SYNCOGEN: SYNTHESIZABLE 3D MOLECULE GENERATION VIA JOINT REACTION AND COORDINATE MODELING

Anonymous authors
Paper under double-blind review

ABSTRACT

Ensuring synthesizability in generative small molecule design remains a major challenge. While recent developments in synthesizable molecule generation have demonstrated promising results, these efforts have been largely confined to 2D molecular graph representations, limiting the ability to perform geometry-based conditional generation. In this work, we present SYNCOGEN (Synthesizable Co-Generation), a single framework that combines simultaneous masked graph diffusion and flow matching for synthesizable 3D molecule generation. SYNCOGEN samples from the joint distribution of molecular building blocks, chemical reactions, and atomic coordinates. To train the model, we curated SYNSPACE, a dataset containing over 600K synthesis-aware building block graphs and 3.3M conformers. SYNCOGEN achieves state-of-the-art performance in unconditional small molecule graph and conformer generation. For protein ligand generation in drug discovery, the model delivers competitive performance in both molecular linker design and pharmacophore-conditioned generation across a range of targets. Overall, this multimodal formulation represents a foundation for future applications enabled by non-autoregressive molecular generation, including analog expansion, lead optimization, and direct structure conditioning.

1 INTRODUCTION

Generative models significantly enhance the efficiency of chemical space exploration in drug discovery by directly sampling molecules with desired properties. However, a key bottleneck in their practical deployment is low synthetic accessibility, i.e. generated molecules are often difficult or impossible to produce in the laboratory (Gao & Coley, 2020). To address this limitation, recent work has turned to template-based methods that emulate the chemical synthesis process by constructing synthesis trees that link molecular building blocks through known reaction templates (Koziarski et al., 2024; Cretu et al., 2024; Seo et al., 2024; Gaiński et al., 2025; Gao et al., 2024; Jocys et al., 2024; Swanson et al., 2024). These representations, while useful for downstream experimental validation, do not describe the underlying 3D geometry and thus cannot capitalize on the conformational information that is often crucial for diverse chemical and biological properties.

Parallel advances in generative molecular design have explored spatial modeling at the atomic level. Inspired by advances in protein structure prediction (Yang et al., 2025; Campbell et al., 2024; Wang et al., 2025) and the development of generative frameworks such as diffusion and flow matching, recent work has focused on directly sampling 3D atomic coordinates of small molecules (Hassan et al., 2024; Jing et al., 2022; Fan et al., 2024). These methods learn to generate spatially meaningful, property-aligned conformations along with molecular graphs. The ability to model atomic coordinates directly increases the expressivity of these approaches, enabling applications such as pocket-conditioned generation (Lee & Cho, 2024), scaffold hopping (Torge et al., 2023; Yoo et al., 2024), analog discovery (Sun et al., 2025), and molecular optimization (Morehead & Cheng, 2024). However, without considering practical synthesis routes, integrating synthesizability constraints into these models remains a major challenge, and most existing 3D generative approaches do not ensure that proposed molecules can be made in practice.

This work introduces SYNCOGEN (Synthesizable Co-Generation), a generative modeling framework aiming to bridge the gap between 3D molecular generation and practical synthetic accessibility (Figure 1). Our main contributions are as follows:

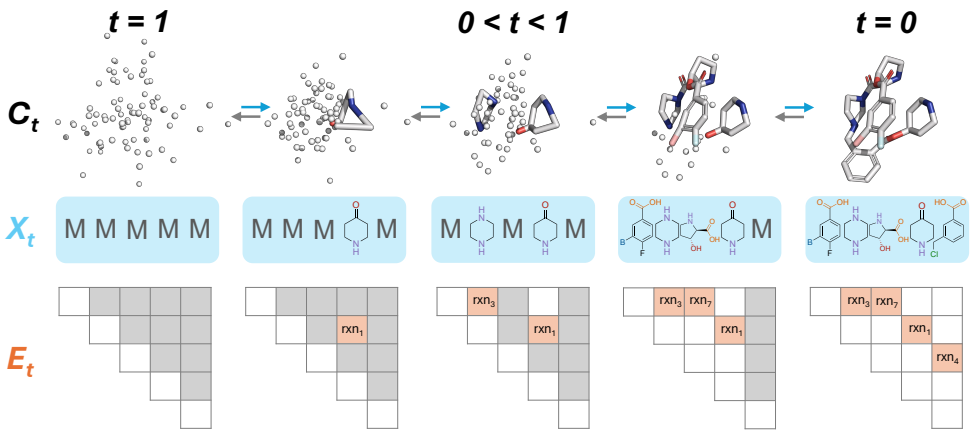


Figure 1: SYNCOGEN is a simultaneous masked graph diffusion and flow matching model that generates synthesizable molecules in 3D coordinate space. Each node corresponds to a building block, and edges encode chemical reactions. Note that nodes are not necessarily linear and that the leaving groups are not displayed.

- **Generative Framework:** We propose a novel generative framework that combines masked graph diffusion with flow matching in unified time to jointly sample from the distribution over building block reaction graphs and of 3D coordinates, tying structure- and synthesis-aware modeling.
- **Molecular Dataset:** We curate a new dataset SYNSPACE, comprising 622,766 synthesizable molecules represented as building block reaction graphs, along with 3,360,908 associated low-energy conformations. Compared to synthon-based datasets, this dataset enables models to generate more readily synthesizable molecules and directly suggest streamlined synthetic routes.
- **Empirical Validation:** We demonstrate that SYNCOGEN achieves state-of-the-art performance in 3D molecule generation, producing physically realistic conformers while explicitly tracing reaction steps. Ablations show our modelling choices are crucial for the performance. Importantly, SYNCOGEN performs 3D conditional molecular generation tasks including linker design and pharmacophore-conditioned generation, highlighting its applicability for drug discovery.

2 BACKGROUND AND RELATED WORK

Flow Matching. Given two distributions ρ_0 and ρ_1, and an interpolating probability path ρ_t such that $\rho_{t=0} = \rho_0$ and $\rho_{t=1} = \rho_1$, flow matching (Lipman et al., 2023; Albergo et al., 2023; Liu et al., 2023; Peluchetti, 2023; Tong et al., 2023) aims to learn the underlying vector field u_t that generates ρ_t. Since u_t is not known in closed form, flow matching instead defines a conditional probability path $\rho_{t|1}$ and its corresponding vector field $u_{t|1}$. The marginal vector field u_t can then be learnt with a parametric v_θ by regressing against $u_{t|1}$ with the CFM objective:

$$\mathcal{L}_{\text{CFM}}(\theta) = \mathbb{E}_{t,\mathbf{x}_1\sim\rho_1,\mathbf{x}\sim\rho_{t|1}(\cdot|x_1)}||v_t(\mathbf{x};\theta) - u_{t|1}(\mathbf{x}|\mathbf{x}_1)||^2 \tag{1}$$

Masked Discrete Diffusion Models. Let $\mathbf{x} \sim \rho_{\text{data}}$ be a one-hot encoding over K categories. Discrete diffusion models (Austin et al., 2021; Sahoo et al., 2024; Shi et al., 2024) map the complex data distribution ρ_{data} to a simpler distribution via a Markov process, with absorbing (or masked) diffusion being the most common. In the masked diffusion framework, the forward interpolation process $(\rho_t)_{t\in[0,1]}$ with the associated noise schedule $(\alpha_t)_{t\in[0,1]}$ results in marginals $q(\mathbf{z}_t|\mathbf{x}) = \text{Cat}(\mathbf{z}_t; \alpha_t\mathbf{x} + (1-\alpha_t)\mathbf{m})$, where $\mathbf{z}_t$ and $\mathbf{m}$ denote intermediate latent variables and the one-hot encoding for the special [MASK] token, respectively. The posterior can be derived as:

$$q(\mathbf{z}_s|\mathbf{z}_t,\mathbf{x}) = \begin{cases} \text{Cat}(\mathbf{z}_s;\mathbf{z}_t), & \mathbf{z}_t \neq \mathbf{m} \\ \text{Cat}(\mathbf{z}_s;\frac{(1-\alpha_t)\mathbf{m}+(\alpha_s-\alpha_t)\mathbf{x}}{1-\alpha_t}), & \mathbf{z}_t = \mathbf{m} \end{cases} \tag{2}$$

The optimal reverse process $p_\theta(z_s \mid z_t)$ takes the same form but with $x_\theta(z_t, t)$ in place of the true $\mathbf{x}$. We adopt the zero-masking and carry-over unmasking modifications of Sahoo et al. (2024).

Multimodal Generative Models. Multimodal data generation (e.g. text-images, audio-vision, sequences/atomic types and 3D structures) represents a challenging frontier for generative models and has seen growing interest in recent times. Current approaches for this task typically either – 1) tokenize multimodal data into discrete tokens, followed by a autoregressive generation (Meta, 2024; Xie et al., 2024; Lu et al., 2024), or 2) utilize diffusion / flow models for each modality in its native space (Lee et al., 2023; Zhang et al., 2024; Campbell et al., 2024; Irwin et al., 2025). Diffusion and flow models also offer flexibility in terms of coupled (Lee et al., 2023; Irwin et al., 2025) or decoupled (Campbell et al., 2024; Bao et al., 2023; Kim et al., 2024) diffusion schedules across modalities. SYNCOGEN uses a coupled diffusion schedule but at two resolutions, with discrete diffusion for graphs of building blocks and reactions, and a flow for atomic coordinates in building blocks.

3D Molecular Generation. Several recent works (Irwin et al., 2025; Le et al., 2024; Vignac et al., 2023; Huang et al., 2023; Dunn & Koes, 2024) have studied unconditional molecular structure generation by sampling from the joint distribution over atom types and coordinates. However, these models lack the ability to constrain the design space to synthetically accessible molecules. In concurrent work, (Shen et al., 2025) uses generated 3D structures to guide GFlowNet policies in designing the graph of *synthon*-based linear molecules, but does not account for structural quality.

Synthesizable Molecule Generation. Beyond directly optimizing synthesizability scores (Liu et al., 2022; Guo & Schwaller, 2025) – which are often unreliable – the predominant approach to ensuring synthetic accessibility involves modifying generative models to incorporate reaction templates. Early methods explored autoencoders (Bradshaw et al., 2019; 2020), genetic algorithms (Gao et al., 2022), and reinforcement learning (Gottipati et al., 2020; Horwood & Noutahi, 2020). More recently, GFlowNet-based (Koziarski et al., 2024; Cretu et al., 2024; Seo et al., 2024; Gaiński et al., 2025) and transformer-based (Gao et al., 2024; Jocys et al., 2024) methods have gained prominence. Such generative models have already shown practical utility in biological discovery tasks (Swanson et al., 2024). However, most methods only generate molecular graphs and do not produce 3D structures. The recent CGFlow Shen et al. (2025) performs conditional 3D generation via a GFlowNet policy augmented with flow matching; however, CGFlow requires a full training for each target pocket.

3 DATASET

Training a synthesizability-aware model to co-generate both 2D structures and 3D positions requires a dataset of easily synthesizable molecules in an appropriate format. In addition to atomic coordinates, this includes a graph-based representation from which plausible synthetic pathways can be inferred. A common approach is to use synthons—theoretical structural units that can be combined to form complete molecules(Baker et al., 2024; Grigg et al., 2025; Medel-Lacruz et al., 2025). Synthon-based representations do not guarantee the existence of a valid synthesis route, and they do not directly provide one even if it exists. Moreover, they lack the flexibility to constrain the reaction space, which is often critical when prioritizing high-yield, high-reliability reactions or operating within the limits of automated synthesis platforms such as self-driving labs (Abolhasani & Kumacheva, 2023).

Alternatively, many synthesis-aware generators employ external reaction simulators, such as RDKit, to couple building blocks iteratively. While convenient, such black-box steps offer no fine-grained control when a reagent has multiple *reaction centers*, distinct atoms or atom sets that can each serve as the specific site of bond formation or cleavage in a coupling reaction. They also do not define atom mappings between reactants and products, making it impossible to trace product atoms back to their parent building blocks, which complicates edge assignment in building block graph generation. To overcome these limitations, we curate a new dataset SYNSPACE (Figure 2) comprising building block-level reaction graphs pairs with corresponding atom- and block-level graphs. We then calculate multiple 3D conformations for each graph using semi-empirical methods (Bannwarth et al., 2019).

3.1 SYNSPACE: GRAPH GENERATION

We begin by constructing a vocabulary of 93 commercially available, low-cost building blocks and 19 high-yield reaction templates. This vocabulary is adapted from the collection proposed by Koziarski et al. (2024), retaining reactions that (1) ensure all product atoms originate from the two input reagents, and (2) involve at most one leaving group per reagent. We emphasize that these are not only feasible chemistries, but also simple and reliable chemical reactions with readily available building blocks that can enable rapid multi-step synthesis.

