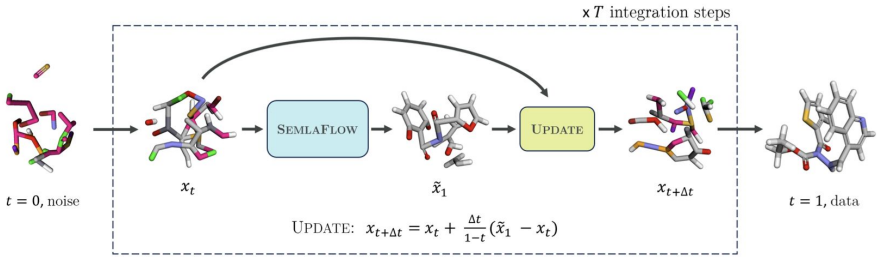
SynCoGen: Synthesizable 3D Molecule Generation via Joint Reaction and Coordinate Modeling

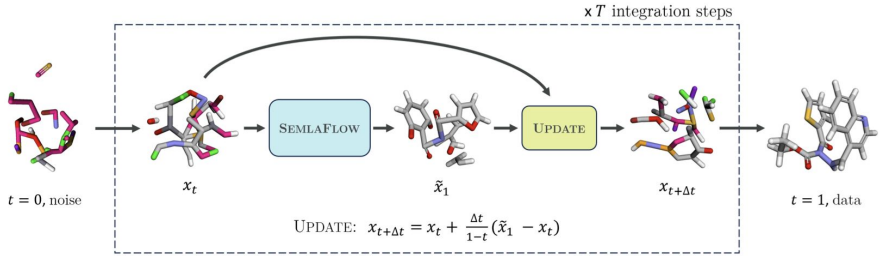
Motivation

Problem: All-atom 3D molecule *co-generation* (structure and conformer) does not consider **synthesizability**!



Motivation

Problem: All-atom 3D molecule *co-generation* (structure and conformer) does not consider **synthesizability**!



- Almost all 3D molecular co-generators (SemlaFlow, JODO, MiDi, EQGAT-Diff) generate atom types and coordinates
- Difficult to guarantee synthesis pathways without building block-level constraints or training!

Motivation

Our approach to tackling this problem: **SynCoGen** (**Syn**thesizable **Co**-**Gen**eration), framework to generate 3D molecules and synthetic pathways.

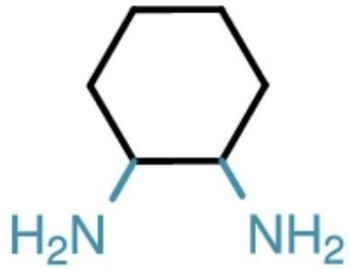
- **Generative Framework**: Generate building blocks, reactions, and coordinates with the same model
- **SynSpace Dataset:** We curate 622,766 synthesizable molecules and reaction paths, 3,360,908 low-energy conformations.
- **Empirical Validation:** 3D molecule generation, applied to basic linker design, pharmacophore analog generation

Motivation: Graph Generation

- Naive representation using BBs as nodes and reactions as edges is insufficient to describe a unique molecule, even if stereochemistry ignored

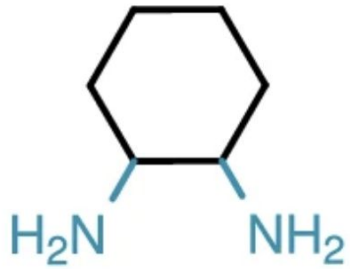
Motivation: Graph Generation

- Naive representation using BBs as nodes and reactions as edges is insufficient to describe a unique molecule, even if stereochemistry ignored
- To see this, consider:



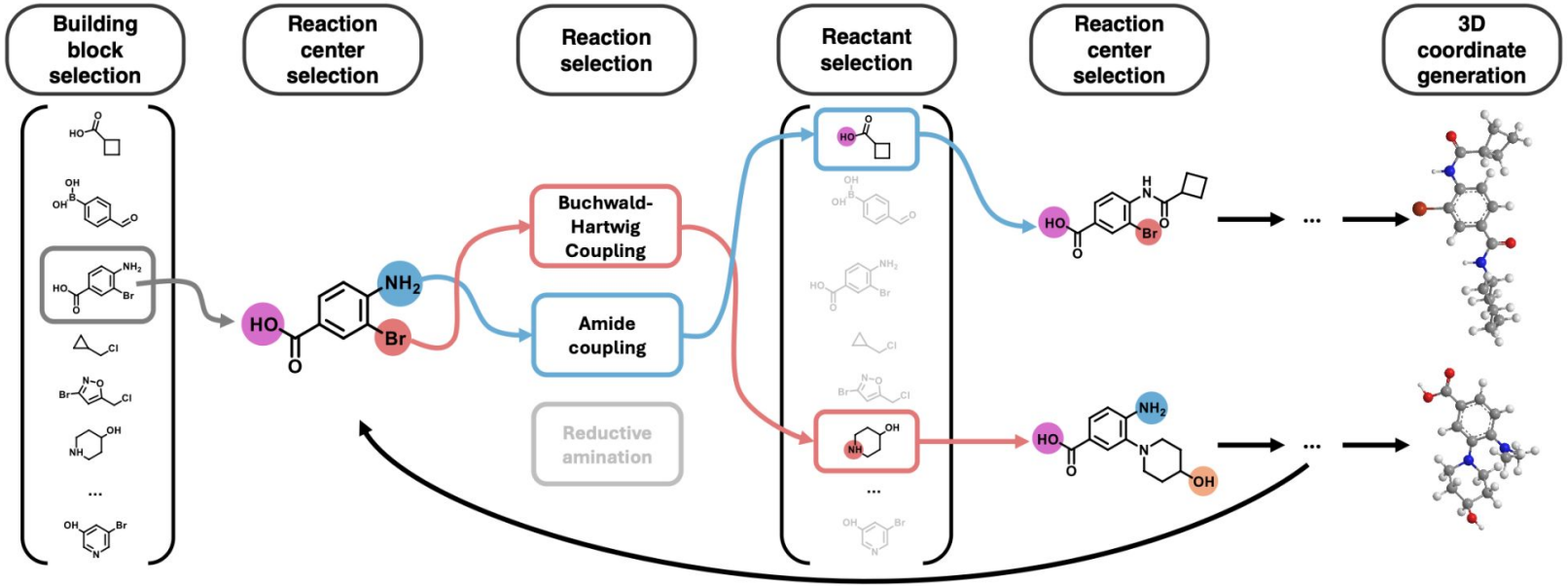
Motivation: Graph Generation

- Naive representation using BBs as nodes and reactions as edges is insufficient to describe a unique molecule, even if stereochemistry ignored
- To see this, consider:



If we couple this to a compatible building block, the BB-Rxn graph can’t tell us which amine group is occupied!

Dataset Generation



622,766 synthesizable molecules and reaction paths, 3,360,908 conformers

Data Format

Each molecule is a triple (X, E, C) where:

$$X \in \{0,1\}^{N \times |\mathcal{B}|+1}$$

N Building Block Onehots!

$$E \in \{0,1\}^{N \times N \times |\mathcal{R}| V_{\max}^2 + 2}$$

N x *N* Reaction Onehots!

