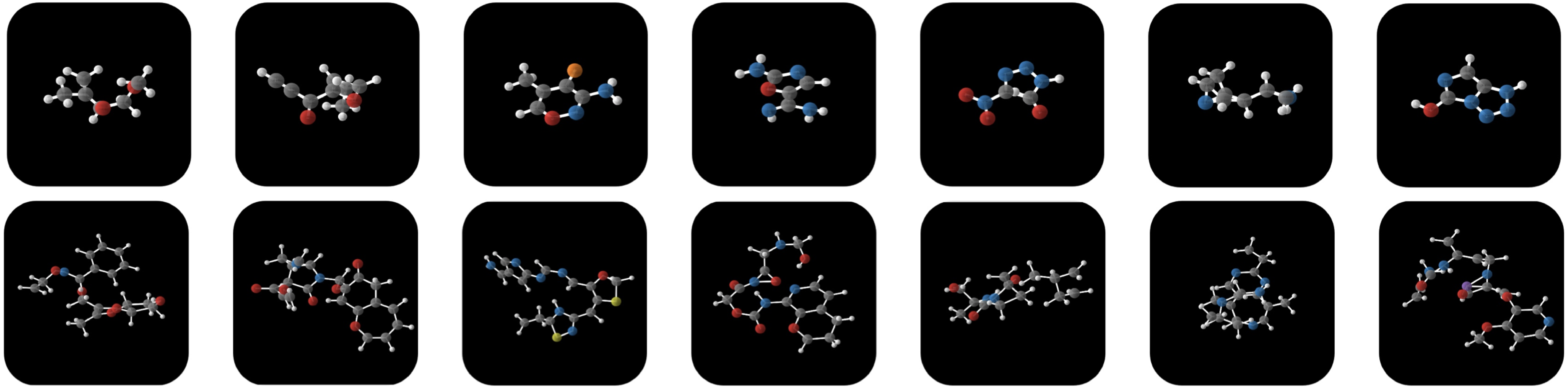
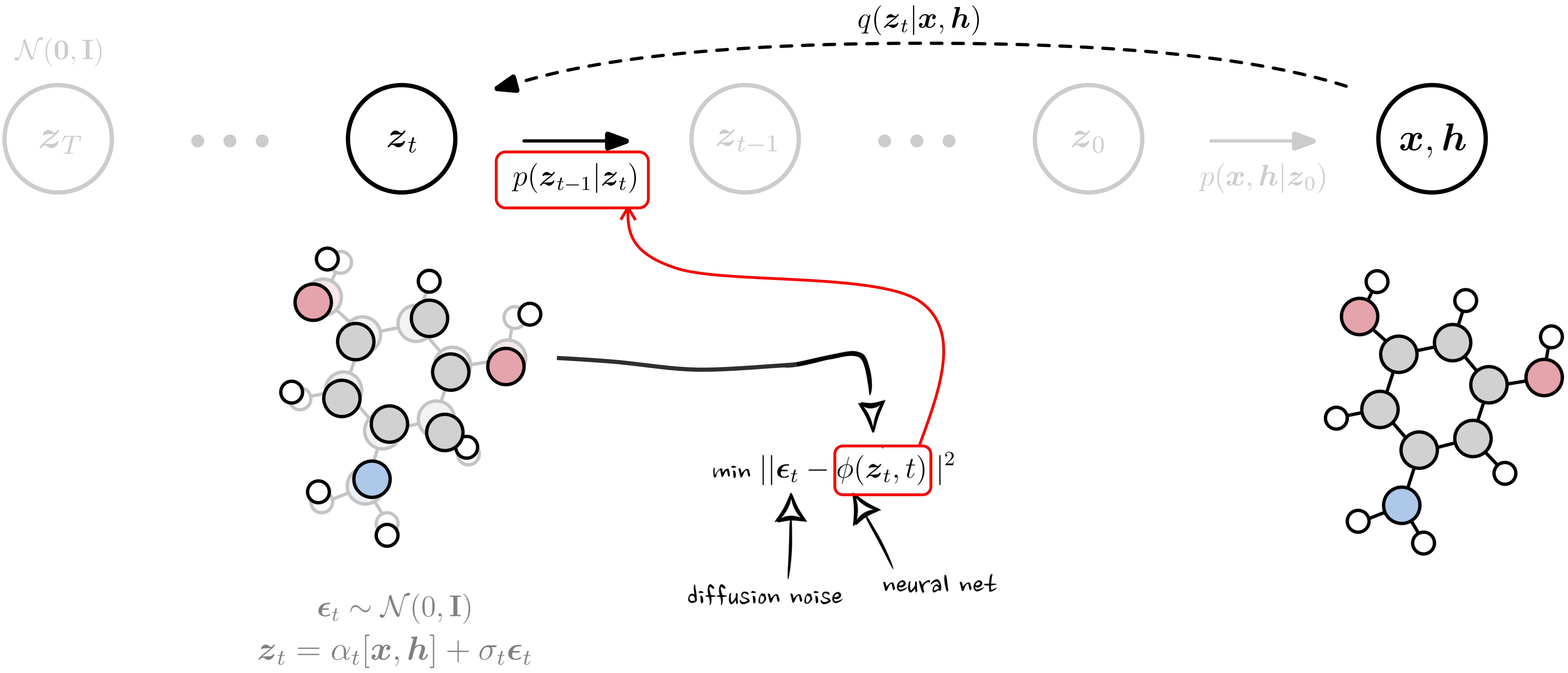