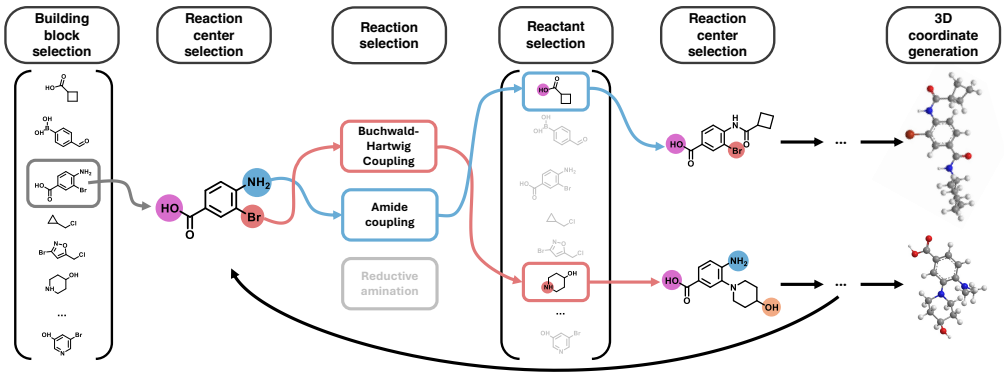


Figure 2: **Graphical overview of SYNSPACE creation process.** Highly synthesizable molecules are procedurally constructed by sampling synthesis pathways from a predefined set of building blocks and reactions. Starting from an initial building block, the procedure selects a reaction center, a compatible reaction, and a suitable reactant. This process is iteratively repeated for a fixed number of reaction steps. After the final structure is assembled, multiple low-energy 3D conformations are generated.

We procedurally generate SYNSPACE from this vocabulary by iteratively coupling building block graphs at their respective reaction centers with compatible chemical reaction templates, described in detail in Appendix A.2. We obtained 622,766 building block reaction graphs, each constructed from 2 to 4 sequential reactions. For each resulting molecule, we generate multiple low-energy conformations and retain their atomic coordinates, with a total of 3,360,908 conformations.

Note: Injectivity. Many commercially available building blocks contain multiple reaction centers, each compatible with a different set of corresponding reaction centers on other building blocks. In this way, a building block-level reaction graph $G_b = (X, E)$ is not fully specified when edges are parametrized by the reaction alone. To achieve an injective correspondence, we therefore label edges from node i to $j > i$ by the triple $e_{ij} = (r, v_i, v_j)$, where r is the coupling reaction and (v_i, v_j) are the participating reaction centers on the source and destination blocks, respectively. Strictly speaking, stereoisomers that form during reactions collapse to the same (X, E) representation, but this granularity suffices for the scope of the current work.

3.2 SYNSPACE: CONFORMATION GENERATION

For each molecular graph, 50 initial conformers were generated using the ETKDG (Riniker & Landrum, 2015) algorithm (RDKit implementation). These structures were energy-minimized using the MMFF94 force field, and all conformers within 10 kcal/mol of the global minimum were retained. The resulting geometries were then re-optimized with the semi-empirical GFN2-xTB (Bannwarth et al., 2019) method, after which the same 10 kcal/mol energy threshold was applied. At every stage, redundant structures were removed by geometry-based clustering (RMSD < 1.5). This workflow yields, on average, 5.4 distinct conformers per graph. Relative to exhaustive approaches such as CREST (Pracht et al., 2024), the workflow is several orders of magnitude faster; despite occasionally omitting some conformations, the retained structures are diverse and reproduce the bond-length, bond-angle, and dihedral-angle distributions observed in CREST-derived datasets (see Section 5.1).

3.3 SYNSPACE: PHARMACOPHORE GENERATION

For each conformer associated with a molecule in SYNSPACE, we generate a pharmacophore profile consisting of one-hot pharmacophore types $X_{\text{pharm}} \in \{0,1\}^{N_{\text{pharm}} \times N_{\text{types}}}$ and positions $C_{\text{pharm}} \in \mathbb{R}^{N_{\text{pharm}} \times 3}$ using ShePhERD Adams et al. (2025). Here, N_{pharm} and N_{types} correspond to the number of pharmacophore features and the number of pharmacophore types, respectively.

4 METHODS

Notation. Let $\mathcal{B}$ be the building-block vocabulary and $\mathcal{R}$ the set of reaction templates, with cardinalities $B := |\mathcal{B}|$ and $R := |\mathcal{R}|$. We write N for the maximum number of building blocks that any molecule in the training set can contain, and M for the maximum number of atoms in a single building block. For each block $b \in \mathcal{B}$ we denote its set of reaction-center atoms by $\mathcal{V}(b)$; the

global maximum of these counts is $V_{\max} := \max_{b\in\mathcal{B}}|\mathcal{V}(b)|$. Hence, tensor shapes contain factors such as $B+1$ (to accommodate the masked token π_X in X), $RV_{\max}^2+2$ (to accommodate the no-edge and masked tokens λ_E and π_E), together with the bounds N and M introduced above. For any coordinate tensor C and binary mask S we define the mask-weighted centroid and its centered version by $\bar{C}_S := \frac{\sum S\odot C}{\sum S}, \qquad \tilde{C} := C - \bar{C}_S$.

SYNCOGEN. SYNCOGEN generates building block-level reaction graphs and coordinates. Each molecule is represented by a triple (X, E, C) where $X \in \{0,1\}^{N\times|\mathcal{B}|+1}$ encodes the sequence of building-block identities, $E \in \{0,1\}^{N\times N\times|\mathcal{R}|V_{\max}^2+2}$ labels the coupling reaction (and centers) between every building block pair, and $C \in \mathbb{R}^{N\times M\times 3}$ stores all atomic coordinates. We detail the parameterization of graphs (X, E) in Appendix B.1. Training combines two diffusion schemes: 1) a **discrete absorbing process** on (X, E) using the categorical forward kernel of Sahoo et al. (2024), and 2) a **continuous, visibility-aware process** on C whose endpoints are (i) a rototranslationally-aligned isotropic Gaussian and (ii) a re-centered ground truth, considering all "visible" atoms in the prior (see Section 4.2). The code is available here.

4.1 MODEL ARCHITECTURE

We adapt $SE(3)$ equivariant architecture originally designed for all-atom molecular design (SEMLAFLOW (Irwin et al., 2025)), as the principal backbone to generate both coordinates and graphs. At each timestep t, SYNCOGEN predicts building block logits L_t^X, L_t^E and a shifted coordinate estimate $\hat{\tilde{C}}_0^t$. The loss is the weighted sum of the cross-entropy term $\mathcal{L}_{\text{graph}}$ on (X, E), the masked coordinate MSE term $\mathcal{L}_{\text{MSE}}$, and the short-range pairwise distance term $\mathcal{L}_{\text{pair}}$ (see Appendices B.4 and B.13 for details). We define additional building-block-to-atom featurization in Appendix B.2 and atom-to-building-block output layers in Appendix B.7.

Pharmacophore Conditioning Backbone. To accommodate pharmacophores as conditioning information, we design a modified backbone to represent each as an "atom" with no weight during centering operations. After atom featurization, pharmacophore types are fed through a separate featurization head and concatenated to invariant atom type features, i.e. $X_{\text{model}} = [\text{MLP}_{\text{atom}}(X_{\text{atom}}), \text{MLP}_{\text{pharm}}(X_{\text{pharm}})] \in \mathbb{R}^{(N+N_{\text{pharm}})\times d_x}$. Pharmacophore coordinates are concatenated directly to atomic coordinates, $C_{\text{model}} = [C, C_{\text{pharm}}] \in \mathbb{R}^{(N+N_{\text{pharm}})\times 3}$, and therefore undergo identical data augmentation beforehand (including that induced by data pairing, see Section 4.2). C_{model} and X_{model} are then passed to the equivariant-invariant dynamics module. Prior to final output layers, expanded atom-level hidden-layer outputs are truncated to the number of atoms $N \times M$.

4.2 NOISING SCHEMES

Graph Noising. We corrupt true graphs (X_0, E_0) using the procedure described in Section 2. In practice, as all true edge matrices E_0 are symmetric, we symmetrize the sampled probabilities for the noising and denoising of E_t correspondingly (see Appendix B.8).

Coordinate Noising. For any time t where X_t contains a masked building block, we cannot distinguish any of its M possible atoms from padding. For this reason, we design a *visibility mask* S_t that considers all M atoms for each noised building block at time t as valid. We then center the prior by its visibility-masked centroid $\tilde{C}_1 = C_1 - \bar{C}_{1S_t}$. Here, all atoms $a \in \text{supp}(S_t) \setminus \text{supp}(S_0)$ are potentially valid at time t, but represent padding indices in the true molecule.

We thus must interpolate a data-prior pair $(\tilde{C}_1, \tilde{C}_0)$ that contains a consistent number of valid atoms $|S_t|$ by which both $\tilde{C}_0$ and $\tilde{C}_1$ are centered. To handle atoms that do not appear in C_0, we record their points in $\tilde{C}_1$, re-center C_0 by the same visibility-masked centroid, then copy the atoms to their respective indices in C_0. Essentially, we task the model with rearranging the true atoms while disregarding padding by learning to fix padding atoms in place. See Algorithm 2 for formalization. We note a caveat in equivariance in Appendix B.3.

Flexible Atom Count. Most 3D molecule generation methods require specifying the number of atoms during inference. Because the prior of SYNCOGEN is over building blocks, we naturally handle a flexible number of atoms during generation and model any excessive atoms as ghost atoms.

4.3 TRAINING-TIME CONSTRAINTS

For discrete diffusion, SYNCOGEN utilizes zero masked logit probabilities and logit unmasking. In addition, we implement the following:

1. **No-Edge Diagonals.** We set the diagonals of all edge logit predictions L_θ^E to no-edge, as no building block has a coupling reaction-induced bond to itself.
2. **Edge Count Limit.** Let $k_t := \sum_{1 \le i < j \le n} \mathbb{1}(E_t[i, j, \cdot] \notin \{\pi_E, \lambda_E\})$ be the number of unmasked true edges in the upper triangle of E_t. If $k_t = n - 1$, we have the correct number of edges for a molecule containing n building blocks and therefore set all remaining edge logits to λ_E.
3. **Compatibility Masking.** Assume that for some E_t an edge entry is already denoised, $E_t[i, j, \cdot] = (r, v_i, v_j)$, meaning that building block i reacts with building block j via reaction r and centers $v_i \in \mathcal{V}(X_i)$, $v_j \in \mathcal{V}(X_j)$. Define the sets of *center-matched reagents*

$$\begin{aligned} \mathcal{B}^A_{r,v} &:= \{\, b \in \mathcal{B} \mid (b, v) \text{ matches reagent A in } r \}, \\ \mathcal{B}^B_{r,v} &:= \{\, b \in \mathcal{B} \mid (b, v) \text{ matches reagent B in } r \}. \end{aligned} \tag{3}$$

   For every node slot i (resp. j) we construct a $|\mathcal{B}|$-dimensional binary mask

$$\mathcal{X}_{i,k} = \mathbb{1}[b_k \in \mathcal{B}^A_{r,v_i}], \mathcal{X}_{j,k} = \mathbb{1}[b_k \in \mathcal{B}^B_{r,v_j}], k = 1, \ldots, |\mathcal{B}|. \tag{4}$$

   so that the soft-max for $X_t[i, \cdot]$ (resp. $X_t[j, \cdot]$) is evaluated only over the 1-entries of $\mathcal{X}_i$ (resp. $\mathcal{X}_j$). Analogously, once a node identity $X_t[j] = b$ is denoised, incoming edge channels (i, j) with $j > i$ are masked to reactions $e = (r, v_i, v_j)$ such that $b \in \mathcal{B}^B_{r,v_i}$.

Put simply, we restrict logits to disallow loops (e.g. macrocycles), to impose a limit on the number of edges, and to better ensure the selection of chemically compatible building blocks and reactions.

4.4 SAMPLING

Sampling begins by drawing a building block count $n \sim \mathrm{Cat}(\pi_{\text{frag}})$, setting the node and edge tensors to the masked tokens, $X_1[i,\cdot] = \pi_X,\ E_1[i,j,\cdot] = \pi_E$ for every $0 \le i, j < N$, and padding all $(i \ge n)$ rows/columns with the no–edge token λ_E. The initial coordinates are an isotropic Gaussian $C_1 \sim \mathcal{N}(0, I)^{N \times M \times 3}$. From this state, the sampler walks backwards in diffusion time, and at each step it (i) recenters the current coordinates by the visibility mask S_t derived from X_t, (ii) generates node and edge logits and coordinate predictions with the trained model, (iii) draws the next discrete state from (ii), and (iv) updates coordinates via an Euler step. After a final, deterministic pass, we calculate $\left(\hat{X}_0, \hat{E}_0\right) = \arg\max_k L_\theta^E[\cdots, k]$ and center the coordinates to yield the molecule $(\hat{X}_0, \hat{E}_0, \hat{C}_0)$. Complete pseudocode is provided in Appendix B.5. We note our discrete and continuous schemes share a unified time. Lastly, we find inference annealing on the coordinates (see Appendix D.2) yields small performance gains at sampling time.

Note: Inference-Time Edge Constraints. By construction, a molecule containing n connected building blocks contains exactly $n - 1$ edges, and building block $j > 0$ has a single unique parent $i < j$. Consequently, sampling of redundant or impossible edges can be eliminated at inference time as described in Appendix B.6.

5 EXPERIMENTS

5.1 *De Novo* 3D MOLECULE GENERATION

We first study SYNCOGEN in unconditional molecule generation jointly with 3D coordinates and reaction graphs. We evaluate SYNCOGEN against several recently published all-atom generation frameworks which produce 3D coordinates, including SemlaFlow (Irwin et al., 2025), EQGAT-Diff (Le et al., 2024), MiDi (Vignac et al., 2023), JODO (Huang et al., 2023), and FlowMol (Dunn & Koes, 2024). To isolate modeling effects from data, we retrain SemlaFlow on atomic types/coordinates in SYNSPACE for the same number of epochs as SYNCOGEN.