Molecular Graph!

$$C \in \mathbb{R}^{N \times M \times 3}$$

3D Coordinates!

Data Format

Each molecule is a triple (X, E, C) where:

$$X \in \{0,1\}^{N \times |\mathcal{B}|+1}$$

N Building Block Onehots!

$$E \in \{0,1\}^{N \times N \times |\mathcal{R}| V_{\max}^2 + 2}$$

N x *N* Reaction Onehots!

Molecular Graph!

$$C \in \mathbb{R}^{N \times M \times 3}$$

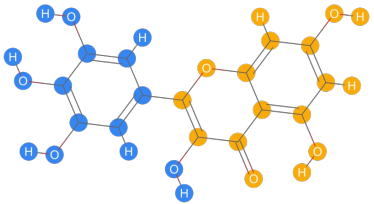
3D Coordinates!

1 extra index in *X*: mask index π_X
2 extra dimensions in *E*: mask index π_E + no-edge index λ_E

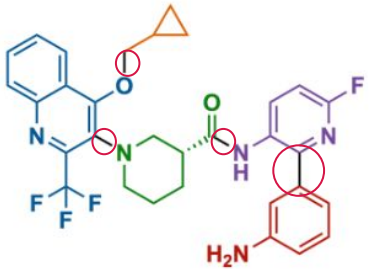
Note: each reaction $e_{ij} = (r, v_i, v_j)$ encodes both the reaction *r* and attachment point on building block *i* and *j* created by *r*.

Data Visualization - Molecular Graphs

- **Previous work - atom graph**
  - Atoms connected by bonds
  - Hard to constrain to pre-cursors this way



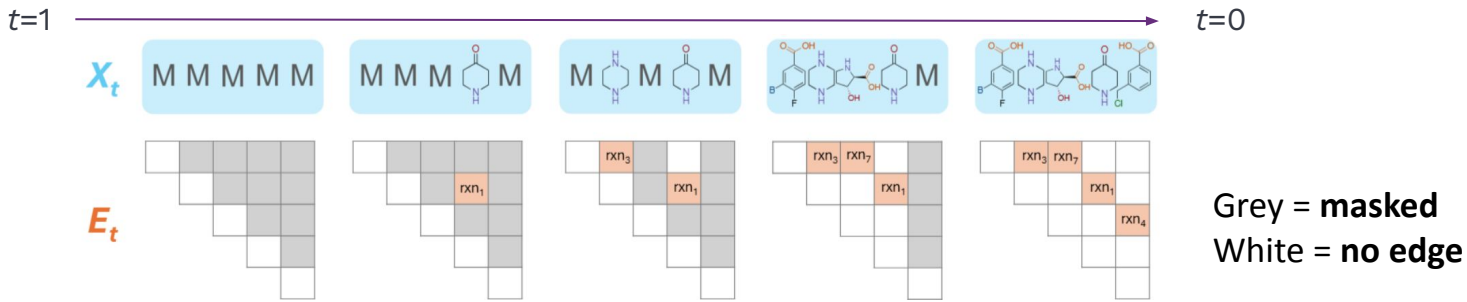
- **Building Block Graph**
  - Building blocks connected by chemical reactions
  - Better reflects experimental synthesis considerations



Reaction-induced bonds connect building blocks!

Training: Continuous FM, Discrete Diffusion

- For coordinates, flow matching should do the trick
- Q: How can we evolve building block and reaction selection along the same time steps?
- A: Masked diffusion! Noising process becomes transition to “mask” token

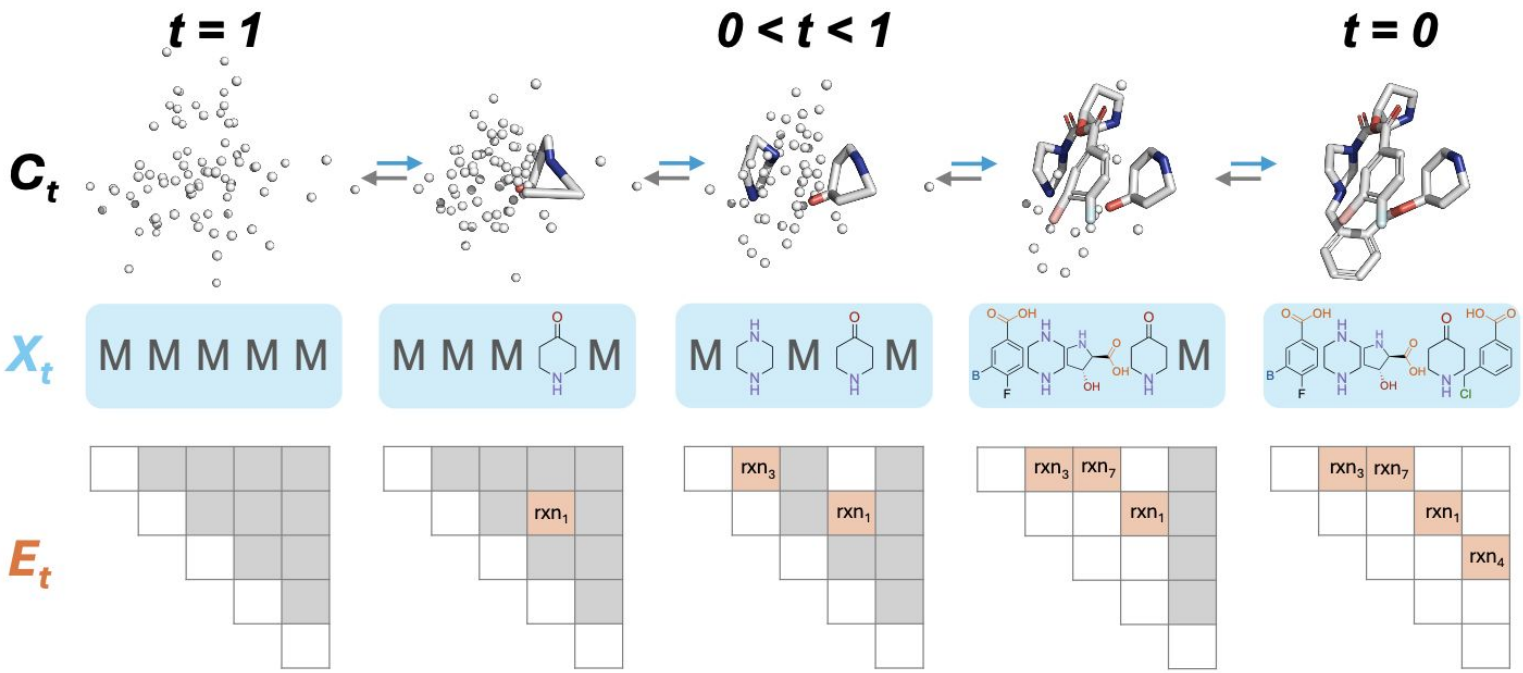


Grey = **masked**
White = **no edge**

* **symmetric**, disregard the lower triangle

Diagonals are no edge - no building block connects to itself

Generation Process



Considerations for Training and Sampling

- Want to support **variable atom counts**
  - Even with a prior over #BBs, atom count varies

Considerations for Training and Sampling

- Want to support **variable atom counts**
  - Even with a prior over #BBs, atom count varies