Figure Selection of samples generated by the denoising process of our EDM trained on QM9 (up) and GEOM-DRUGS (down).

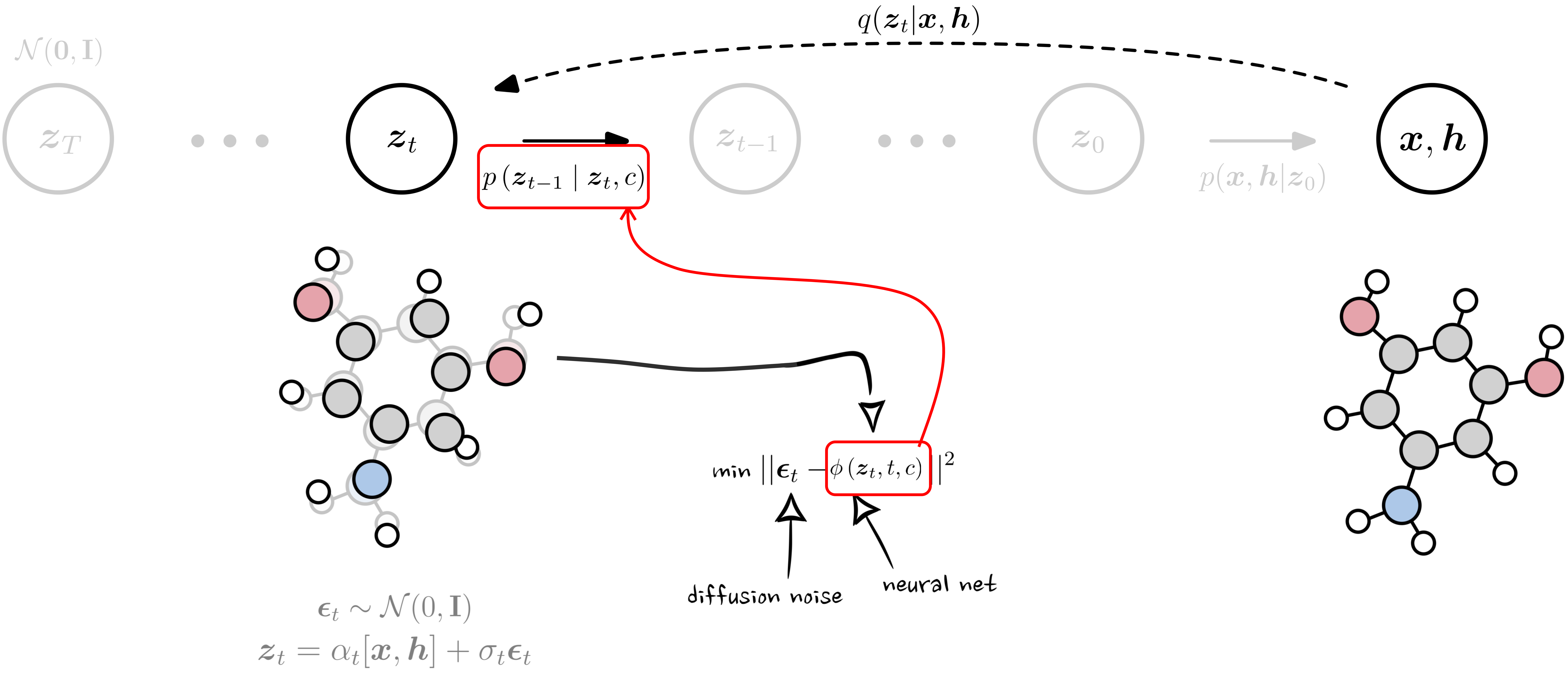
Conditional Generation

Equivariant Diffusion Models



Conditional Generation

Equivariant Diffusion Models



Results

- Conditional generation, increase polarizability.