For each model, we sample 1000 molecules and compute stringent metrics capturing chemical soundness, synthetic accessibility, conformer quality, and distributional fidelity. Pertaining to the molecular graphs, we report the RDKit sanitization validity (Valid.) and retrosynthetic solve rate (AiZynthFinder (Genheden et al., 2020) (AiZyn.) and Syntheseus (Maziarz et al., 2025) (Synth.)). For conformers, we introduce two physics-based metrics: the median non-bonded interaction energies per

atom via the forcefield method GFN-FF and via the semiempirical quantum chemistry method GFN2-xTB Bannwarth et al. (2019); Spicher & Grimme (2020); we also check PoseBusters (Buttenschoen et al., 2024) validity rate (PB). We evaluate the diversity (Div.) as the average pairwise Tanimoto dissimilarity of the Morgan2 fingerprints, novelty (Nov.) as the percentage of candidates not appearing in the training set, and the Fréchet ChemNet Distance (Preuer et al., 2018) (FCD) between generated samples and the training distribution. See Appendix D.3 for details.

Table 1: **Comparison of generative methods for *de novo* 3D molecule generation.**

		Primary metrics					Secondary metrics			
Group	**Method**	Valid. ↑	AiZyn. ↑	Synth. ↑	GFN-FF ↓	xTB ↓	PB ↑	FCD ↓	Div. ↑	Nov. ↑
Rxns & coords	SYNCOGEN	**96.7**	**50**	**72**	**3.01**	**-0.91**	**87.2**	**2.91**	0.78	93.9
	SEMLAFLOW	93.3	38	36	5.96	-0.72	**87.2**	7.21	0.85	99.6
	SEMLAFLOW SYNSPACE	72.0	27	48	3.27	-0.80	60.3	2.95	0.80	93.0
	EQGAT-diff	85.9	37	24	4.89	-0.73	78.9	6.75	**0.86**	99.5
Atoms & coords	MiDi	74.4	33	31	4.90	-0.74	63.0	6.00	0.85	99.6
	JODO	91.1	38	31	4.72	-0.74	84.1	4.22	0.85	99.4
	FlowMol-CTMC	89.5	24	25	5.91	-0.68	69.3	13.0	**0.86**	**99.8**
	FlowMol-Gaussian	48.3	6	8	4.24	-0.71	30.7	21.0	**0.86**	99.7

See Table 1 for results, and Figures 9 and 12 for examples. For chemical reasonableness, SYNCOGEN generates almost entirely valid molecules. Our generation details the reaction and building blocks in a multi-step reaction pathway, and as a result, our molecules are significantly more synthesizable compared to baseline methods. Because AiZynthFinder and Syntheseus solve only 50–70 % of known drug-like molecules, our 50–72 % scores likely underestimate true synthesizability. A rigorous conformer geometry and energy comparison between all methods is provided in Appendix D.4.

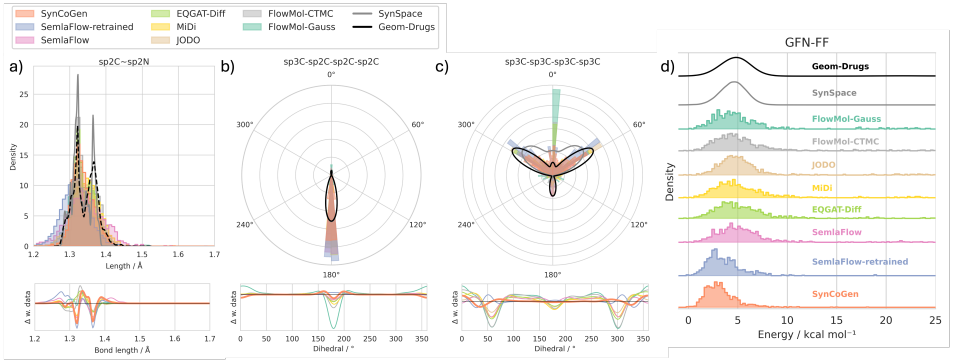


Figure 3: **Conformer geometry and energy distribution comparisons.** Distributions of a) bond lengths, b-c) dihedral angles, d) average per-atom GFN-FF non-bonded interaction energies. Solid curves denote training data densities; lower subpanels in (a-c) show deviations between generated samples and data.

Structurally, the generated conformers reproduce the data energy distributions and have very favorable non-covalent interaction energies as evaluated by semi-empirical quantum-chemistry methods, especially when compared to the baseline methods (Table 1 and Figure 3). This is evident from the lack of structural changes upon further geometric relaxation (Figure 10). The Wasserstein-1 distances and Jensen-Shannon divergence can be found in Appendix D.4 and Figure 8. The low non-bonded energies indicate SYNCOGEN learns to sample many intramolecular interactions (Figure 9). Quantitatively, 87% of these conformers pass PoseBusters pose plausibility checks. Furthermore, SYNCOGEN reproduces the delicate data distribution of bond lengths, angles, and dihedrals (Figures 3 and 8). For example, SYNCOGEN generates fewer sp^2C-sp^2N bonds that are too short, captures sharp bond angle distributions (e.g., sp^3C-sp^3C-sp^3N), and replicates both flexible dihedral angle distribution (e.g. sp^3C-sp^3C-sp^3C-sp^3C) and rigid dihedral angles (e.g. sp^3C-sp^2C-sp^2C-sp^2C).

Our various training-time ablations (Table 2) show that the largest performance gains originate from our chemistry-sensitive graph constraints and self-conditioning, with small contributions from other training/sampling details. A large performance gap between SYNCOGEN and SemlaFlow retrained on SYNSPACE further shows that our training procedure, rather than the architecture or dataset, is the primary driver of performance. Sampling-time ablations are presented in Appendix D.2.

The multi-modal model can perform other tasks; for example, given randomly reaction-graphs, SYNCOGEN can perform zero-shot conformer generation at a quality similar to ETKDG as implemented in RDKit (Table 5). Finally, SYNCOGEN captures the training distribution (low FCD), while generally producing novel molecules. The generated samples have slightly lower diversity as a trade-off for using a (limited) set of reaction building blocks. All generated samples are unique.

5.2 MOLECULAR INPAINTING FOR FRAGMENT LINKING

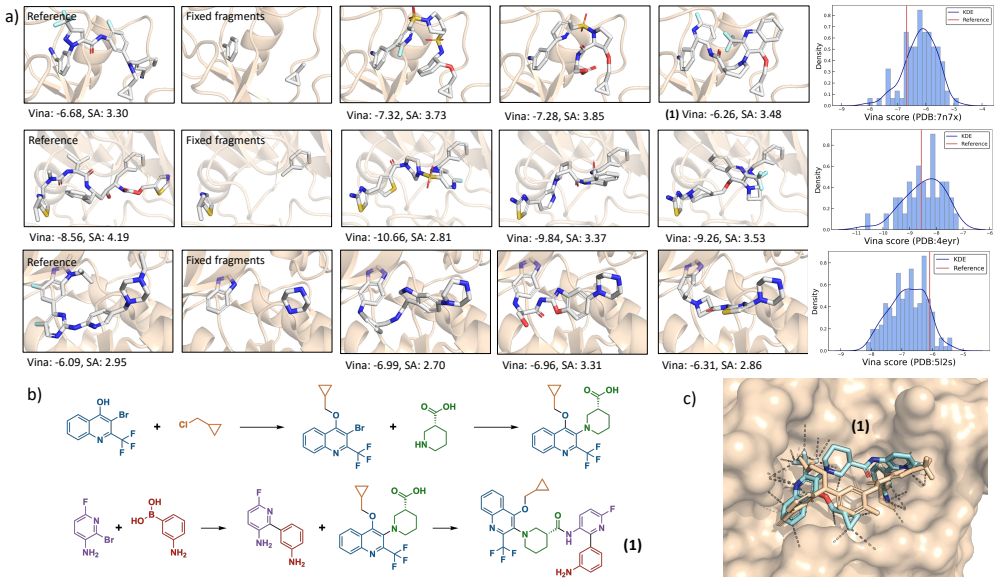


Figure 4: **Molecular inpainting.** a) Fragment linking starts from three experimentally identified ligands in the PDB that contain substructure matches with our building blocks. We show three examples of linkers generated by SYNCOGEN per structure and the distribution of Vina docking scores. b) Proposed synthesis pathway for molecule **(1)** sampled from our model and c) structure of **(1)** (blue) docked onto PDB 7N7X using AlphaFold3 compared against the PDB ligand (beige).

To demonstrate the applicability of SYNCOGEN, we study fragment linking (Bancet et al., 2020) to design *easily synthesizable* analogs of hard-to-make drugs. Fragment linking in drug design enables the construction of potent molecules by connecting smaller fragments that are known to bind distinct regions of a target site. We formulate fragment linking as a molecular inpainting task, where we fix the identity and coordinates of two fragments in a known ligand and sample its missing parts consistent with both geometry and reaction grammar.

As a case study, we pick several FDA-approved, hard-to-synthesize small molecules with experimental crystal structures, each bound to a different target protein. We select human plasma kallikrein (PDB: 7N7X), multidrug-resistant HIV protease 1 (PDB: 4EYR), and human cyclin-dependent kinase 6 (PDB: 5L2S), where each structure is complexed with a ligand that contains at least two of our building blocks. At sampling time, we condition on the substructure match by keeping fixed fragments denoised and interpolating the remaining coordinates (Appendix B.15).

Generated molecules are evaluated with AutoDock Vina (Figure 4) (Eberhardt et al., 2021). SYNCOGEN consistently produces molecules with docking scores on par with or better than the native ligand while satisfying constraints on the presence of specific building blocks. AlphaFold3 (Abramson et al., 2024b) predictions on selected protein-ligand pairs show similar binding positions in the selected pockets as well (Figures 4 and 11). Crucially, unlike existing approaches (Schneuing et al., 2024; Igashov et al., 2024), the model links fragments using building blocks *and* reactions to ensure streamlined synthetic routes of the designs (Table 6 and Figure 12).

Using SYNCOGEN for fragment-linking does not require retraining; however, improved validities and energies can be observed after fine-tuning for motif scaffolding (Table 6. We benchmarked SYNCOGEN against the state-of-the-art, purpose-built fragment-linking model DiffLinker (Igashov et al., 2024). SYNCOGEN is the only method that produces synthesizable molecules with 58-79% retrosynthesis solve rate (0% for DiffLinker, Table 6). Compared to DiffLinker, our molecules have lower interaction energies, no disconnected fragments, and similar PoseBuster validity rate. The synthesizable inpainted molecules now enables wet-lab tasks such as scaffold hopping, synthesizable analog generation, or PROTAC design (Békés et al., 2022; Chirnomas et al., 2023).

5.3 PHARMACOPHORE CONDITIONING

We evaluate SYNCOGEN on the practical, structure-informed generation task of designing *de novo* small-molecule binders conditioned solely on pharmacophore profiles (Sections 3.3 and 4.1). This setting avoids any external reward models (which can encourage reward hacking) and instead asks the generator to directly realize 3D arrangements of *interaction features* that are compatible with a target pocket or reference ligand. We select three disease-relevant targets with hard-to-synthesize reference ligands: ozanimod, scopolamine, and TR-107 (PDB IDs: 7EW0, 8CVD, 7UVU, respectively). For each target we compute a pharmacophore profile from the cognate ligand and generate n=100 molecules with SYNCOGEN. We compare against the state-of-the-art method ShEPhERD (Adams et al., 2025) (also n=100 per target). Valid samples are docked with AutoDock Vina.

Model	Target	Val. ↑	AiZyn. ↑	Synth. ↑	PB ↑
SYNCOGEN	7EW0	89	63	73	66
	7UVU	86	59	75	58
	8CVD	94	59	68	64
ShEPhERD	7EW0	41	10	10	33
	7UVU	43	7	4	36
	8CVD	44	13	12	43

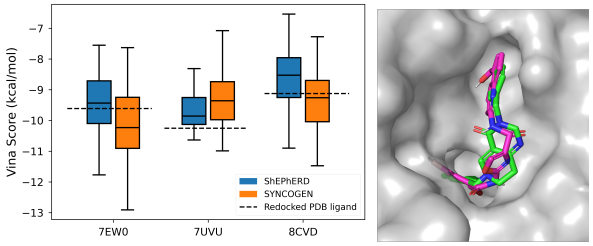


Figure 5: **Pharmacophore-conditioned generation**. Left to right: Pharmacophore conditional generation metrics of SYNCOGEN vs. ShEPhERD. Docking score box-plot comparison, SYNCOGEN vs. ShEPhERD vs. PDB. SYNCOGEN-generated (purple) overlaid with PDB ligand (green, 7UVU).

On average, SYNCOGEN produces *de novo* molecules with better or competitive docking scores compared to both ShEPhERD and the known ligand, with the top samples consistently surpass both across different targets (Section 5.3). Qualitatively, SYNCOGEN-generated molecules dock to the same pocket and replicate key pharmacophoric contacts of the known ligand with a high shape overlap. Compared to the baseline, SYNCOGEN generated molecules have a higher RDKit validity, and most importantly, retrosynthesis solve rates are several times higher (e.g., 63% vs. 10% on 7EW0). PoseBusters validity rate also increases by 20–30%, reflecting better local geometry.