- At sampling time, extra atoms should be disregarded when no longer necessary
  - E.g, when building block identities are known

Considerations for Training and Sampling

- Want to support **variable atom counts**
  - Even with a prior over #BBs, atom count varies

- At sampling time, extra atoms should be disregarded when no longer necessary
  - E.g, when building block identities are known

Therefore, sampling should start with an upper bound on the possible number of atoms per building block and progressively remove them as we select BBs

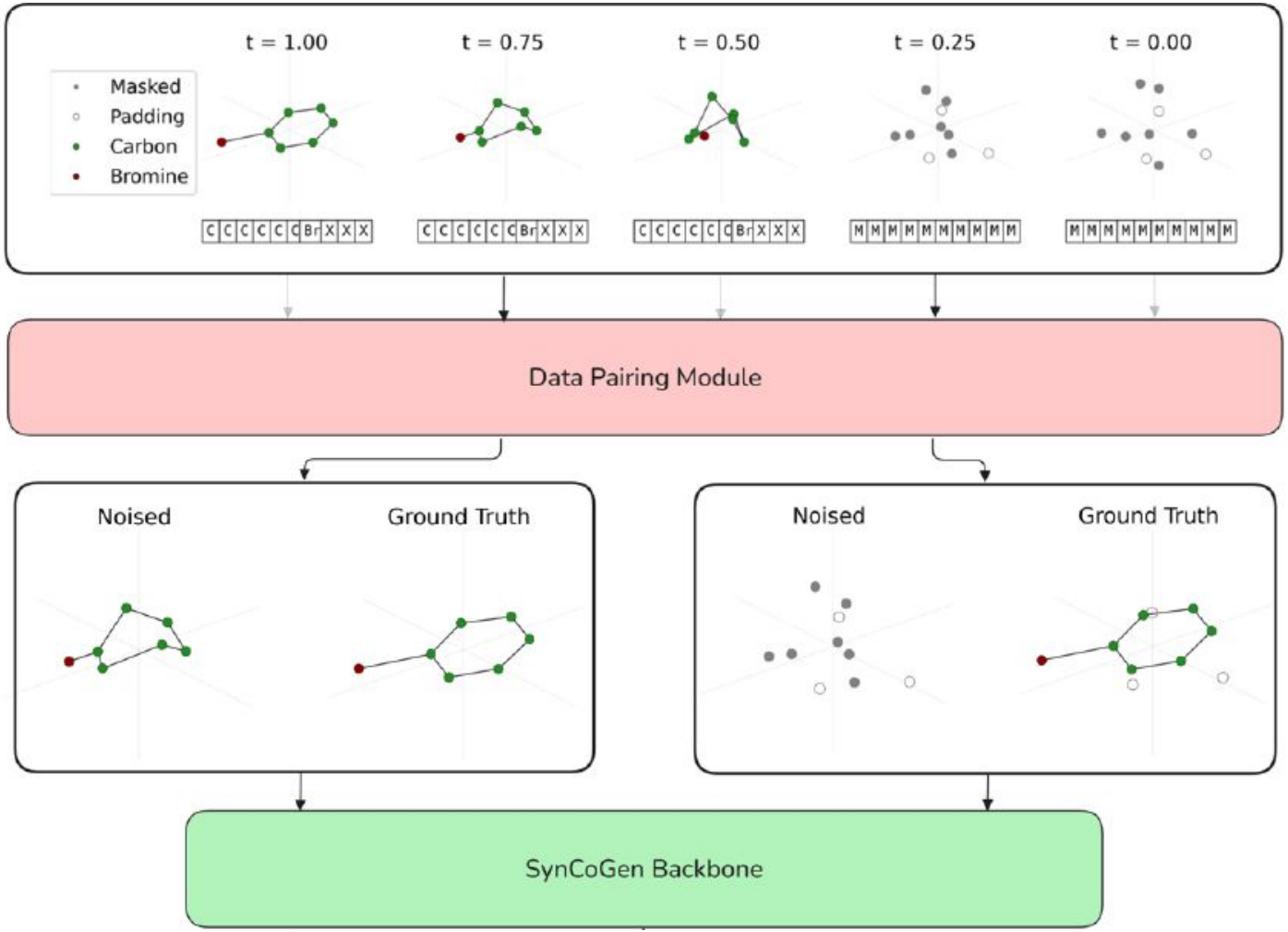
Considerations for Training and Sampling

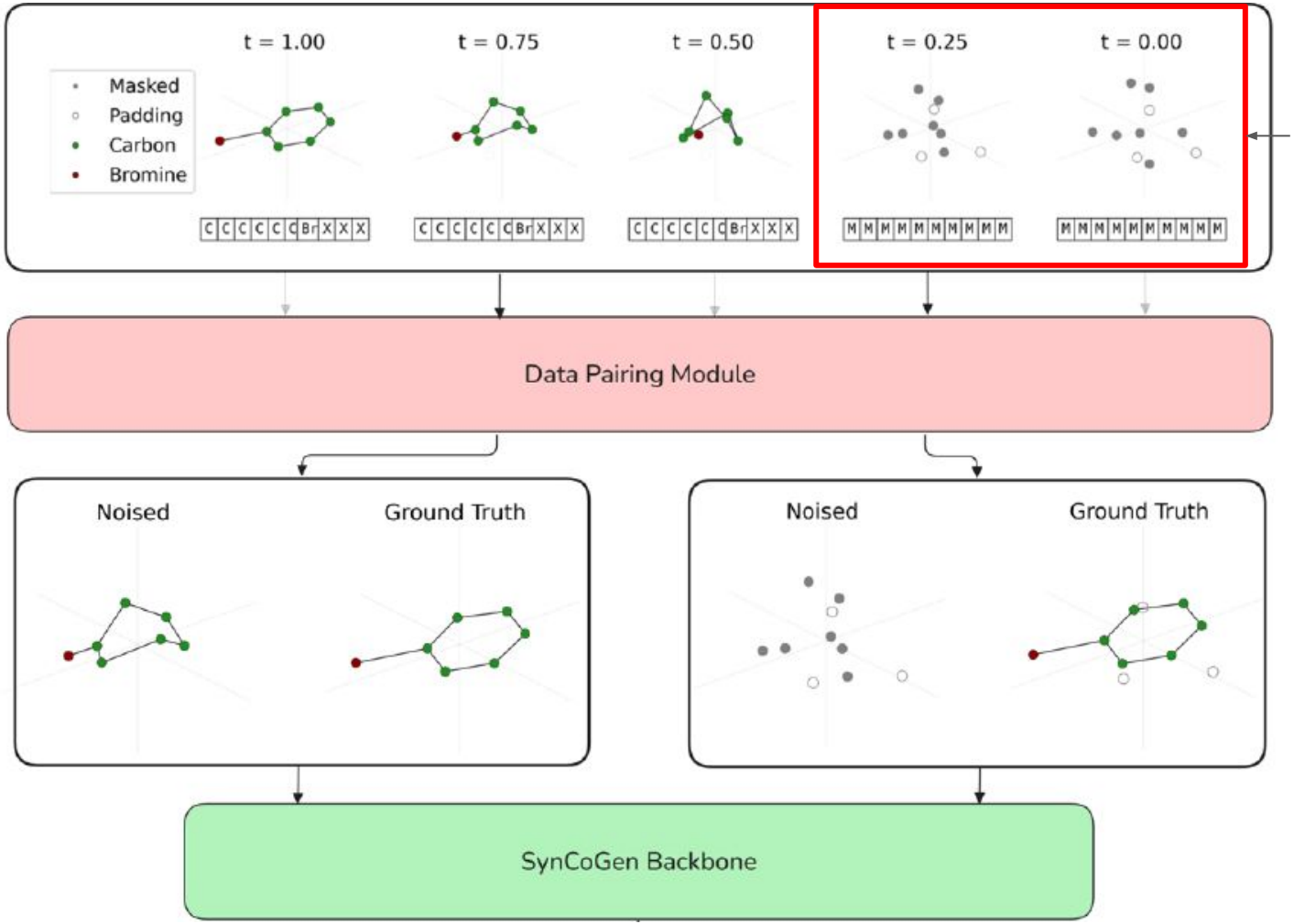
- Want to support **variable atom counts**
  - Even with a prior over #BBs, atom count varies