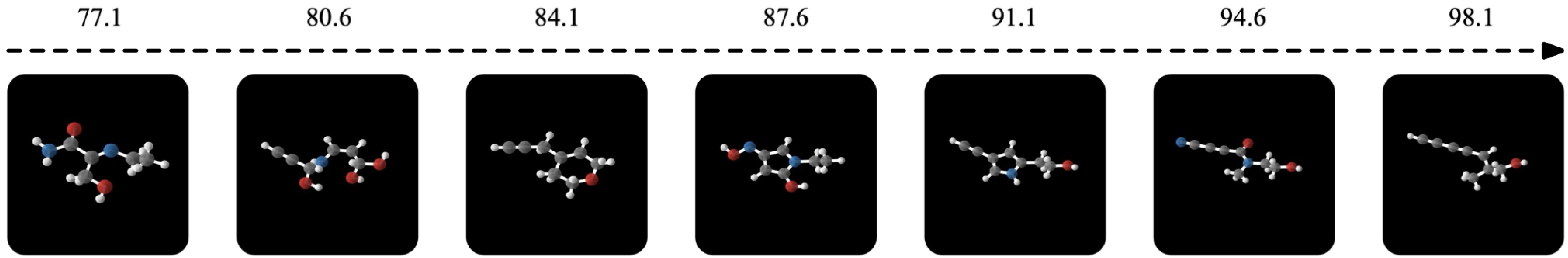


Figure Generated molecules by our Conditional EDM when interpolating among different Polarizability α values with the same reparametrization noise ϵ. Each α value is provided on top of each image.

Conclusion

- Diffusion models have demonstrated outstanding performance as generative models in the image domain. And recently are being extended to the 3D Euclidean domain for molecular modelling.
- Interesting future directions:
  - Scaling up linker generation for docking problems to real size proteins/dockers.
  - Diffusion models for Potential Energy Surface exploration.
  - Diffusion models for Molecular Dynamics.

Contents

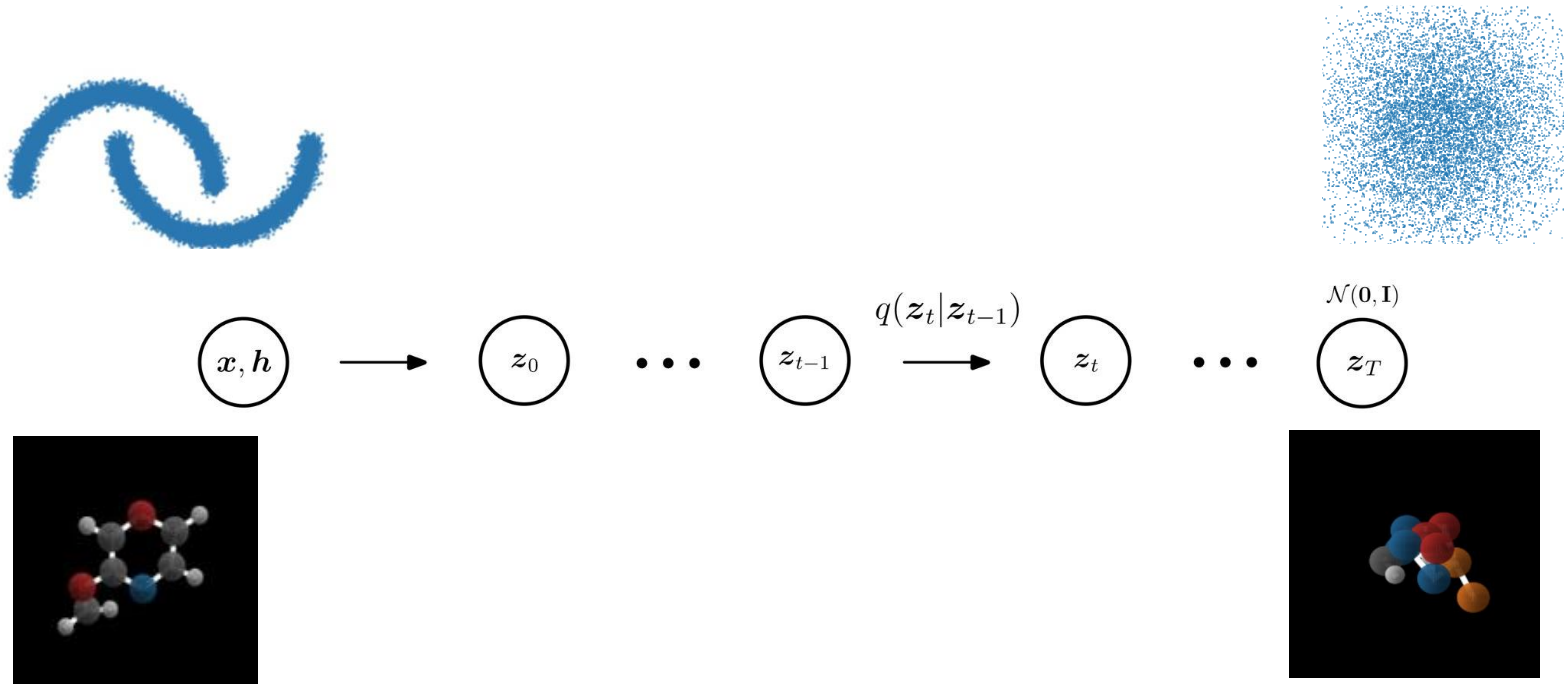
Paper Presentation (Equivariant Diffusion for Molecule Generation in 3D)