6 CONCLUSION

In this work, we introduced SYNCOGEN, a multimodal generative model that jointly samples building-block reaction graphs and atomic coordinates. Our chemistry-aware training procedures enable this model to learn to design synthesizable molecules directly in Cartesian space. We curated SYNSPACE, a new dataset of 600k readily synthesizable molecules paired with low-energy 3D conformations. SYNCOGEN achieves state-of-the-art performance across 3D molecular generation benchmarks, while natively returning a tractable synthetic route for each structure. The practical utility of SYNCOGEN is demonstrated in both molecular inpainting and pharmacophore-conditioned generation to design easily synthesizable drug analogs using chemical and geometric profiles.

The design space of SYNCOGEN is not limited to SYNSPACE. Our code base includes a data preparation and finetuning pipeline by which interested researchers can easily add their own building blocks and reactions and finetune/retrain our model. Similarly, the separate output heads enables pretraining on other all-atom structural datasets. Subsequent work will explore finetuning in details.

With SYNCOGEN showing *in-silico* amortized synthesizability-aware 3D generation, future works needs to prove rapid experimental synthesis and binding of the *de novo* molecular designs conditioned with 3D information. Beyond drug design, the same co-generative principle may be extended to organic materials discovery, where synthesizability and 3D accuracy are equally pivotal.

Ethics Statement

This work develops a method for synthesizability-aware 3D molecular generation. While intended for research in drug discovery, any generative chemistry system has dual-use risk (e.g., suggesting toxic, hazardous, or otherwise harmful compounds). We mitigate this by (i) constraining generation to a predetermined vocabulary of commercially available building blocks and a limited set of high-yield reaction templates, (ii) representing products as explicit reaction graphs, which enables expert review of routes, and (iii) framing all docking and scoring results as in-silico hypotheses that require independent experimental validation. The dataset is procedurally constructed from public chemistries and contains no human subjects or personally identifiable information. We will release usage guidelines that prohibit targeting dangerous biological pathways or deploying the model for autonomous synthesis without qualified oversight. We recognize the environmental impact of training large models and will report estimated energy use and emissions; we encourage reusing our checkpoints rather than retraining when possible.

Reproducibility Statement

We provide an anonymized code for this study. The repository includes: (1) end-to-end training and sampling scripts for the joint multi-modal model; (2) configuration files with all hyperparameters, noise schedules, and random seeds; (3) an environment file specifying exact versions for supporting libraries; (4) evaluation pipelines that reproduce the metrics; and (5) data preparation code to regenerate the conformer sets and pharmacophore features. We also will release pretrained checkpoints and commands to reproduce: unconditional generation, fragment-linking inpainting, and pharmacophore-conditioned sampling. At camera-ready, we will add a one-command script that reproduces the main results on a single multi-GPU node and logs all artifacts for auditability.

References

Milad Abolhasani and Eugenia Kumacheva. The rise of self-driving labs in chemical and materials sciences. *Nature Synthesis*, 2(6):483–492, 2023.

Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J Ballard, Joshua Bambrick, Sebastian W Bodenstein, David A Evans, Chia-Chun Hung, Michael O'Neill, David Reiman, Kathryn Tunyasuvunakool, Zachary Wu, Akvilė Žemgulytė, Eirini Arvaniti, Charles Beattie, Ottavia Bertolli, Alex Bridgland, Alexey Cherepanov, Miles Congreve, Alexander I Cowen-Rivers, Andrew Cowie, Michael Figurnov, Fabian B Fuchs, Hannah Gladman, Rishub Jain, Yousuf A Khan, Caroline M R Low, Kuba Perlin, Anna Potapenko, Pascal Savy, Sukhdeep Singh, Adrian Stecula, Ashok Thillaisundaram, Catherine Tong, Sergei Yakneen, Ellen D Zhong, Michal Zielinski, Augustin Žídek, Victor Bapst, Pushmeet Kohli, Max Jaderberg, Demis Hassabis, and John M Jumper. Accurate structure prediction of biomolecular interactions with AlphaFold 3. *Nature*, 630(8016):493–500, June 2024a.

Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J Ballard, Joshua Bambrick, et al. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, 630(8016):493–500, 2024b.

Keir Adams, Kento Abeywardane, Jenna Fromer, and Connor W. Coley. Shepherd: Diffusing shape, electrostatics, and pharmacophores for bioisosteric drug design, 2025. URL `https://arxiv.org/abs/2411.04130`.

Michael S Albergo, Nicholas M Boffi, and Eric Vanden-Eijnden. Stochastic interpolants: A unifying framework for flows and diffusions. *arXiv preprint arXiv:2303.08797*, 2023.

Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in Neural Information Processing Systems*, 34:17981–17993, 2021.

Simon Axelrod and Rafael Gomez-Bombarelli. Geom, energy-annotated molecular conformations for property prediction and molecular generation. *Scientific Data*, 9(1):185, 2022.

Frazier N Baker, Ziqi Chen, Daniel Adu-Ampratwum, and Xia Ning. RLSynC: Offline–online reinforcement learning for synthon completion. *Journal of Chemical Information and Modeling*, 64(17):6723–6735, 2024.

Alexandre Bancet, Claire Raingeval, Thierry Lomberget, Marc Le Borgne, Jean-François Guichou, and Isabelle Krimm. Fragment linking strategies for structure-based drug design. *Journal of medicinal chemistry*, 63(20):11420–11435, 2020.

Christoph Bannwarth, Sebastian Ehlert, and Stefan Grimme. GFN2-xTB—An accurate and broadly parametrized Self-Consistent Tight-Binding quantum chemical method with multipole electrostatics and Density-Dependent dispersion contributions. *J. Chem. Theory Comput.*, 15(3):1652–1671, March 2019.

Fan Bao, Shen Nie, Kaiwen Xue, Chongxuan Li, Shi Pu, Yaole Wang, Gang Yue, Yue Cao, Hang Su, and Jun Zhu. One transformer fits all distributions in multi-modal diffusion at scale. In *International Conference on Machine Learning*, pp. 1692–1717. PMLR, 2023.

Miklós Békés, David R Langley, and Craig M Crews. PROTAC targeted protein degraders: the past is prologue. *Nat. Rev. Drug Discov.*, 21(3):181–200, March 2022.

Joey Bose, Tara Akhound-Sadegh, Guillaume Huguet, Kilian FATRAS, Jarrid Rector-Brooks, Cheng-Hao Liu, Andrei Cristian Nica, Maksym Korablyov, Michael M Bronstein, and Alexander Tong. Se(3)-stochastic flow matching for protein backbone generation. In *The Twelfth International Conference on Learning Representations*, 2024.

John Bradshaw, Brooks Paige, Matt J Kusner, Marwin Segler, and José Miguel Hernández-Lobato. A model to search for synthesizable molecules. *Advances in Neural Information Processing Systems*, 32, 2019.

John Bradshaw, Brooks Paige, Matt J Kusner, Marwin Segler, and José Miguel Hernández-Lobato. Barking up the right tree: an approach to search over molecule synthesis DAGs. *Advances in Neural Information Processing systems*, 33:6852–6866, 2020.

Martin Buttenschoen, Garrett M. Morris, and Charlotte M. Deane. Posebusters: Ai-based docking methods fail to generate physically valid poses or generalise to novel sequences. *Chemical Science*, 15(9):3130–3139, 2024.

Andrew Campbell, Jason Yim, Regina Barzilay, Tom Rainforth, and Tommi Jaakkola. Generative flows on discrete state-spaces: Enabling multimodal flows with applications to protein co-design. In *International Conference on Machine Learning*, pp. 5453–5512. PMLR, 2024.

Deborah Chirnomas, Keith R Hornberger, and Craig M Crews. Protein degraders enter the clinic - a new approach to cancer therapy. *Nat. Rev. Clin. Oncol.*, 20(4):265–278, April 2023.

Miruna Cretu, Charles Harris, Ilia Igashov, Arne Schneuing, Marwin Segler, Bruno Correia, Julien Roy, Emmanuel Bengio, and Pietro Liò. Synflownet: Design of diverse and novel molecules with synthesis constraints. *arXiv preprint arXiv:2405.01155*, 2024.

Ian Dunn and David Ryan Koes. Mixed continuous and categorical flow matching for 3d de novo molecule generation. *arXiv preprint arXiv:2404.19739*, 2024.

Jerome Eberhardt, Diogo Santos-Martins, Andreas F Tillack, and Stefano Forli. Autodock vina 1.2. 0: New docking methods, expanded force field, and python bindings. *Journal of Chemical Information and Modeling*, 61(8):3891–3898, 2021.

Zhiguang Fan, Yuedong Yang, Mingyuan Xu, and Hongming Chen. EC-Conf: A ultra-fast diffusion model for molecular conformation generation with equivariant consistency. *Journal of Cheminformatics*, 16(1):107, September 2024.

Rémi Flamary, Nicolas Courty, Alexandre Gramfort, Mokhtar Z. Alaya, Aurélie Boisbunon, Stanislas Chambon, Laetitia Chapel, Adrien Corenflos, Kilian Fatras, Nemo Fournier, Léo Gautheron, Nathalie T.H. Gayraud, Hicham Janati, Alain Rakotomamonjy, Ievgen Redko, Antoine Rolet, Antony Schutz, Vivien Seguy, Danica J. Sutherland, Romain Tavenard, Alexander Tong, and Titouan Vayer. Pot: Python optimal transport. *Journal of Machine Learning Research*, 22(78):1–8, 2021.

Piotr Gaiński, Oussama Boussif, Dmytro Shevchuk, Andrei Rekesh, Ali Parviz, Mike Tyers, Robert A Batey, and Michał Koziarski. Scalable and cost-efficient de novo template-based molecular generation. In *ICLR 2025 Workshop on Generative and Experimental Perspectives for Biomolecular Design*, 2025.

Wenhao Gao and Connor W Coley. The synthesizability of molecules proposed by generative models. *Journal of Chemical Information and Modeling*, 60(12):5714–5723, 2020.

Wenhao Gao, Rocío Mercado, and Connor W Coley. Amortized tree generation for bottom-up synthesis planning and synthesizable molecular design. In *The Tenth International Conference on Learning Representations*, 2022.

Wenhao Gao, Shitong Luo, and Connor W Coley. Generative artificial intelligence for navigating synthesizable chemical space. *arXiv preprint arXiv:2410.03494*, 2024.

Samuel Genheden, Amol Thakkar, Veronika Chadimová, Jean-Louis Reymond, Ola Engkvist, and Esben Bjerrum. AiZynthFinder: a fast, robust and flexible open-source software for retrosynthetic planning. *Journal of Cheminformatics*, 12(1):70, November 2020.

Sai Krishna Gottipati, Boris Sattarov, Sufeng Niu, Yashaswi Pathak, Haoran Wei, Shengchao Liu, Simon Blackburn, Karam Thomas, Connor Coley, Jian Tang, et al. Learning to navigate the synthetically accessible chemical space using reinforcement learning. In *International conference on machine learning*, pp. 3668–3679. PMLR, 2020.

Tom George Grigg, Mason Burlage, Oliver Brook Scott, Dominique Sydow, and Liam Wilbraham. Active learning on synthons for molecular design. In *ICLR 2025 Workshop on Generative and Experimental Perspectives for Biomolecular Design*, 2025.

Jeff Guo and Philippe Schwaller. Directly optimizing for synthesizability in generative molecular design using retrosynthesis models. *Chemical Science*, 16(16):6943–6956, 2025.

Majdi Hassan, Nikhil Shenoy, Jungyoon Lee, Hannes Stärk, Stephan Thaler, and Dominique Beaini. Et-flow: Equivariant flow-matching for molecular conformer generation. *Advances in Neural Information Processing Systems*, 37:128798–128824, 2024.

Julien Horwood and Emmanuel Noutahi. Molecular design in synthetically accessible chemical space via deep reinforcement learning. *ACS omega*, 5(51):32984–32994, 2020.

Han Huang, Leilei Sun, Bowen Du, and Weifeng Lv. Learning joint 2d & 3d diffusion models for complete molecule generation. *arXiv preprint arXiv:2305.12347*, 2023.

Ilia Igashov, Hannes Stärk, Clément Vignac, Arne Schneuing, Victor Garcia Satorras, Pascal Frossard, Max Welling, Michael Bronstein, and Bruno Correia. Equivariant 3d-conditional diffusion model for molecular linker design. *Nature Machine Intelligence*, 2024.

Ross Irwin, Alessandro Tibo, Jon Paul Janet, and Simon Olsson. Semlaflow–efficient 3d molecular generation with latent attention and equivariant flow matching. In *The 28th International Conference on Artificial Intelligence and Statistics*, 2025.

Bowen Jing, Gabriele Corso, Jeffrey Chang, Regina Barzilay, and Tommi Jaakkola. Torsional diffusion for molecular conformer generation. *Advances in Neural Information Processing Systems*, 35:24240–24253, 2022.

Zygimantas Jocys, Henriette MG Willems, and Katayoun Farrahi. SynthFormer: Equivariant pharmacophore-based generation of molecules for ligand-based drug design. *arXiv preprint arXiv:2410.02718*, 2024.