- At sampling time, extra atoms should be disregarded when no longer necessary
  - E.g, when building block identities are known

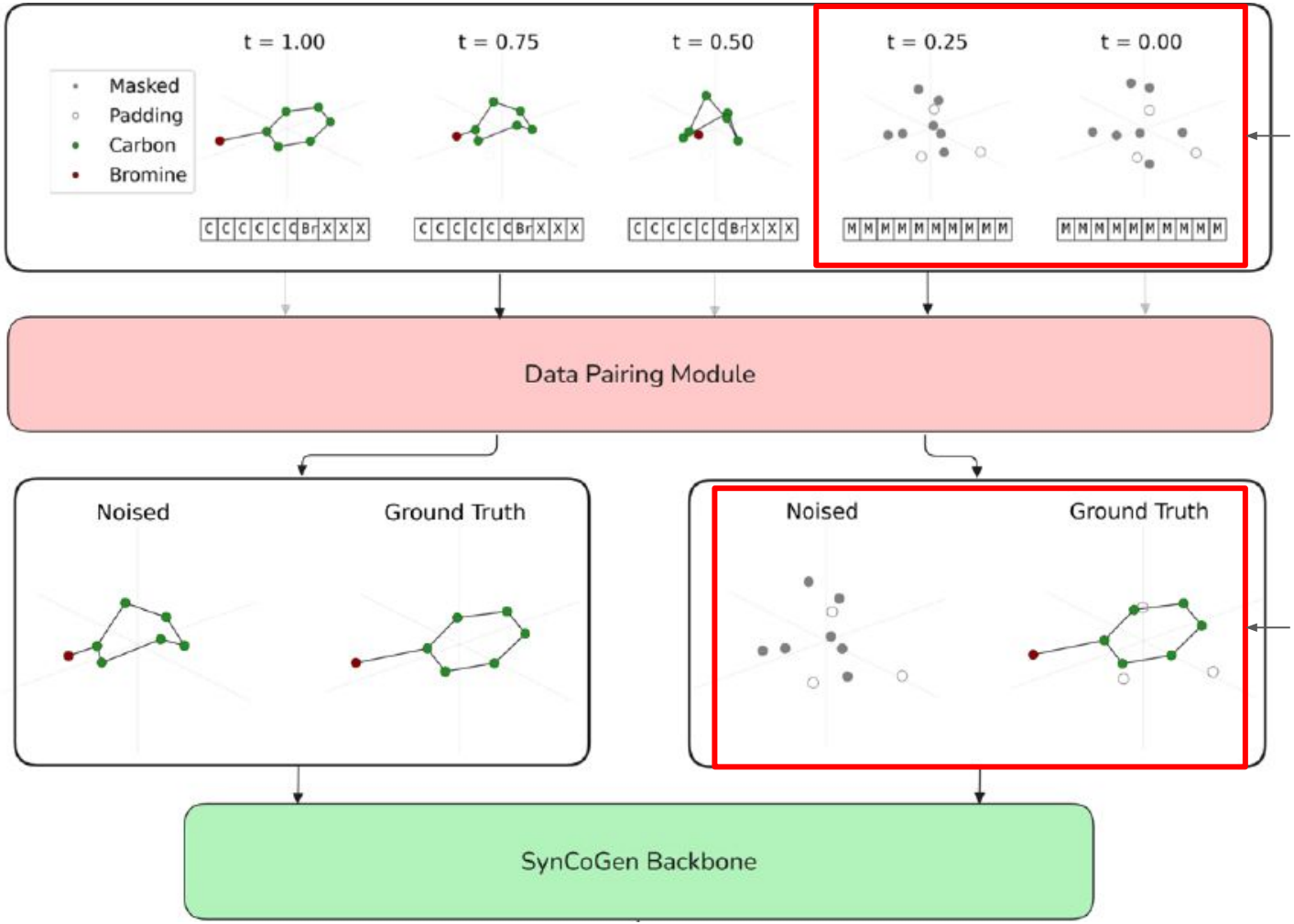
Therefore, sampling should start with an upper bound on the possible number of atoms per building block and progressively remove them as we select BBs

How can we construct a training regime that corresponds to this?


t = 1.00
t = 0.75
t = 0.50
t = 0.25
t = 0.00
Masked
Padding
Carbon
Bromine
C C C C C C Br X X X
C C C C C C Br X X X
C C C C C C Br X X X
M M M M M M M M M M
M M M M M M M M M M
Data Pairing Module
Noised
Ground Truth
Noised
Ground Truth
SynCoGen Backbone



Extra atoms (white) from the Gaussian prior - at this point in sampling, we wouldn't know what BB this will become



Extra atoms (white) from the Gaussian prior - at this point in sampling, we wouldn't know what BB this will become

Train the model to **fix** extra atoms when BB is not known: encourage the model to discriminate between real and extra atoms at **every sampling step**