- Why Generate Molecules?
- Background
  - Denoising Diffusion
  - Equivariant Graph Neural Networks
- EDMs: Equivariant Diffusion Models
  - Diffusion Denoising Process
  - Optimization
  - Results

Workshop

- **Coding a diffusion model for toy data**
- **Coding a diffusion model for molecular data**

Workshop



$\mathcal{N}(\mathbf{0}, \mathbf{I})$
$q(\boldsymbol{z}_t|\boldsymbol{z}_{t-1})$
$\boldsymbol{x}, \boldsymbol{h}$
$\boldsymbol{z}_0$
$\boldsymbol{z}_{t-1}$
$\boldsymbol{z}_t$
$\boldsymbol{z}_T$

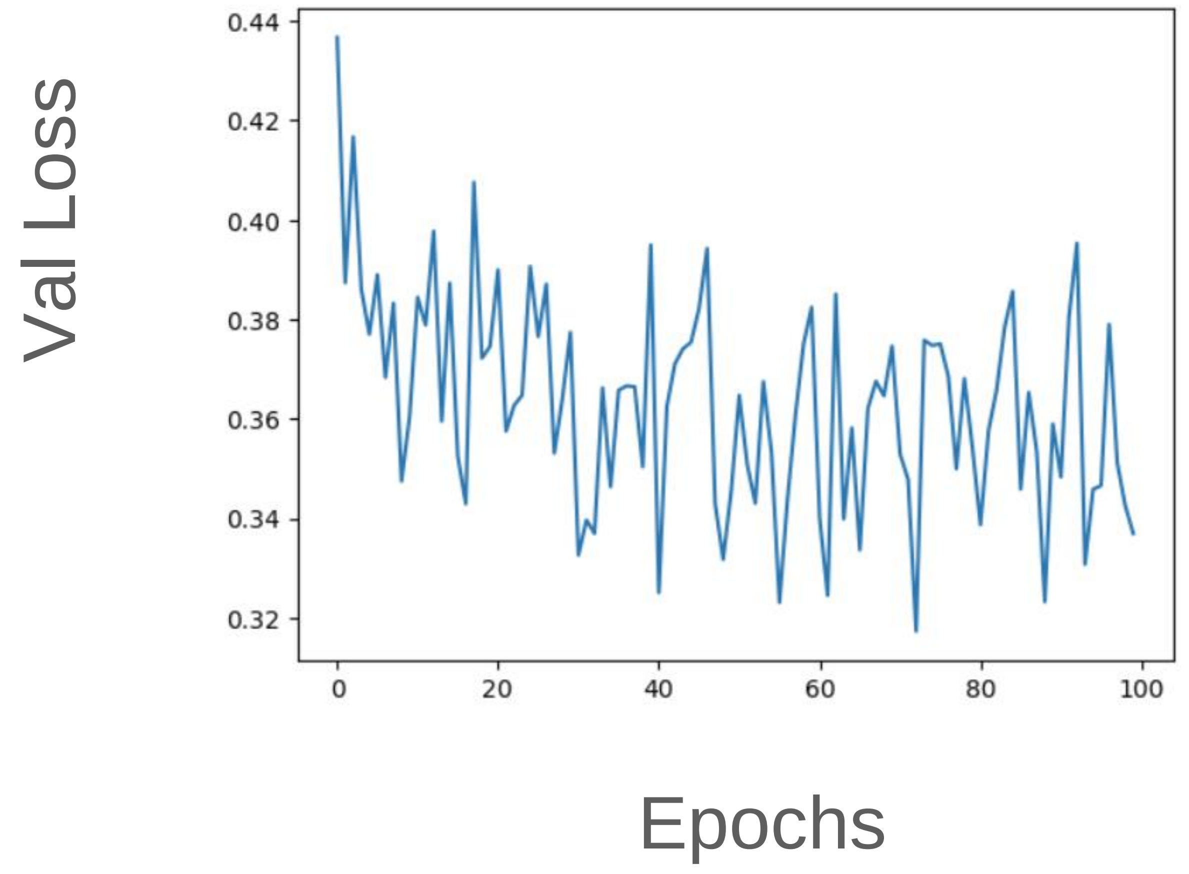
For the coding session please go to colab
https://colab.research.google.com/drive/1P-5yL4PFONx03Ekpu1vsNp3ghkFoOR_2?usp=sharing
(We shared the link in Slack #day-3)