Gwanghyun Kim, Alonso Martinez, Yu-Chuan Su, Brendan Jou, José Lezama, Agrim Gupta, Lijun Yu, Lu Jiang, Aren Jansen, Jacob Walker, et al. A versatile diffusion transformer with mixture of noise levels for audiovisual generation. *arXiv preprint arXiv:2405.13762*, 2024.

Michał Koziarski, Andrei Rekesh, Dmytro Shevchuk, Almer van der Sloot, Piotr Gaiński, Yoshua Bengio, Chenghao Liu, Mike Tyers, and Robert Batey. RGFN: Synthesizable molecular generation using GFlowNets. *Advances in Neural Information Processing Systems*, 37:46908–46955, 2024.

Tuan Le, Julian Cremer, Frank Noe, Djork-Arné Clevert, and Kristof T Schütt. Navigating the design space of equivariant diffusion-based generative models for de novo 3d molecule generation. In *The Twelfth International Conference on Learning Representations*, 2024.

Chaejeong Lee, Jayoung Kim, and Noseong Park. Codi: Co-evolving contrastive diffusion models for mixed-type tabular synthesis. In *International Conference on Machine Learning*, pp. 18940–18956. PMLR, 2023.

Daeseok Lee and Yongjun Cho. Fine-tuning pocket-conditioned 3D molecule generation via reinforcement learning. In *ICLR 2024 Workshop on Generative and Experimental Perspectives for Biomolecular Design*, 2024.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2023.

Cheng-Hao Liu, Maksym Korablyov, Stanisław Jastrzebski, Paweł Włodarczyk-Pruszynski, Yoshua Bengio, and Marwin Segler. Retrognn: fast estimation of synthesizability for virtual screening and de novo design by learning from slow retrosynthesis software. *Journal of Chemical Information and Modeling*, 62(10):2293–2300, 2022.

Xingchao Liu, Chengyue Gong, et al. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *The Eleventh International Conference on Learning Representations*, 2023.

Jiasen Lu, Christopher Clark, Sangho Lee, Zichen Zhang, Savya Khosla, Ryan Marten, Derek Hoiem, and Aniruddha Kembhavi. Unified-io 2: Scaling autoregressive multimodal models with vision language audio and action. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26439–26455, 2024.

Krzysztof Maziarz, Austin Tripp, Guoqing Liu, Megan Stanley, Shufang Xie, Piotr Gaiński, Philipp Seidl, and Marwin H. S. Segler. Re-evaluating retrosynthesis algorithms with syntheseus. *Faraday Discussions*, 256:568–586, 2025.

Brian Medel-Lacruz, Albert Herrero, Fernando Martín, Enric Herrero, F Javier Luque, and Javier Vázquez. Synthon-based strategies exploiting molecular similarity and protein–ligand interactions for efficient screening of ultra-large chemical libraries. *Journal of Chemical Information and Modeling*, 2025.

Chameleon Meta. Chameleon: Mixed-modal early-fusion foundation models. *arXiv preprint arXiv:2405.09818*, 2024.

Alex Morehead and Jianlin Cheng. Geometry-complete diffusion for 3D molecule generation and optimization. *Communications Chemistry*, 7(1):150, 2024.

Stefano Peluchetti. Diffusion bridge mixture transports, schrödinger bridge problems and generative modeling. *Journal of Machine Learning Research*, 24(374):1–51, 2023.

Philipp Pracht, Stefan Grimme, Christoph Bannwarth, Fabian Bohle, Sebastian Ehlert, Gereon Feldmann, Johannes Gorges, Marcel Müller, Tim Neudecker, Christoph Plett, Sebastian Spicher, Pit Steinbach, Patryk A. Wesołowski, and Felix Zeller. Crest—a program for the exploration of low-energy molecular chemical space. *The Journal of Chemical Physics*, 160(11):114110, 2024.

Kristina Preuer, Philipp Renz, Thomas Unterthiner, Sepp Hochreiter, and Günter Klambauer. Fréchet ChemNet distance: A metric for generative models for molecules in drug discovery. *Journal of Chemical Information and Modeling*, 58(9):1736–1741, 2018.

Sereina Riniker and Gregory A Landrum. Better informed distance geometry: Using what we know to improve conformation generation. *Journal of Chemical Information and Modeling*, 55(12): 2562–2574, December 2015.

Subham Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin Chiu, Alexander Rush, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. *Advances in Neural Information Processing Systems*, 37:130136–130184, 2024.

Arne Schneuing, Charles Harris, Yuanqi Du, Kieran Didi, Arian Jamasb, Ilia Igashov, Weitao Du, Carla Gomes, Tom L Blundell, Pietro Lio, et al. Structure-based drug design with equivariant diffusion models. *Nature Computational Science*, 4(12):899–909, 2024.

Seonghwan Seo, Minsu Kim, Tony Shen, Martin Ester, Jinkyoo Park, Sungsoo Ahn, and Woo Youn Kim. Generative flows on synthetic pathway for drug design. *arXiv preprint arXiv:2410.04542*, 2024.

Tony Shen, Seonghwan Seo, Ross Irwin, Kieran Didi, Simon Olsson, Woo Youn Kim, and Martin Ester. Compositional flows for 3D molecule and synthesis pathway co-design. *arXiv preprint arXiv:2504.08051*, 2025.

Jiaxin Shi, Kehang Han, Zhe Wang, Arnaud Doucet, and Michalis Titsias. Simplified and generalized masked diffusion for discrete data. *Advances in Neural Information Processing Systems*, 37: 103131–103167, 2024.

Sebastian Spicher and Stefan Grimme. Robust atomistic modeling of materials, organometallic, and biochemical systems. *Angewandte Chemie International Edition*, 59(36):15665–15673, 2020.

Michael Sun, Alston Lo, Minghao Guo, Jie Chen, Connor W Coley, and Wojciech Matusik. Procedural synthesis of synthesizable molecules. In *The Thirteenth International Conference on Learning Representations*, 2025.

Kyle Swanson, Gary Liu, Denise B Catacutan, Autumn Arnold, James Zou, and Jonathan M Stokes. Generative AI for designing and validating easily synthesizable and structurally novel antibiotics. *Nature Machine Intelligence*, 6(3):338–353, 2024.

Alexander Tong, Kilian FATRAS, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Guy Wolf, and Yoshua Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport. *Transactions on Machine Learning Research*, 2023.

Jos Torge, Charles Harris, Simon V Mathis, and Pietro Lio. DiffHopp: A graph diffusion model for novel drug design via scaffold hopping. *arXiv preprint arXiv:2308.07416*, 2023.

Clement Vignac, Nagham Osman, Laura Toni, and Pascal Frossard. Midi: Mixed graph and 3d denoising diffusion for molecule generation. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 560–576. Springer, 2023.

Chentong Wang, Sarah Alamdari, Carles Domingo-Enrich, Ava P. Amini, and Kevin K. Yang. Toward deep learning sequence–structure co-generation for protein design. *Current Opinion in Structural Biology*, 91:103018, 2025.

Jinheng Xie, Weijia Mao, Zechen Bai, David Junhao Zhang, Weihao Wang, Kevin Qinghong Lin, Yuchao Gu, Zhijie Chen, Zhenheng Yang, and Mike Zheng Shou. Show-o: One single transformer to unify multimodal understanding and generation. *arXiv preprint arXiv:2408.12528*, 2024.

Sen Yang, Lingli Ju, Cheng Peng, JiangLin Zhou, Yamin Cai, and Dawei Feng. Co-design protein sequence and structure in discrete space via generative flow. *Bioinformatics*, pp. btaf248, 04 2025.

Kiwoong Yoo, Owen Oertell, Junhyun Lee, Sanghoon Lee, and Jaewoo Kang. TurboHopp: Accelerated molecule scaffold hopping with consistency models. *Advances in Neural Information Processing Systems*, 37:41157–41185, 2024.

Hengrui Zhang, Jiani Zhang, Zhengyuan Shen, Balasubramaniam Srinivasan, Xiao Qin, Christos Faloutsos, Huzefa Rangwala, and George Karypis. Mixed-type tabular data synthesis with score-based diffusion in latent space. In *The Twelfth International Conference on Learning Representations*, 2024.

A Chemistry Details

A.1 Building Blocks and Reactions

The 93 selected commercial building blocks and their respective reaction centers are shown in Figure 6. For chemical reactions, we focused on cross-coupling reactions to link fragments together. We chose 8 classes of robust reactions, which can be subdivided into 19 types of reaction templates, see Figure 7. We note that our reaction modeling is simplified. For example, boronic acids in building blocks ($B(OH)_2$) are replaced with boranes (BH_2); we do not consider the need for chemical protection on certain functional groups (e.g. N-Boc); we do not consider directing group effects or stoichiometry when multiple reaction centers are available; we do not consider macrocycles. These edge cases are limitations of the current method, but they are comparably minimal through the careful curation of building blocks to avoid such infeasible chemical reactions.

A.2 Graph Generation

Helper definitions. We annotate each building block with its reaction center atom indices $\mathcal{V}(b) \subseteq V(b)$ and its and each intrinsic atom-level graph by $H(b) := \big(V(b),\, L(b)\big)$, where $V(b)$ is the set of atoms in b and $L(b) \subseteq V(b) \times V(b)$ is the set of covalent bonds internal to the block. Each reaction template r is annotated with a Boolean tuple $\big((l_A(r),\, l_B(r)\,\big) \in \{0,1\}^2$ describing whether reagent A or reagent B in r, respectively, contains a leaving atom.

Given the current atom graph $G_a = (V_a, L_a)$ and an atom $v \in V_a$ of degree 1, the routine UniqueNeighbor(v) returns the *single* atom $u \in V_a$ such that $(u, v) \in L_a$. Throughout the vocabulary, every leaving-group center has exactly one neighbour.

A reaction template r is considered compatible with (b_i, v) and $(\tilde{b}, \tilde{v})$ if it queries for first and second reagent substructures that match (b_i, v) and $(\tilde{b}, \tilde{v})$, respectively.

Algorithm 1 Fragment-by-fragment assembly with Couple

Inputs: vocab $\mathcal{B}$, reactions $\mathcal{R}$, depth limit T
Output: atom graph G_a, building block graph $G_f = (X, E)$

function Couple($G_a,\ b_i,\ \tilde{b},\ r,\ (v_i, \tilde{v})$)
  append all atoms and bonds of $H(\tilde{b})$ to G_a ▷ **1. Handle leaving groups**
  if $l_A(r) = 1$ **then** ▷ v_i leaves in reagent A
    $u_i \leftarrow$ UniqueNeighbor(v_i)
    delete atom v_i (and its bond) from G_a
    $v_i \leftarrow u_i$ ▷ reroute to neighbour
  end if
  if $l_B(r) = 1$ **then** ▷ $\tilde{v}$ leaves in reagent B
    $u_{\mathrm{t}} \leftarrow$ UniqueNeighbor($\tilde{v}$)
    delete atom $\tilde{v}$ (and its bond) from G_a
    $\tilde{v} \leftarrow u_{\mathrm{t}}$ ▷ reroute to neighbour
  end if
  add covalent bond between v_i and $\tilde{v}$ ▷ **2. Add the cross-bond**
  return G_a
end function
$b_0 \leftarrow$ UniformPick($\mathcal{B}$); $G_a \leftarrow H(b_0)$; $G_f \leftarrow (b_0)$
for $t = 1$ **to** T **do**
  $L \leftarrow$ enumerate compatible 5-tuples $\langle b_i, v, r, \tilde{b}, \tilde{v}\rangle$
  if $L = \varnothing$ **then break**
  end if
  $(b_i, v, r, \tilde{b}, \tilde{v}) \leftarrow$ UniformPick(L)
  $e \leftarrow (r, v, \tilde{v})$
  $G_a \leftarrow$ Couple($G_a,\ b_i,\ \tilde{b},\ r, (v, \tilde{v})$)
  $G_f \leftarrow G_f \cup \big(b_i \xrightarrow{e} \tilde{b}\big)$
end for
return (G_a, G_f)