Losses

The basic loss for SynCoGen is given by

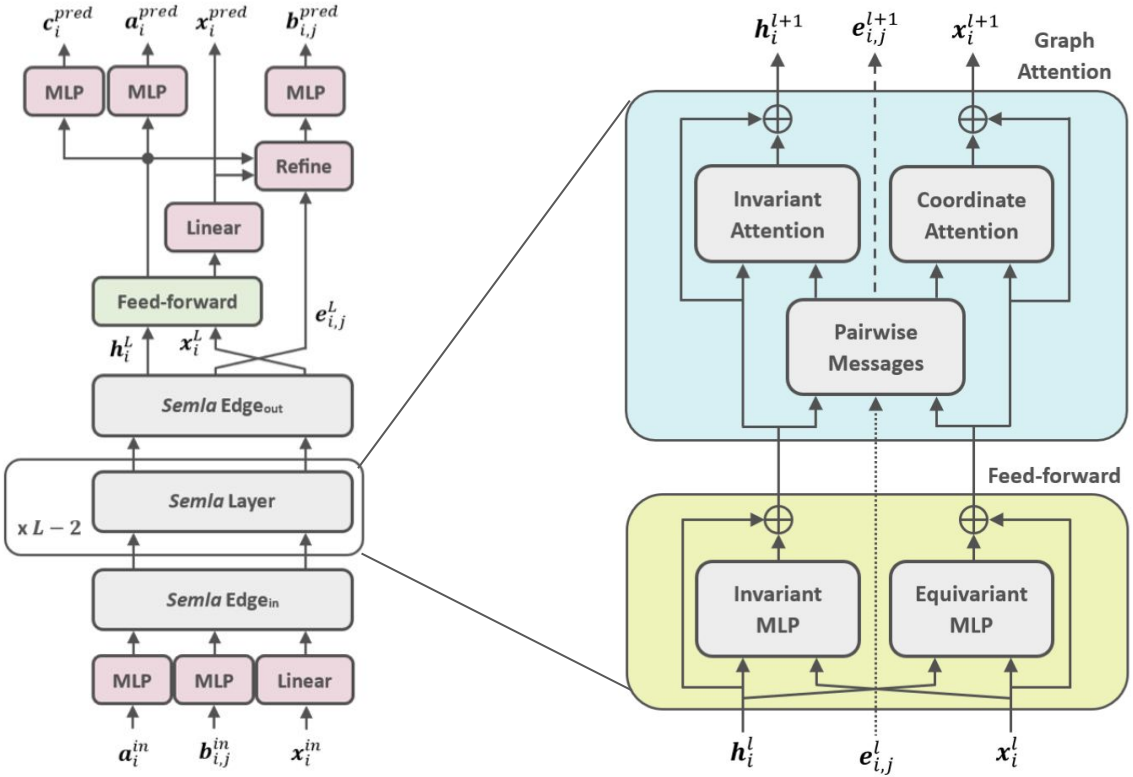
1. **Graph**: NLL of true building block / reaction index
2. **MSE**: Simple MSE between predicted and ground-truth coordinates
3. **Pair**: Thresholded pairwise atomic distance loss

$$\mathcal{L}_{\text{SYNCOGEN}} = \mathcal{L}_{\text{graph}} + \mathcal{L}_{\text{MSE}} + \mathcal{L}_{\text{pair}}.$$

In practice, adding auxiliary bond-length and SmoothLDDT losses also improves model performance slightly

Model Diagram

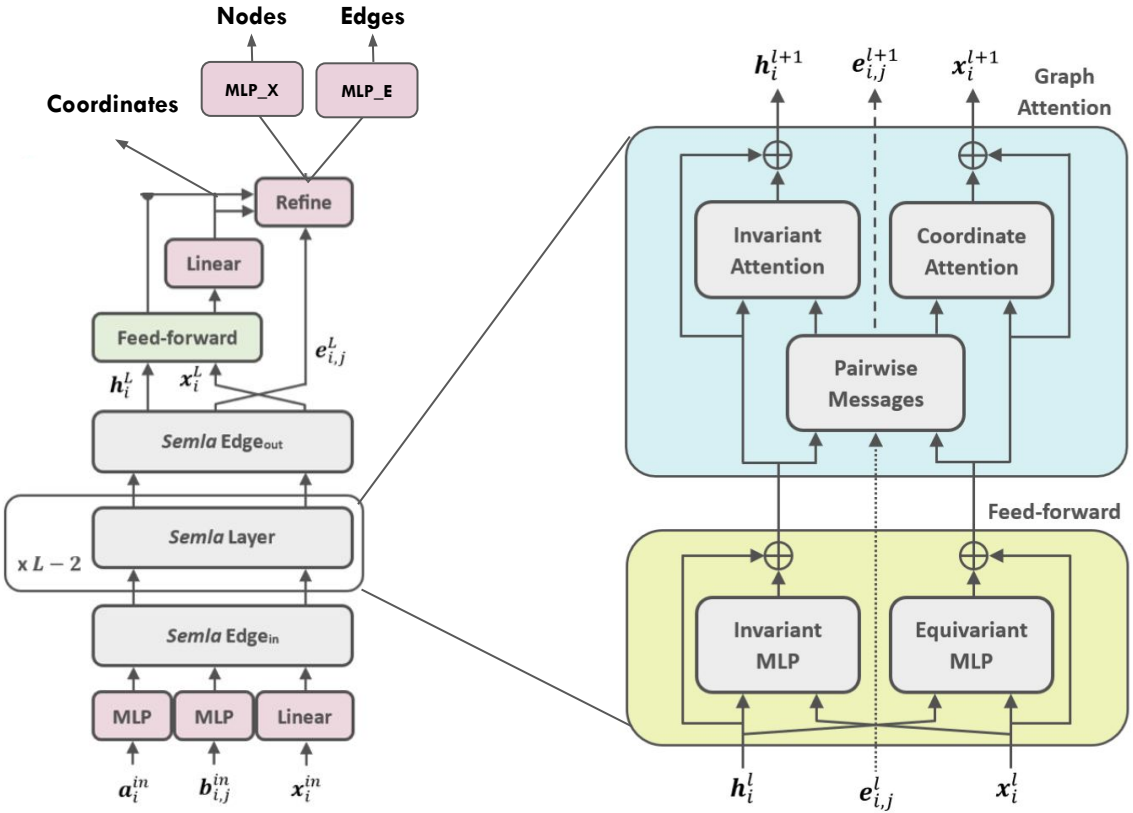
SemlaFlow: Equivariant architecture for atom-based molecular generation



Model Diagram

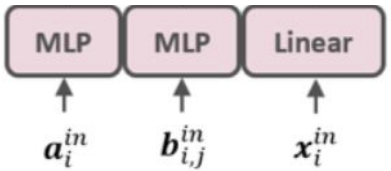
SemlaFlow: Equivariant architecture for atom-based molecular generation

Replace output heads.



Model Diagram: Input

SemlaFlow takes atom-level inputs (bond orders, atom identities, etc)

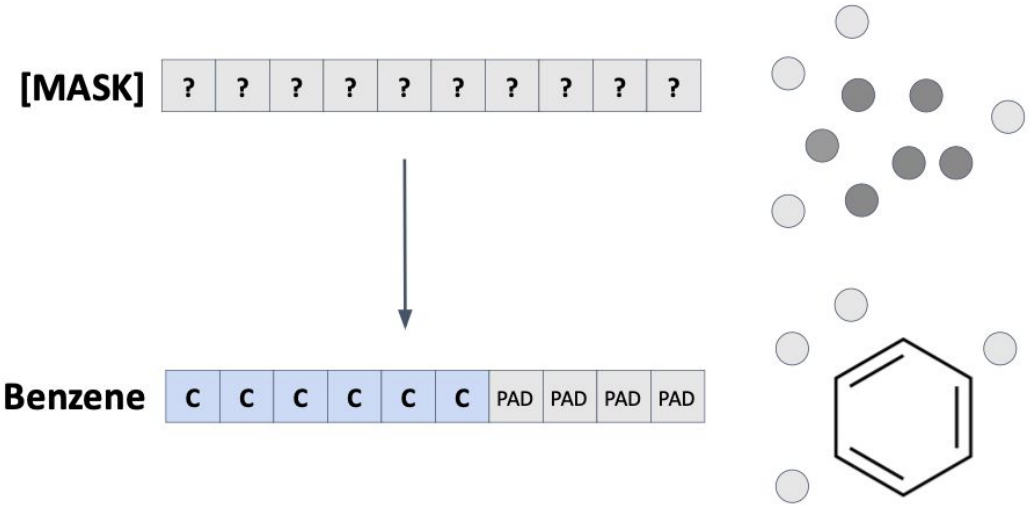


Convert building blocks to all-atom input features

- If building block unmasked: use its true atoms/bond identities (onehot for atomic symbol, bond order)
- If building block masked: use a special “mask” atom symbol and bond order symbol

Inference

1. Sample n < *N* building blocks with *M* atoms, upper bound on # atoms in each building block in our library.
2. Whenever a BB is denoised, mask out extra atoms in all subsequent steps and re-center.