Getting Started:
Accessing code and installing environment

Notebook 01: Toy Diffusion

Task: Code a Diffusion Model for the Two Moons toy dataset

Expected loss: Between 0.3 and 0.4

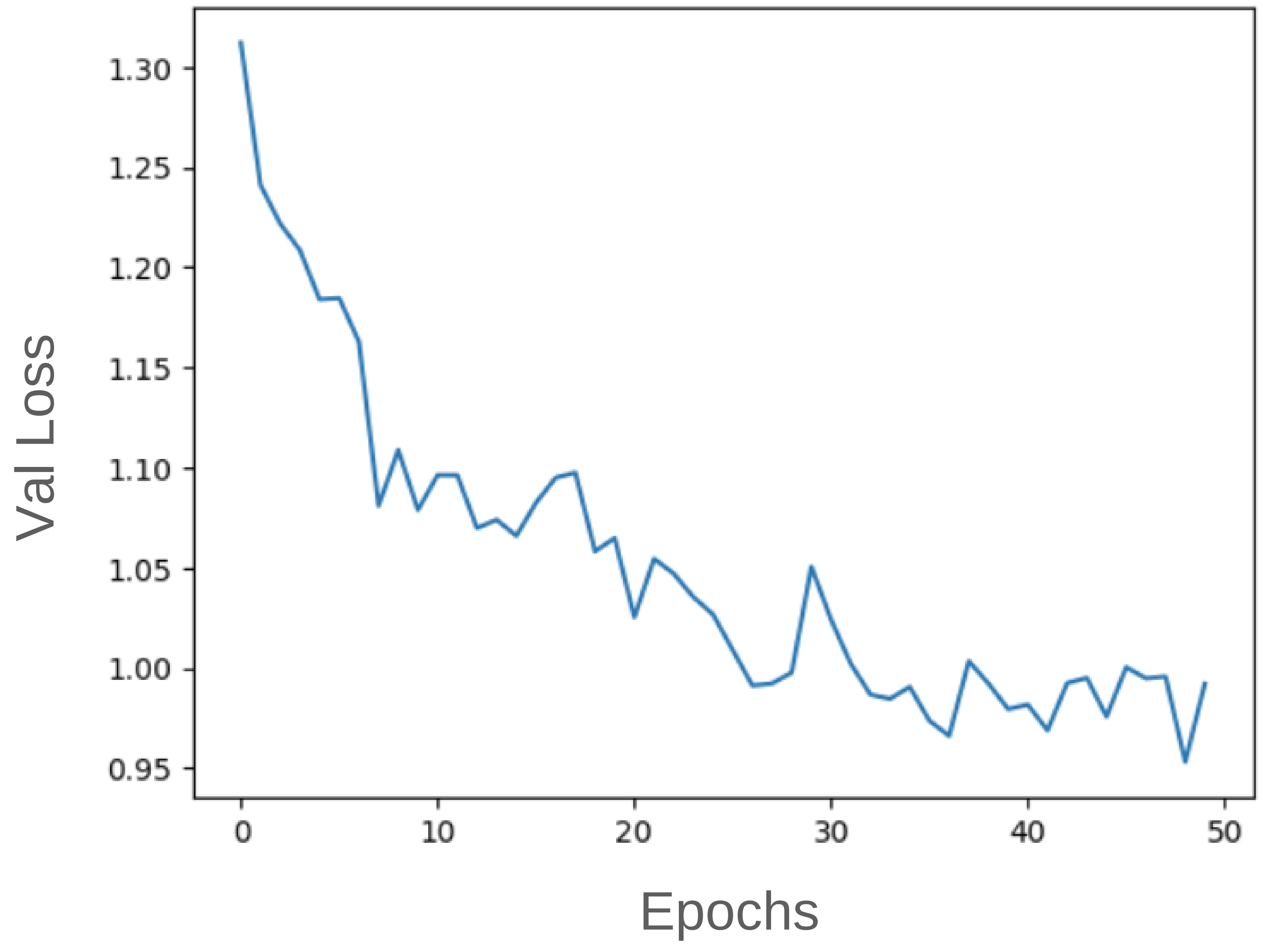


Sampled distribution after 100 epochs

Notebook 03: Equivariant Diffusion Model

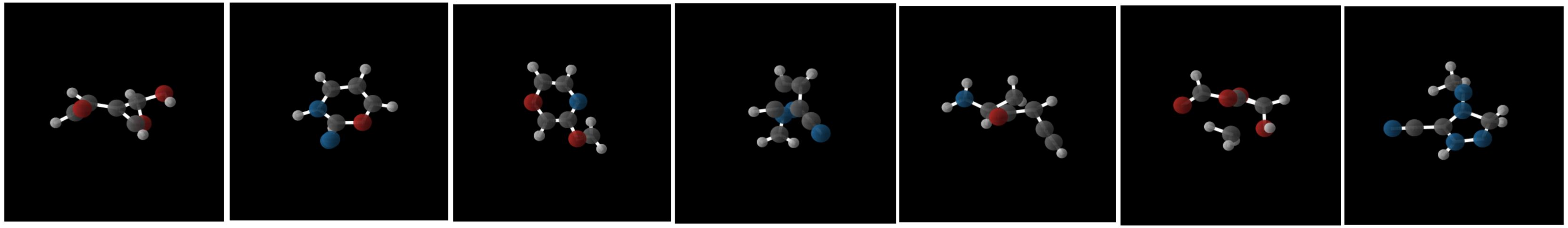
Task: Adapt the previous Denoising Diffusion model from the toy dataset to Mini-QM9