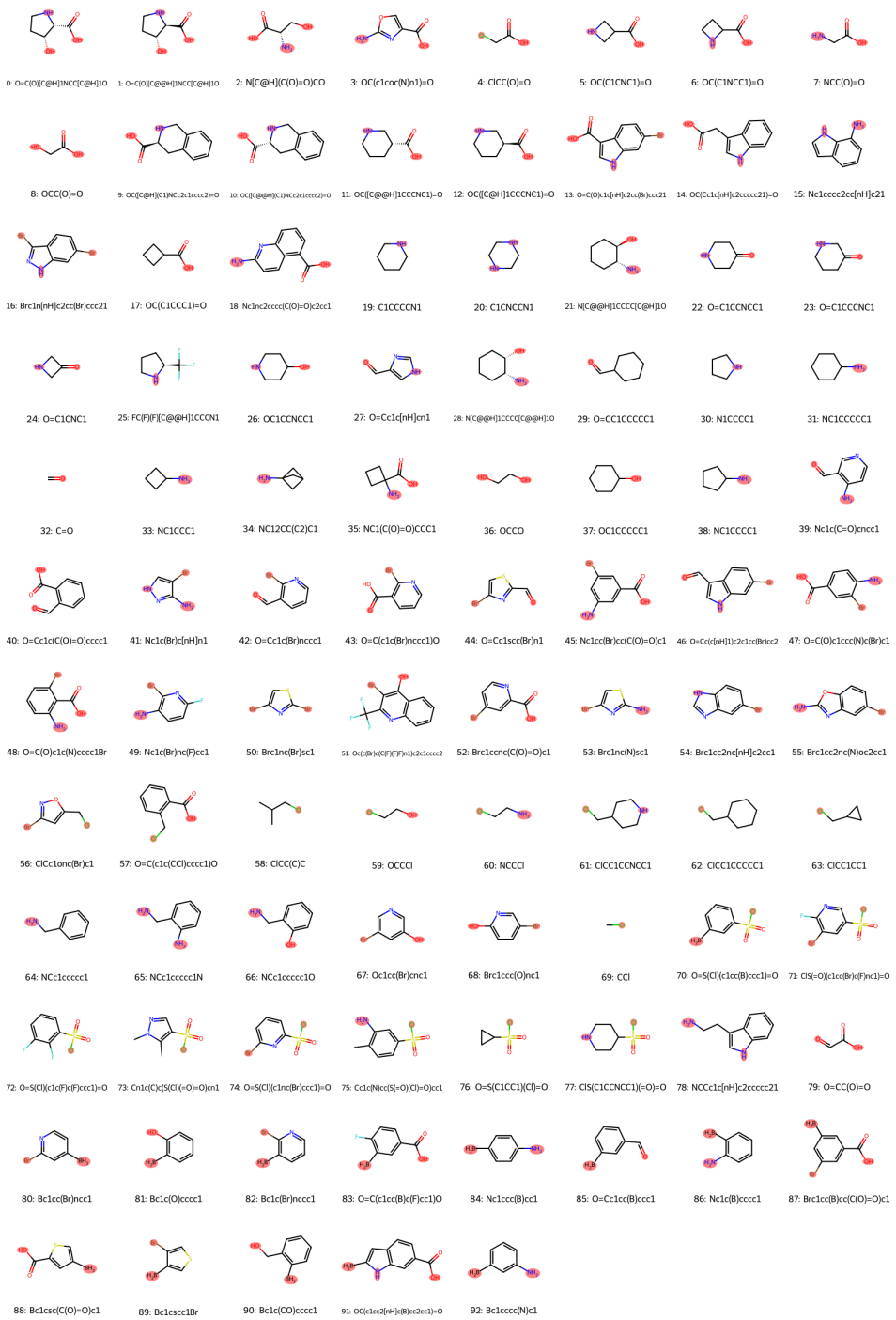


Figure 6: List of building blocks, their respective reaction centers (in red), and their canonical SMILES representation.

B Method Details

B.1 Building Block-Level Representations

Let $X \in \{0,1\}^{N\times|\mathcal{B}|+1}$ be a one-hot matrix where the i^{th} row encodes the identity of the i^{th} building block, and let $E \in \{0,1\}^{N\times N\times|\mathcal{R}|V_{max}^2+2}$, where $V_{max} = \max_b |\mathcal{V}(b)|$. A non-zero entry

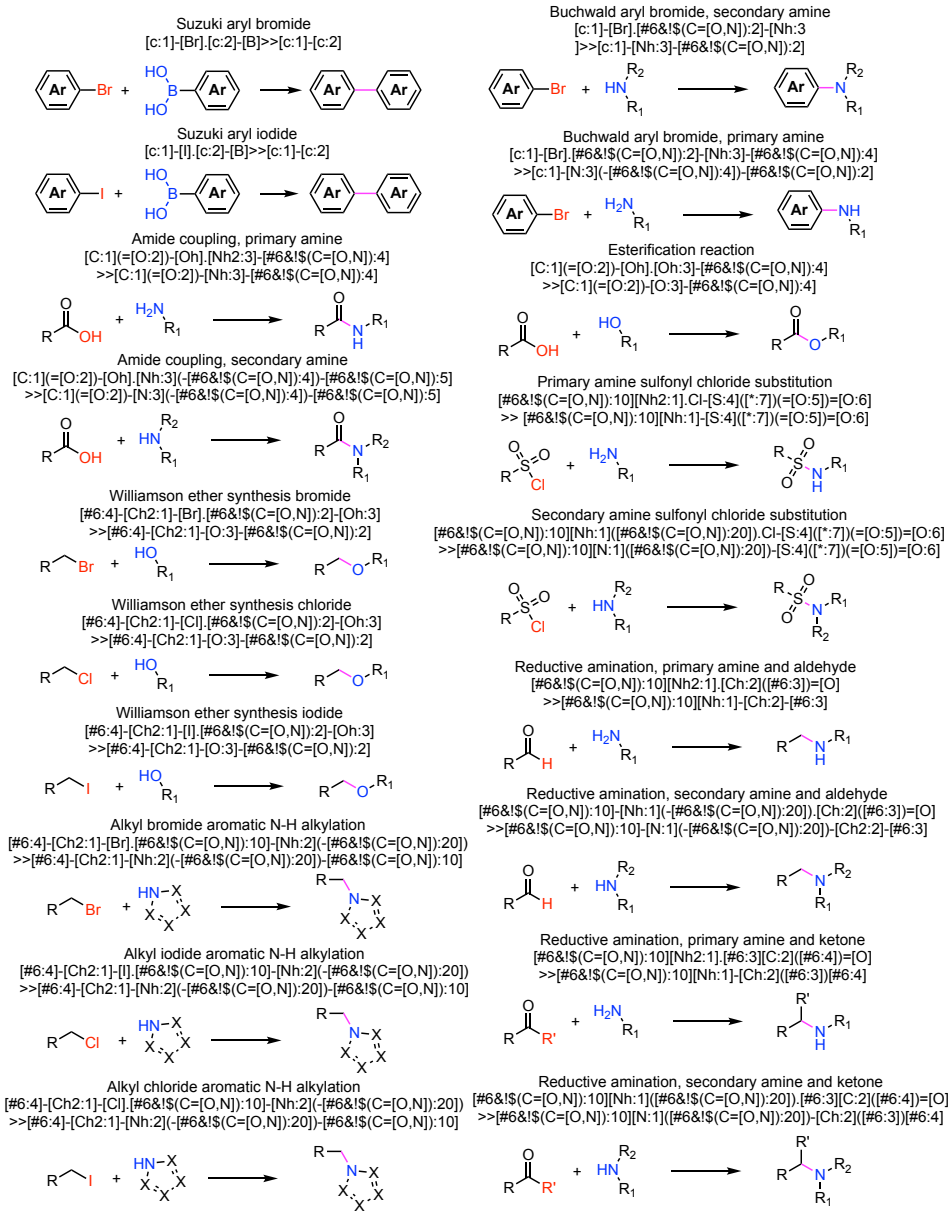


Figure 7: List of chemical reactions used to connect building blocks and their SMARTS representation. Newly formed bonds are highlighted in pink.

$E_{ijr(v_i,v_j)} = 1$ signals that block i (center v_i) couples to block j (center v_j) via reaction r. Graphs (X, E) belonging to molecules containing $n < N$ building blocks are padded to N.

Reserved Channels. We reserve a dedicated *masked* (absorbing) token in both vocabularies:

$$\pi_X \in \{0,1\}^{|\mathcal{B}|}, \qquad \pi_E \in \{0,1\}^{|\mathcal{R}|\,V_{\max}^2}, \tag{5}$$

where π_X (resp. π_E) is the one-hot vector whose single 1-entry corresponds to the masked node (resp. edge) channel. Besides the masked channel, we keep a dedicated *no-edge* channel, encoded by the one-hot vector

$$\lambda_E \in \{0,1\}^{|\mathcal{R}|\,V_{\max}^2}, \tag{6}$$

so every edge slot may take one of three mutually exclusive states: a concrete coupling label, the no-edge token λ_E, or the masked token π_E.

B.2 Atom-Level Representations

The SemlaFlow (Irwin et al., 2025) architecture propagates and updates invariant and equivariant features at the atom level. To ensure consistency with this framework, we calculate for each input graph (X_t, E_t) atom-level one-hot atom and bond features. Crucially, these features must be flexible to arbitrary masking present in X_t and E_t. With this in mind we set each atom feature $X_t^{atom}[i, a]$ to a concatenation of one-hot encodings

$$X_t^{atom}[i,a] \;=\; \Big(\underbrace{\delta_{\mathrm{sym}(i,a)}}_{\text{9-way one-hot}},\; \mathbb{1}[\mathrm{ring}(i,a)],\; \mathbb{1}[a \in \mathcal{V}(X_i)] \Big) \in \{0,1\}^{9+2}, \tag{7}$$

where $\delta_{\mathrm{sym}}(i, a)$ is the one-hot vector over possible atom types (C, N, O, B, F, Cl, Br, S, `[MASK]`) and $\mathrm{ring}(i, a)$ denotes whether or not the atom is a member of a ring. Similarly, we calculate a bond feature matrix

$$E_t^{\text{atom}}[a_i, a_j] = \begin{cases} \delta_{\mathrm{order}}(a_i, a_j), & \text{bond is present,} \\ \mathbf{0}_5, & \text{otherwise.} \end{cases} \tag{8}$$

where $\delta_{\mathrm{order}}(a_i, a_j)$ is the one-hot tensor over possible bond orders (single, double, triple, aromatic, `[MASK]`) between a_i and a_j. E_t^{atom} is populated by loading the known bonds and respective bond orders within denoised building blocks. If some building block X_i is noised, all edges between its constituent atoms $E_t^{atom}[i : i + M, i : i + M]$ are set to the masked one-hot index. For graphs (X_t, E_t) corresponding to valid molecules in which all nodes and edges are denoised, we simply obtain the full bond feature matrix from the molecule described by (X_t, E_t).

B.3 Data Pairing

Algorithm 2 PairData$\big(C_0, S_0, C_1, t, X_t\big)$

Input: C_0 (clean coordinates), S_0 (atom mask), C_1 (prior sample), $t \in [0, 1]$, X_t (partially masked nodes)
Output: $\tilde{C}_0$ (re-centered ground truth), C_t (interpolated noisy coords)

$\mathcal{D}_t \leftarrow \{\, i \mid X_t[i] \neq \pi_X \,\}$ ▷ denoised blocks
$S_t[i, a] \leftarrow \mathbf{1}[i \notin \mathcal{D}_t \vee a \in \mathcal{A}_i]$ ▷ visibility
$\tilde{C}_1 \leftarrow C_1 - \bar{C}_{1 S_t}$
$\tilde{C}_0 \leftarrow$ ZeroTensor()
for all (i, a) **do**
  if $S_0[i, a] = 1$ **then**
    $\tilde{C}_0[i, a] \leftarrow C_0[i, a] - \bar{C}_{1 S_t}$
  else if $S_t[i, a] = 1$ **then** ▷ dummy atom
    $\tilde{C}_0[i, a] \leftarrow \tilde{C}_1[i, a]$
  end if
end for
$C_t \leftarrow (1 - t)\, \tilde{C}_0 + t\, \tilde{C}_1$
return $\big(\tilde{C}_0,\, C_t\big)$

Here, $\mathcal{A}_i$ is the set of all atom indices a that constitute true atoms in X_0. Note that $S_t = S_0$ for all t where X_t contains no masked building blocks.

Note: Non-Equivariance. Our data pairings result in both C_0 and C_t that are properly centered according to atoms that are possibly valid at time t. It is important to note that under this scheme, while the model is $SE(3)$-equivariant with respect to the system defined by the partial mask S_t, it is not equivariant with respect to the orientation of the molecule itself unless $\mathcal{D}_t^c = \varnothing$, as the presence and temporary validity of masked dummy atoms offsets the true atom centering and thus breaks both translational and rotational equivariance.