Note: Equivariance

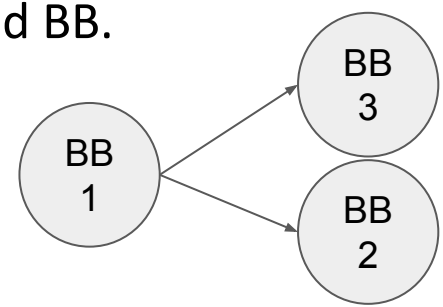
The SemlaFlow backbone is E(3)-equivariant. However, SynCoGen is only E(3)-equivariant **WRT to a molecule** at steps t >= T_u, where T_u is the step at which no padding atoms remain and the molecule is centered with respect to its constituent atoms only.

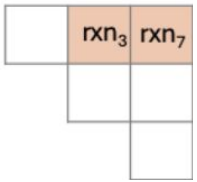
Training/Sampling Tricks

Generation is not autoregressive! Can we still constrain it with what we know?

- **Compatibility**: Given a denoised reaction in *E,* **restrict** source and destination building blocks to compatible ones.

- **Edge Limit**: Only sample one incoming edge per building block - molecule graphs are assembled as trees, so each BB is attached to at most one previously selected BB.



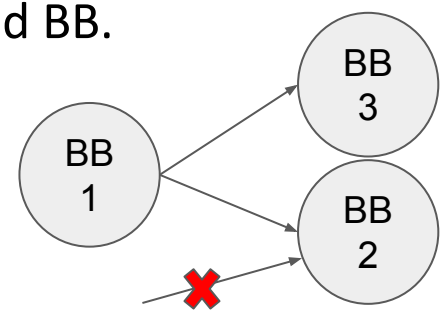


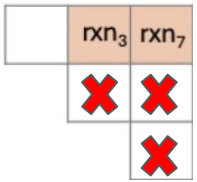
Training/Sampling Tricks

Generation is not autoregressive! Can we still constrain it with what we know?

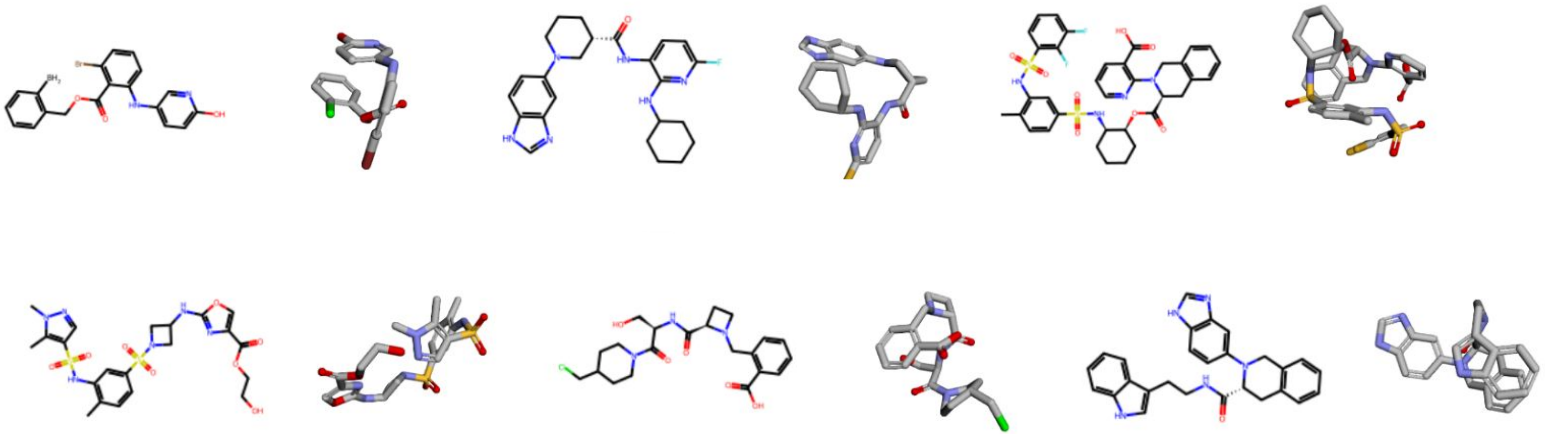
- **Compatibility**: Given a denoised reaction in *E,* **restrict** source and destination building blocks to compatible ones.

- **Edge Limit**: Only sample one incoming edge per building block - molecule graphs are assembled as trees, so each BB is attached to at most one previously selected BB.





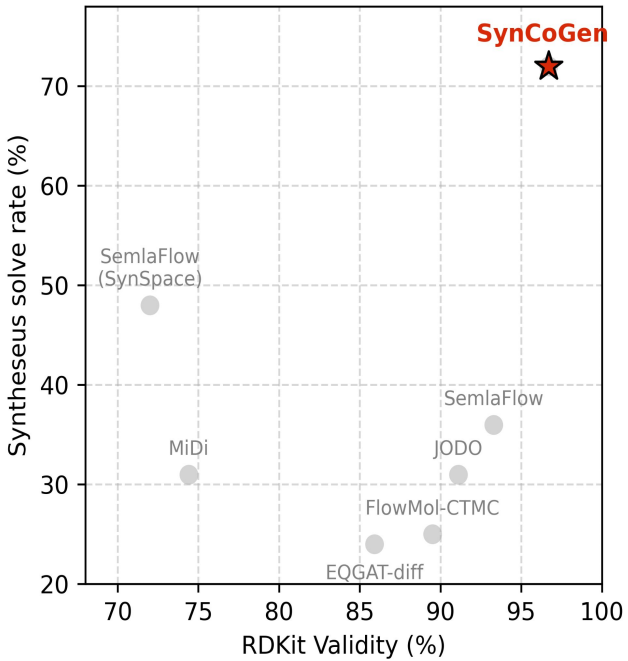
Experiments: Unconditional Generation

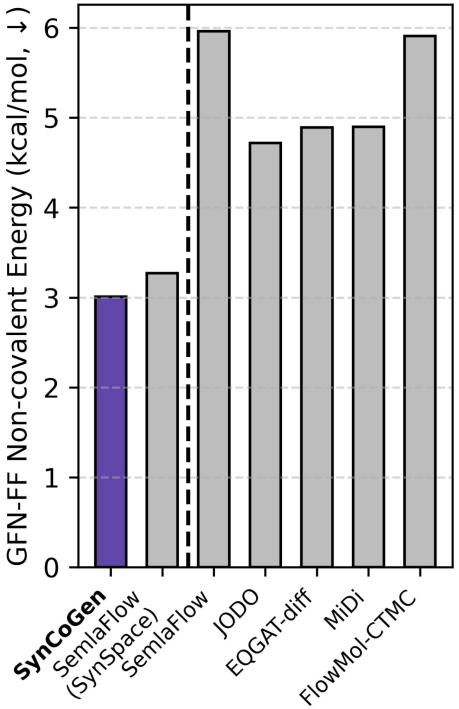


All examples are paired. For each pair, left: 2D generated structure, right: 3D generated conformer.