Expected loss: The validation loss should be below 1.0 if you implemented the cosine dataset Next we show a plot of what loss to expect

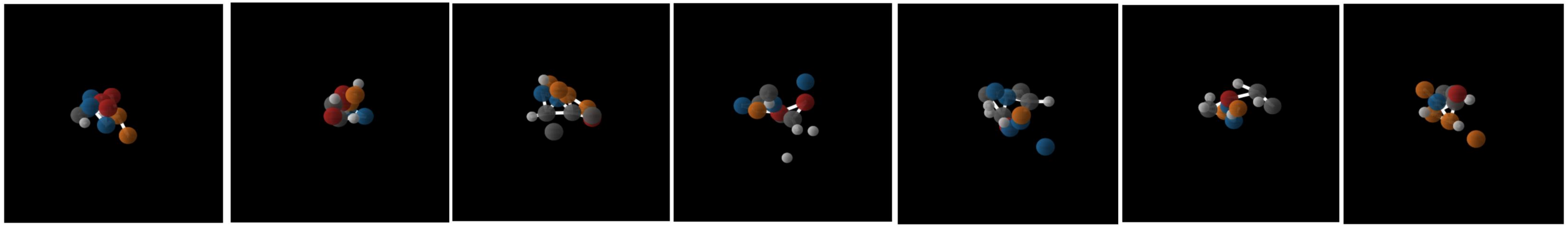


Notebook 04: EDM Evaluation

Trained model
- Atom stability: 0.5 ~ 0.8
- Molecule Stability: 0.0



Wrong model
- Atom stability: ~0.0
- Molecule Stability: 0.0



Additional content

Two For one: Diffusion Models and Force Fields for Coarse-Grained Molecular Dynamics

[2302.00600] Two for One: Diffusion Models and Force Fields for Coarse-Grained Molecular Dynamics (arxiv.org)