B.4 TRAINING ALGORITHM

Algorithm 3 Training step for SYNCOGEN

$t \sim \mathcal{U}(0,1)$
$(X_t, E_t) \leftarrow q_t(X_0, E_0)$
$C_1 \sim \mathcal{N}(0, I)$
$(\tilde{C}_0, \tilde{C}_t) \leftarrow \textsc{Pair}(C_0, S_0, C_1, t, X_t)$ ▷ center and interpolate coordinates (Algorithm 2)
$(L_t^X, L_t^E, \hat{\tilde{C}}_0^t) \leftarrow f_\theta(X_t, E_t, \tilde{C}_t, n, t)$
$\mathcal{L} \leftarrow \mathcal{L}_{\text{graph}} + \mathcal{L}_{\text{MSE}} + \mathcal{L}_{\text{pair}}$ ▷ total loss (Appendix B.13)
$\theta \leftarrow \theta - \eta - -bla_\theta \mathcal{L}$

B.5 SAMPLING ALGORITHM

Algorithm 4 Sampling procedure for SYNCOGEN

$n \sim \text{Cat}(\pi_{\text{frag}})$; $(X_1, E_1) \leftarrow (\pi_X, \pi_E)$; $S_1[i,a] \leftarrow \mathbf{1}[i < n]$ ▷ draw n, initialize masks
$C_1 \sim \mathcal{N}(0, I)$; $\tilde{C}_1 \leftarrow C_1 - \bar{C}_{1,S_1}$ ▷ center Gaussian prior by initial mask
for $t = 1$ **down to** 0 **do**
  $\tilde{C}_t \leftarrow C_t - \bar{C}_{t,S_t}$;
  $(L_t^X, L_t^E, \hat{\tilde{C}}_0^t) \leftarrow f_\theta(X_t, E_t, \tilde{C}_t, n, t)$
  $\tilde{L}_t^E \leftarrow \textsc{SampleEdges}(L_t^E, n)$ ▷ enforce one parent per building block (Algorithm 5)
  $X_{t-\Delta t} \leftarrow \textsc{CatSample}(L_t^X)$; $E_{t-\Delta t} \leftarrow \textsc{CatSample}(\tilde{L}_t^E)$ ▷ take reverse step (Appendix B.8)
  $C_{t-\Delta t} \leftarrow C_t + \Delta t(\hat{\tilde{C}}_0^t - \tilde{C}_t)$
  $(X_t, E_t, C_t, S_t) \leftarrow (X_{t-\Delta t}, E_{t-\Delta t}, C_{t-\Delta t}, S_{t-\Delta t})$
end for
$(L^X, L^E, \hat{\tilde{C}}_0) \leftarrow f_\theta(X_0, E_0, \tilde{C}_0, n, 0)$ ▷ final deterministic denoise ($t = 0$)
$\hat{X}_0 \leftarrow \arg\max_k L_\theta^X[\cdots, k]$; $\hat{E}_0 \leftarrow \arg\max_k L_\theta^E[\cdots, k]$; $\hat{C}_0 \leftarrow \hat{\tilde{C}}_0 - \bar{\hat{\tilde{C}}}_{0,S_0}$
return $(\hat{X}_0, \hat{E}_0, \hat{C}_0)$

B.6 INFERENCE-TIME EDGE CONSTRAINTS

Let $E_\theta^t \in [0,1]^{n \times n \times |\mathcal{R}| V_{\max}^2}$ be the soft-max edge probabilities produced at step t. The routine below resolves the unique parent for every building block column $j > 0$ and returns a probability tensor $\tilde{E}_\theta^t$ with exactly one non–zero entry per column.

Algorithm 5 $\textsc{SampleEdges}\big(E_\theta^t, n\big)$

Input: edge probabilities E_θ^t
Output: pruned probabilities $\tilde{E}_\theta^t$

$\tilde{E}_\theta^t \leftarrow \mathbf{0}$
for $j = 1$ **to** $n - 1$ **do**
  $(i_j, e_j) \sim \text{Cat}\big(\{E_\theta^t[i, j, e] \mid 0 \le i < j\}\big)$
  $\tilde{E}_\theta^t[i_j, j, e_j] \leftarrow 1$
end for
return $\tilde{E}_\theta^t$

$\tilde{E}_\theta^t$ is then symmetrized and fed to the discrete reverse sampler described in Appendix B.8.

B.7 BUILDING BLOCK LOGIT PREDICTIONS

The SEMLAFLOW(Irwin et al., 2025) backbone outputs atom–atom edge features $E_\theta^{\text{atom}} \in \mathbb{R}^{B \times (NM) \times (NM) \times d_{\text{edge}}}$. To obtain building block-level tensors, we apply two parallel 2-D con-

volutions (one for nodes, one for edges) with stride M, followed by MLP classifiers that map the pooled features back to their original one-hot vocabularies.

Stride-pooled convolution. Let d_{edge} be the latent edge feature dimension. Each stream uses the block

$$\mathrm{Conv2d}(d_{\text{edge}} \to d_{\text{edge}},\, k = M,\, s = M) \xrightarrow{\text{SiLU}} \mathrm{Conv2d}(d_{\text{edge}} \to d_{\text{edge}},\, k = 1,\, s = 1), \quad (9)$$

so every $M \times M$ atom patch collapses to a single building block entry. This produces

$$X_{\text{pool}} \in \mathbb{R}^{B \times d_{\text{edge}} \times N}, \qquad E_{\text{pool}} \in \mathbb{R}^{B \times d_{\text{edge}} \times N \times N}. \quad (10)$$

Node head. We flatten X_{pool} along its channel axis, concatenate the residual building block one-hot matrix X_t, and pass the result through a two-layer MLP to obtain

$$L_\theta^{X_t} \in \mathbb{R}^{B \times N \times |\mathcal{B}|}. \quad (11)$$

Edge head. We concatenate E_{pool} with the residual building block-edge one-hot tensor E_t, apply an analogous two-layer MLP, and symmetrize to produce

$$L_\theta^{E_t} \in \mathbb{R}^{B \times N \times N \times |\mathcal{R}| V_{\max}^2}. \quad (12)$$

Atom Features. The SEMLAFLOW(Irwin et al., 2025) backbone additionally outputs atom-level node features $X_\theta^{\text{atom}} \in \mathbb{R}^{B \times (NM) \times d_{\text{node}}}$, which are incorporated into E_θ^{atom} via a bond refinement message-passing layer. We find that extracting both building block and edge logits directly from the refined features E_θ^{atom} marginally improves performance relative to separately predicting $L_\theta^{X_t}$ from X_θ^{atom} and $L_\theta^{E_t}$ from E_θ^{atom}.

B.8 DISCRETE NOISING SCHEME

Following (Sahoo et al., 2024), we adopt an absorbing (masked) state noising scheme for X_0 and E_0:

$$q(X_t \mid X_0) = \mathrm{Cat}\big(X_t;\, \alpha_t X_0 + (1 - \alpha_t)\pi_X\big), \qquad q(E_t \mid E_0) = \mathrm{Cat}\big(E_t;\, \alpha_t E_0 + (1 - \alpha_t)\pi_E\big). \quad (13)$$

where $(\alpha_t)_{t \in [0,1]}$ is the monotonically decreasing noise schedule introduced in Section 2.

Reverse categorical posterior. For node identities, we have

$$q(X_s \mid X_t,\, X_0) = \begin{cases} \mathrm{Cat}(X_s; X_t), & X_t \neq \pi_X, \\ \mathrm{Cat}\big(X_s;\, \dfrac{(1 - \alpha_s)\pi_X + \alpha_s\, X_\theta^t}{1 - \alpha_t}\big), & X_t = \pi_X, \end{cases} \quad (14)$$

and, analogously, for edge labels

$$q(E_s \mid E_t,\, E_0) = \begin{cases} \mathrm{Cat}(E_s; E_t), & E_t \neq \ \pi_E \\ \mathrm{Cat}\big(E_s;\, \dfrac{(1 - \alpha_s)\pi_E + \alpha_s\, E_\theta^t}{1 - \alpha_t}\big), & E_t = \pi_E, \end{cases} \quad (15)$$

where $s < t$. Equations (14) and (15) are the direct translation of the reverse denoising process described by (Sahoo et al., 2024) into SYNCOGEN's node–edge representation.

B.9 NOISE SCHEDULE PARAMETERIZATION

Following MDLM (Sahoo et al., 2024), we parameterize the discrete noising schedule via $\alpha_t = e^{-\sigma(t)}$, where $\sigma(t) : [0, 1] \to \mathbb{R}^+$. In all experiments, we adopt the **linear schedule**:

$$\sigma(t) = \sigma_{\max} t, \quad (16)$$

where $\sigma_{\max}$ is a large constant; we use $\sigma_{\max} = 10^8$ as in the original MDLM setup.

Edge Symmetrization. After drawing the upper-triangle entries of the one-hot edge tensor E_s in either the forward or reverse (de)noising process, we enforce symmetry by copying them to the lower triangle:

$$E_{s,jie} \;=\; E_{s,ije}, \qquad 0 \le i < j < n,\;\; e \in \mathcal{R} V_{\max}^2.$$

B.10 POSITIONAL EMBEDDINGS

Though SEMLAFLOW(Irwin et al., 2025) is permutationally invariant by design with respect to atom positions, SYNCOGEN dataset molecules require that atom order be fixed and grouped by building

block for reconstruction purposes. To enforce this during training, we intentionally break permutation invariance by generating and concatenating to each input coordinate sinusoidal positional embeddings representing both global atom index and building block index.

B.11 HYPERPARAMETERS

We train SYNCOGEN for 100 epochs with a batch size of 128 and a global batch size of 512. Note that SEMLAFLOW and Midi are trained for 200 epochs, and EQGAT-diff is trained for up to 800 epochs. All models are trained with a linear noise schedule (see Appendix B.9), with the SUBS parameterization enabled. During training, a random conformer for each molecule is selected, then centered and randomly rotated to serve as the ground-truth coordinates C_0. All atomic coordinates are normalized by a constant Z_c describing the standard deviation across all training examples. For the pairwise distance loss $\mathcal{L}_{pair}$, we set d to 3Å, adjusted for normalization. During training, for each recentered input-prior pair $(\tilde{C}_1, \tilde{C}_0)$ we rotationally align C_1 to C_0. When training with noise scaling and the bond loss time threshold, we set the noise scaling coefficient to 0.2 and the time threshold to 0.25, above which bond length losses are zeroed. When training with auxiliary losses, we set the weights for the pairwise, sLDDT, and bond length loss components to 0.4, 0.4, and 0.2, respectively.

B.12 COMPUTATIONAL RESOURCES USED

We train all models on 2 H100-80GB GPUs.

B.13 TRAINING LOSSES

Here, we define several loss terms that have proved useful for stabilizing training on 3-D geometry. By default, SYNCOGEN is trained with $\mathcal{L}_{\text{MSE}}$ and L_{pair} as coordinate losses.

For a prediction $\big(L_\theta^{X_t}, L_\theta^{E_t}, \hat{\tilde{C}}_0^t\big) = f_\theta\big(X_t, E_t, \tilde{C}_t, n, t\big)$, $X_\theta^t = \text{softmax}\big(L_\theta^{X_t}\big)$, $E_\theta^t = \text{softmax}\big(L_\theta^{E_t}\big)$:

Graph loss. Let X_0 and E_0 be the clean node and edge tensors. Following the MDLM implementation (Sahoo et al., 2024), we weigh the negative log-likelihood at step t by

$$w_t = \frac{\Delta\sigma_t}{\exp(\sigma_t) - 1}, \qquad \Delta\sigma_t = \sigma_t - \sigma_{t-1}, \;\; \sigma_0 = 0, \tag{17}$$

where σ_t is the discrete noise level. The discrete (categorical) loss is then

$$\mathcal{L}_{\text{graph}} = w_t\big(\text{–log}\, X_\theta^t[X_0] \text{ –log}\, E_\theta^t[E_0]\big), \tag{18}$$

i.e. the cross-entropy between the one-hot ground truth and the predicted distributions for both nodes and edges.

MSE loss. Let $S_0 \in \{0,1\}^{N\times M}$ mask the atoms that exist in the clean structure and C_t be the noisy coordinates. Denote $\mathcal{A}_{S_0} = \big\{(i,a) : S_0[i,a] = 1\big\}$.

$$\mathcal{L}_{\text{MSE}} = \frac{1}{|\mathcal{A}_{S_0}|} \sum_{(i,a)\in\mathcal{A}_{S_0}} \big\|\hat{C}_0[i,a] - C_0[i,a]\big\|_2^2, \tag{19}$$

Pairwise loss.

$$\mathcal{L}_{\text{pair}} = \sum_{\substack{(i,a)<(j,b) \\ \|C_0[i,a]-C_0[j,b]\|_2 \le d}} S_0[i,a]\, S_0[j,b]\, \big(\|\hat{C}_0[i,a] - \hat{C}_0[j,b]\|_2 - \|C_0[i,a] - C_0[j,b]\|_2\big)^2, \tag{20}$$

where d is the distance cut-off for pairwise terms. The default total loss value for the model is therefore

$$\mathcal{L}_{\text{SYNCOGEN}} = \mathcal{L}_{\text{graph}} + \mathcal{L}_{\text{MSE}} + \mathcal{L}_{\text{pair}}. \tag{21}$$

Smooth-LDDT loss (Abramson et al., 2024a). Let $d_{ij}^0 := \|C_0[i] - C_0[j]\|_2$ and $d_{ij}^{\text{pred}} := \|\hat{C}_0[i] - \hat{C}_0[j]\|_2$ be ground-truth and predicted inter-atomic distances, respectively. For each pair of atoms

within a 15 Å cutoff in the reference structure, we compute the per-pair score

$$\mathrm{sLDDT}_{ij} = \frac{1}{4}\sum_{k=1}^{4} \sigma\big(\tau_k - |d_{ij}^{\text{pred}} - d_{ij}^{0}|\big), \quad \big[\tau_1, \tau_2, \tau_3, \tau_4\big] = [0.5, 1, 2, 4]\ \text{Å},$$

where $\sigma(x) = 1/(1+e^{-x})$ is the logistic function. The smooth-LDDT loss averages $1 - \mathrm{sLDDT}_{ij}$ over all valid pairs,

$$\mathcal{L}_{\text{sLDDT}} = \frac{\sum_{i<j} \mathbb{1}[d_{ij}^{0} < 15]\, S_0[i]\, S_0[j]\, \big(1 - \mathrm{sLDDT}_{ij}\big)}{\sum_{i<j} \mathbb{1}[d_{ij}^{0} < 15]\, S_0[i]\, S_0[j]}. \tag{22}$$

Bond-length loss. Given a set of intra-fragment bonds $\mathrm{bonds} = \{(p, q)\}$ extracted from the vocabulary, we penalize deviations in predicted bond lengths:

$$\mathcal{L}_{\text{bond}} = \frac{1}{|\mathrm{bonds}|} \sum_{(p,q)\in \mathrm{bonds}} \big|\|\hat{C}_0[p] - \hat{C}_0[q]\|_2 - \|C_0[p] - C_0[q]\|_2\big|. \tag{23}$$

Self-Conditioning. The modified SEMLAFLOW (Irwin et al., 2025) backbone operates on node and edges features at the atomic level, but outputs unnormalized prediction logits $\hat{X}_0 \in \{0,1\}^{N\times|\mathcal{B}|}$ and $\hat{E}_0 \in \{0,1\}^{N\times N\times|\mathcal{R}|V_{max}^2}$. We therefore implement modified self-conditioning for SYNCOGEN that projects previous step graph predictions $\hat{X}_{0cond}$ and $\hat{E}_{0cond}$ to the shape of X_t^{atom} and E_t^{atom} using an MLP.