Experiments: Unconditional Generation

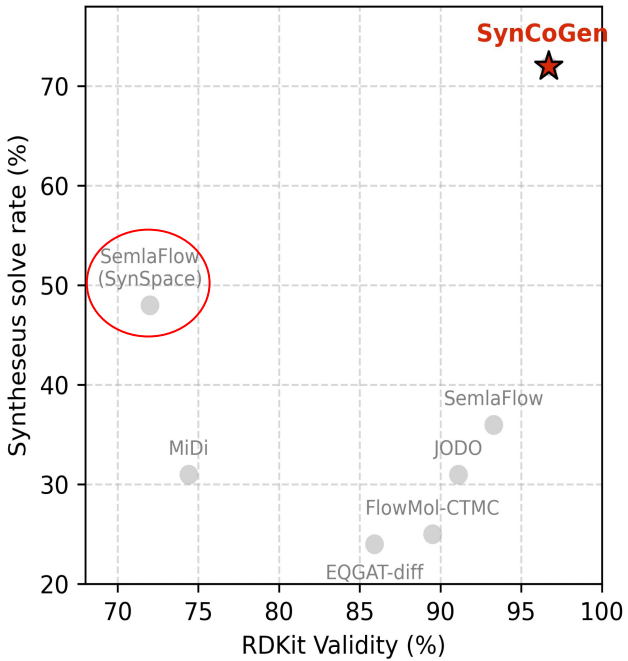
Per 100 molecules sampled by all models.

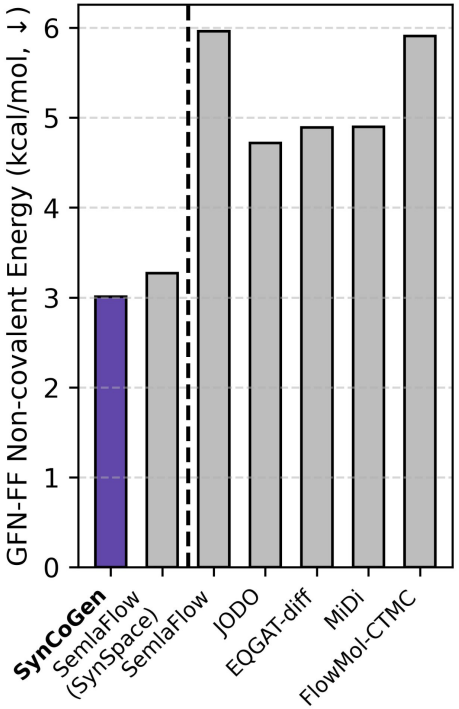




Experiments: Unconditional Generation

Per 100 molecules sampled by all models.



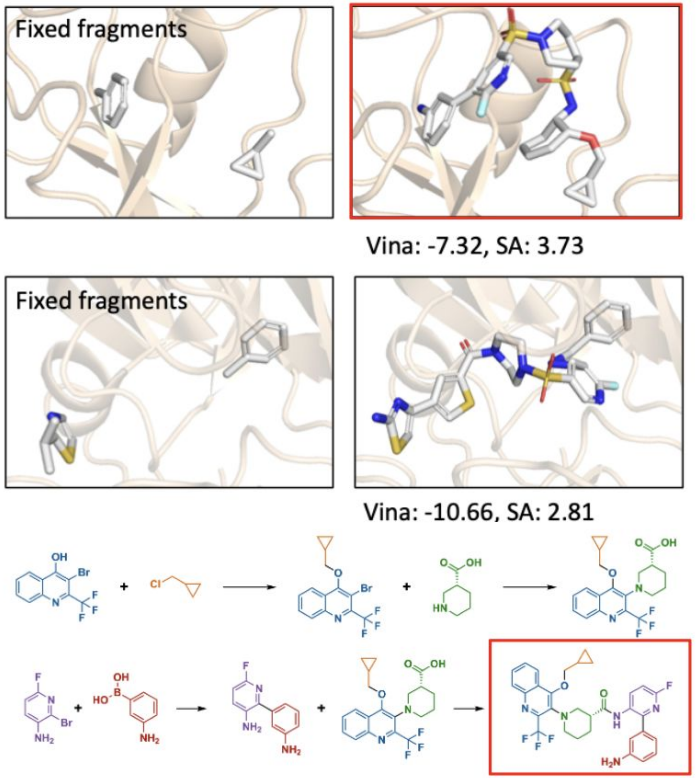


Experiments: Fragment Linking

Top: Fixed-fragment inpainting for PDB 7N7X

Middle: Fixed-fragment inpainting for PDB 4EYR

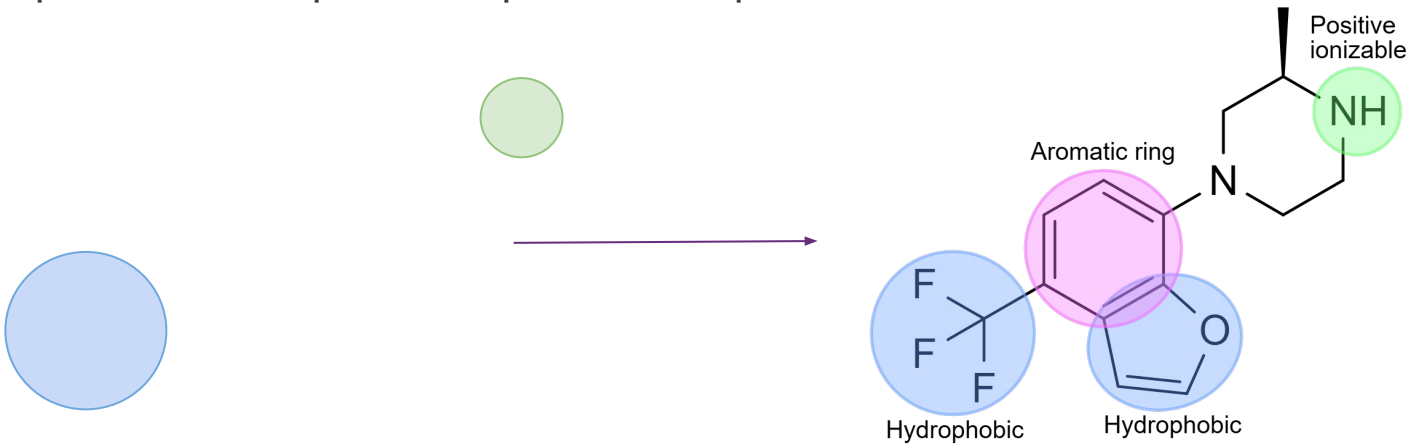
Bottom: Synthesis route proposed by SynCoGen for top molecule



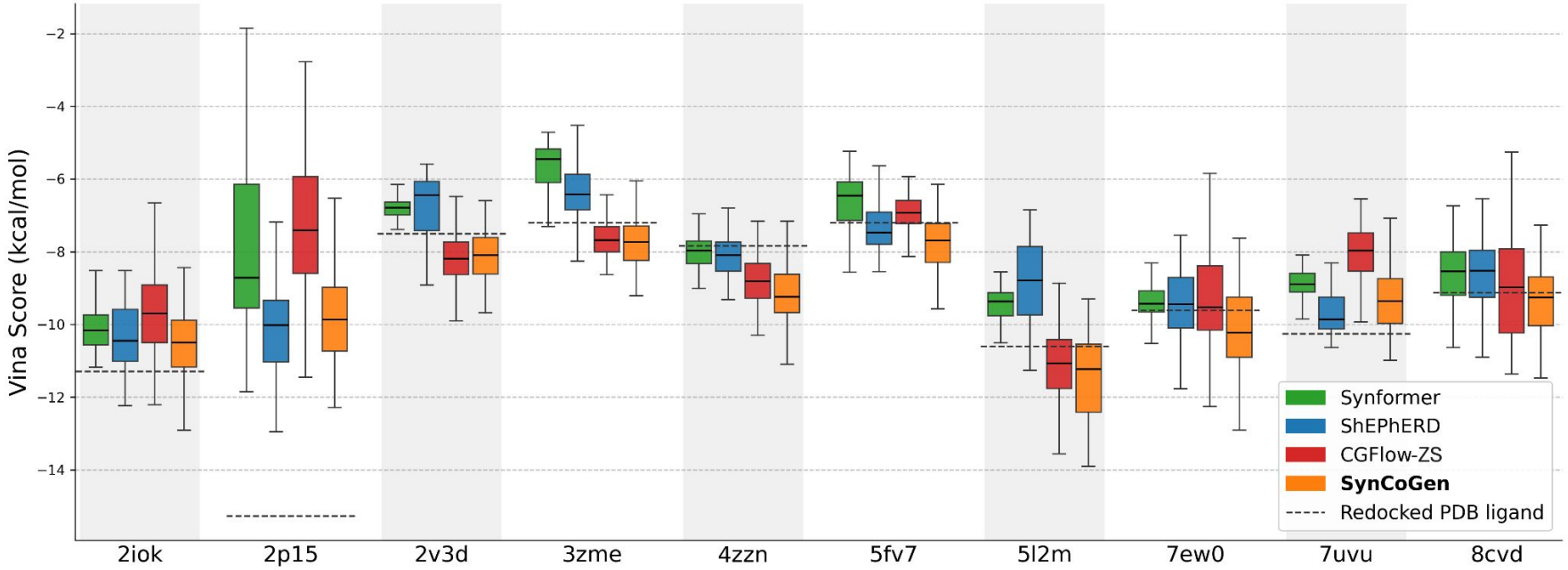
Experiments: Pharmacophore Analog Generation

- **Observation: Conditioning on all pharmacophores causes overfitting.**

  Step 1: Generate Pharmacophores
  Step 2: Subset and shuffle
  Step 3: Train with pharmacophores as input



Results: Docking Scores



Metrics calculated per 100 generated molecules.

Results: Synthetic Route-Finding, 3 Targets

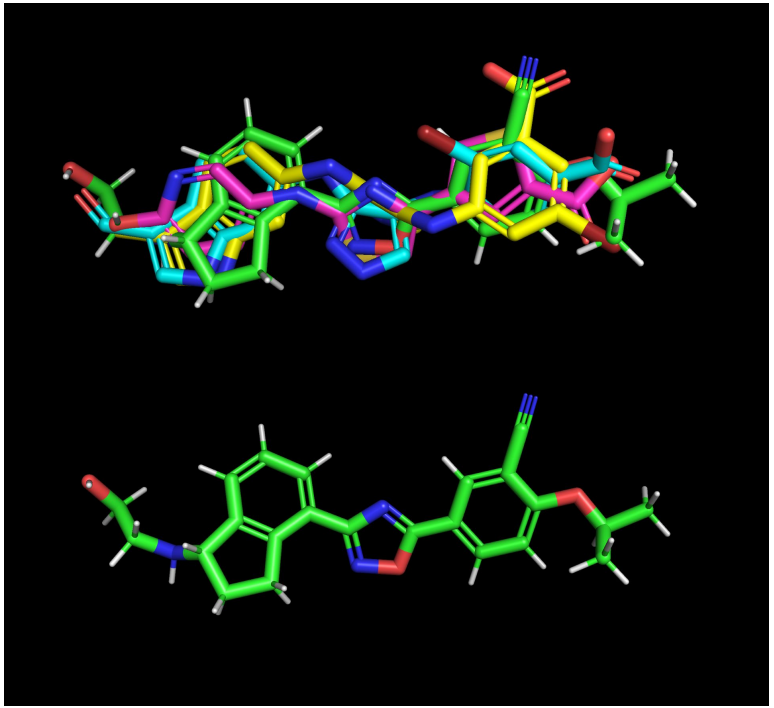
Model	Val. ↑	AiZyn. ↑	Synth. ↑	PB ↑	Div. ↑
SynFormer	**100.0**	34	42	–	0.82
ShEPhERD	38.5	14	12	0.34	**0.86**
CGFlow-ZS	**100.0**	45	16	–	0.75
SYNCOGEN	86.3	**61**	**78**	**0.59**	0.80

Metrics calculated per 100 generated molecules.

Example: 7EW0

Top: Top 3 generated analogs by similarity, overlaid with PDB ligand

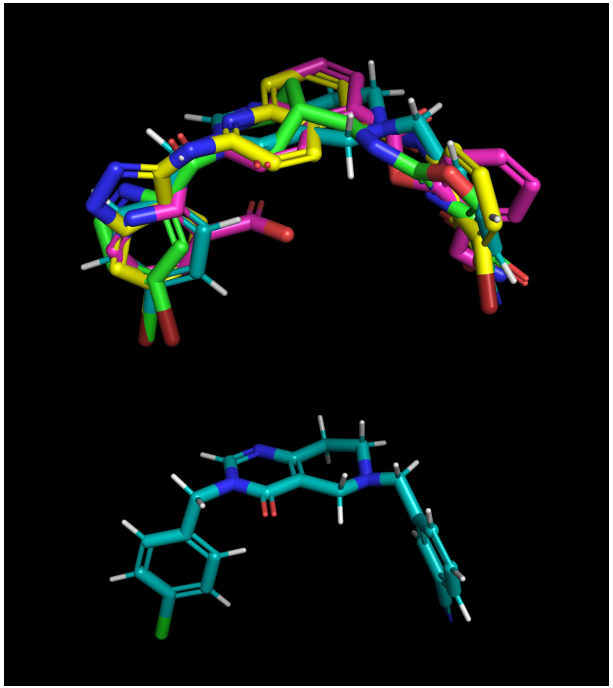
Bottom: PDB ligand only



Example: 7UVU

Top: Top 3 generated analogs by similarity, overlaid with PDB ligand

Bottom: PDB ligand only



Future Work

- Customize building block subsets
  - For example, require generated molecules to contain a particular core or set of cores
- Incorporate protein pocket information for analog design
- Release SynCoGen as a public tool