B.14 CONFORMER GENERATION

We randomly assembled 50 molecules with the reaction graph and used the standard conformational search (iMTD-GC) in CREST with GFN-FF to find all reference conformers. For both SYNCOGEN and RDKit ETKDG, we sampled 50 conformers per molecule and computed the coverage and matching scores. We used a relatively strict RMSD threshold of $\tau = 0.75$.

Formally, COV is defined as:

$$\mathrm{COV} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}\Big[\min_{1\le j\le M} \mathrm{RMSD}(m_i,\, g_j) \le \tau\Big], \tag{24}$$

where $\mathbf{1}[\cdot]$ is the indicator function, m_i are the N generated conformers and g_j are the M reference conformers. And MAT is defined as:

$$\mathrm{MAT} = \frac{1}{N}\sum_{i=1}^{N} \min_{1\le j\le M} \mathrm{RMSD}(m_i,\, g_j). \tag{25}$$

B.15 MOLECULAR INPAINTING

For the inpainting experiments in Section 5.2, we keep two fragments $\mathcal{D} = \{\mathcal{D}^{(1)}, \mathcal{D}^{(2)}\}$ and their coordinates fixed and sample the remaining part of the molecule. We follow Appendix B.5 and initialize the graph prior X_1 with the one-hot encoding of the desired fragment i at a specified node index in the graph (decided at random or based on the structure of the original molecule, so that it matches its scaffold). For each denoised fragment $\mathcal{D}^{(i)}$, we replace its coordinates at each time $t > 0.03$ during sampling by

$$C_t^{(i)} \;=\; (1-t)\,\tilde{C}_0^{(i)} \;+\; t\,\tilde{C}_1^{(i)},$$

where $\tilde{C}_0^{(i)}$ and $\tilde{C}_1^{(i)}$ are the centered ground-truth and prior coordinates of fragment i, respectively, and all other fragments are updated as shown in Appendix B.5. For any $t \le 0.03$, which for 100 sampling steps amounts to the last three steps in the path, we follow normal Euler steps as shown in Appendix B.5 to allow a refinement of the fixed coordinates in line with the rest of the predicted ones for the rest of the fragments. We empirically observed that this led to molecules with lower average energies.

C BASELINE COMPARISONS.

For all baselines, we sampled 1000 molecules with random seeds on an A100 GPU and reported averaged results over three runs.

SemlaFlow We evaluated SemlaFlow using the sampling script in the official codebase on GitHub[1]. We reported results for a model trained on the GEOM (Axelrod & Gomez-Bombarelli, 2022) dataset (by sampling from the checkpoints provided in the repository) and from a model trained on our dataset (see Table 1). We trained SemlaFlow using the default hyperparameters for 150 epochs on a single conformer per molecule.

EQGAT-diff, MiDi, JODO, FlowMol We evaluated EQGAT-diff, Midi, JODO, using their official implementations provided on GitHub[2]. We modified the example sampling script to save molecules as outputted from the reverse sampling, without any post-processing. For MiDi, we evaluated the uniform model. For FlowMol, both CTMC and Gaussian models were evaluated and reported.

D EXTENDED RESULTS AND DISCUSSION

D.1 TRAINING ABLATIONS

Table 2: Training ablations. We incrementally remove inference annealing, auxiliary losses, self-conditioning, scaled-noise, and constraints to see the performance difference. All results shown are at 50 epochs rather than 100 epochs in Table 1. See Sections 4.3 and 4.4, (Appendices B.4, B.11 and B.13).

Method	Valid. ↑	GFN-FF ↓
Base	93.5	4.871
- Inference annealing	93.5	4.933
- Auxiliary losses	85.3	5.194
- Self-conditioning	69.0	6.424
- Scaled noise	70.4	5.091
- Constraints	42.4	67.006

D.2 SAMPLING ABLATIONS

By default, SYNCOGEN implements a linear noise schedule and samples for 100 timesteps. To evaluate the effect of step count and noise schedule choice on performance, we provide experiments with step count decreased to 50 and 20, as well as modified noising to follow a log-linear and geometric schedule. All results listed subsequently can be assumed to use the default noise schedule and step count.

We additionally follow FoldFlow to implement *inference annealing*, a time-dependent scaling on Euler step size that was found to empirically improve in-silico results in protein design Bose et al. (2024). We studied multiplying the Euler step size at time t by $5t$, $10t$, and $50t$. In practice, we employ $10t$ for our experiments unless otherwise noted.

We find that noising and de-noising building blocks according to a linear noise schedule generally achieves good performance, which during inference sees most unmasking occur in the final steps. An aggressive denoising schedule for the discrete fragments yields significantly worse validity (Geometric and Loglinear). Inference annealing that speeds up continuous denoising in the beginning but slows it down near the end helps to inform discrete unmasking and can slightly improve discrete generation validity, energies, and PoseBusters validity. As a sanity check to evaluate whether simultaneous generation is necessary for good performance using SYNCOGEN, we evaluate an inference configurations where all building blocks and reactions are noised until a single final prediction step (FinalOnly) where we find performance using the default parameters to be superior.

[1] `https://github.com/rssrwn/semla-flow/`, available under the MIT License

[2] `https://github.com/jule-c/eqgat_diff/`, `https://github.com/cvignac/MiDi`, `https://github.com/GRAPH-0/JODO`, `https://github.com/Dunni3/FlowMol`, available under the MIT License

Table 3: Sampling ablations. Results are averaged over 1000 generated samples, except retrosynthesis solve rate (out of 100). All results shown are at 50 epochs rather than 100 epochs in Table 1.

	Primary metrics					Secondary metrics		
Method	Valid. ↑	AiZyn. ↑	Synth. ↑	GFN-FF ↓	GFN2-xTB ↓	PB ↑	Div. ↑	Nov. ↑
Linear-100	93.5	55	70	4.933	-0.92	78.3	0.79	94.1
Linear-20	82.4	56	68	5.102	-0.91	71.3	0.78	94.9
Linear-50	92.0	50	65	4.890	-0.91	78.9	0.78	93.6
Geometric-100	48.2	61	68	5.206	-0.84	72.0	0.80	91.7
Loglinear-100	60.3	56	64	5.182	-0.87	70.1	0.80	91.7
Annealing-$5t$	94.7	52	58	5.001	-0.93	79.1	0.78	94.1
Annealing-$10t$ (default)	93.5	42	68	4.870	-0.91	82.8	0.78	94.2
Annealing-$50t$	85.1	51	64	4.972	-0.82	86.7	0.76	94.6
FinalOnly	69.7	39	68	5.260	-0.92	70.1	0.76	94.1

D.3 METRICS

We here describe metric computation details that are absent in the main text.

For synthesizability evaluation, we used the public AiZynthFinder and Syntheseus models. Due to the speed of these models, we only evaluate 100 randomly sampled generated examples. For AiZynthFinder, we used the USPTO policy, the Zinc stock, and we extended the search time to 800 seconds with an iteration limit of 200 seconds. For Syntheseus, we used the LocalRetro model with Retro* search under default settings, with Enamine REAL strict fragments as the stock. We additionally appended our building blocks as the stock, but found no meaningful difference in solved rates, presumably as most of our building blocks are already in the utilized stock. We note that we replaced all boranes with boronic acids due to simplifications made in our modeling (see Appendix A.2).

For energy evaluation, all results are from single-point calculations. For GFN-FF, we report the total energy minus the bond energies (equivalent to the sum of angle, dihedral, bond repulsion, electrostatic, dispersion, hydrogen bond, and halogen bond energies) as the intramolecular non-bond energies, and average it over the number of atoms. For GFN2-xTB, we report the dispersion interaction energies as the intramolecular non-covalent energies. We note that the total energies and bonded energies follow very similar trends. We note that MMFF94 energies are not parameterized for boron; therefore, we report them only for the Wasserstein distances in Appendix D.4 and inpainting task in Table 6. Figures 3 and 8 show distributions obtained from 1,000 molecules generated by each generative method, along with 50,000 subsampled molecules from their respective training datasets. Gaussian kernel density estimation (bandwidth = 0.15) was used for linear distributions, while von Mises kernel density estimation ($\kappa = 25$) was applied for circular distributions. Wasserstein-1 distances (computed linearly for lengths and energies, and on the circle for angles and dihedrals) were calculated using the Python Optimal Transport Package (Flamary et al., 2021).

D.4 *De novo* 3D MOLECULE GENERATION

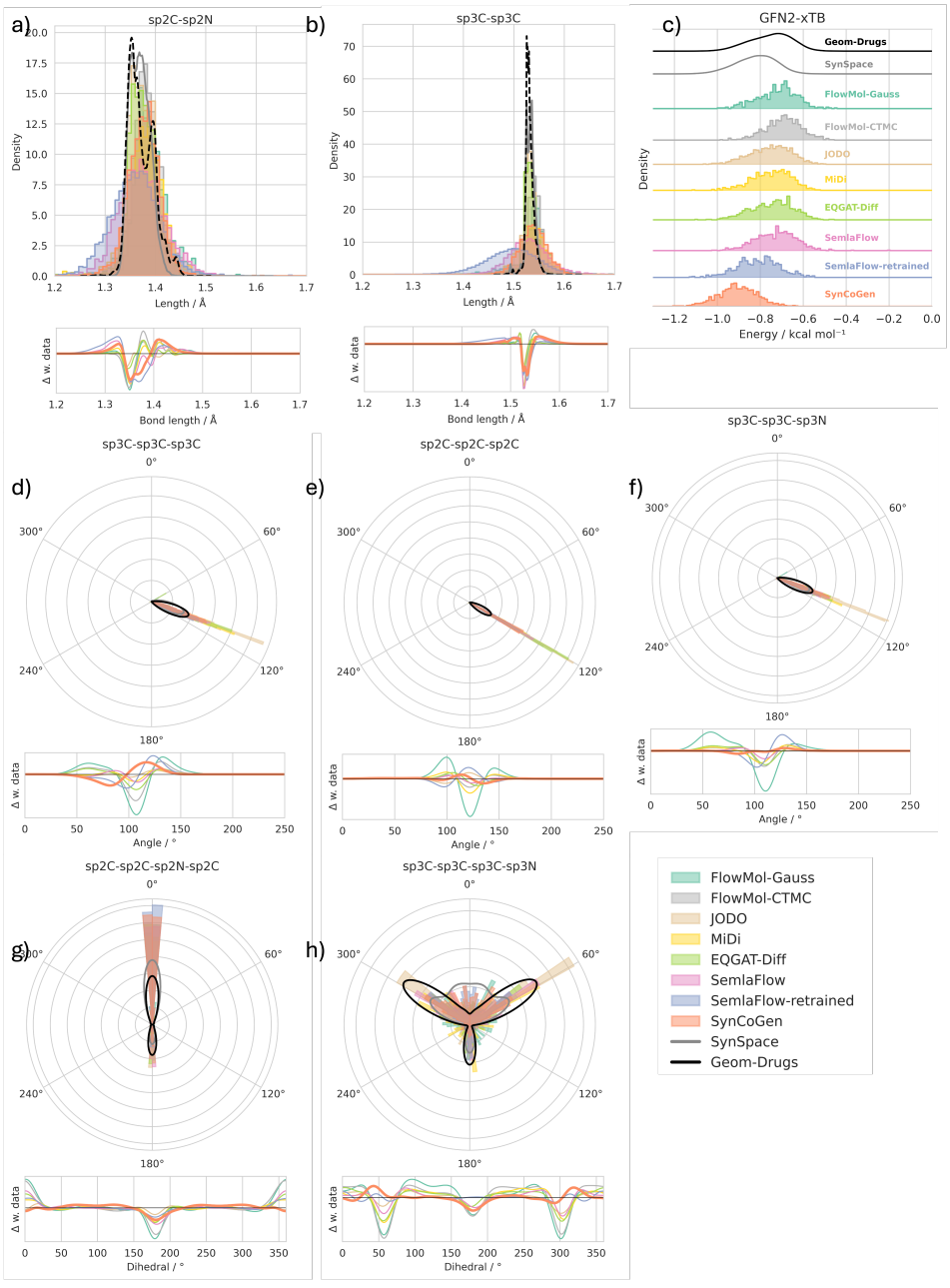


Figure 8: **Additional conformer bond length, angle, dihedral, and energy distribution comparisons.** a-b) Bond lengths, c) GFN-2xTB energy distribution, d-f) bond angles, g-h) dihedral angles. Solid curves denote training data densities; lower subpanels show deviations between generated samples and data.

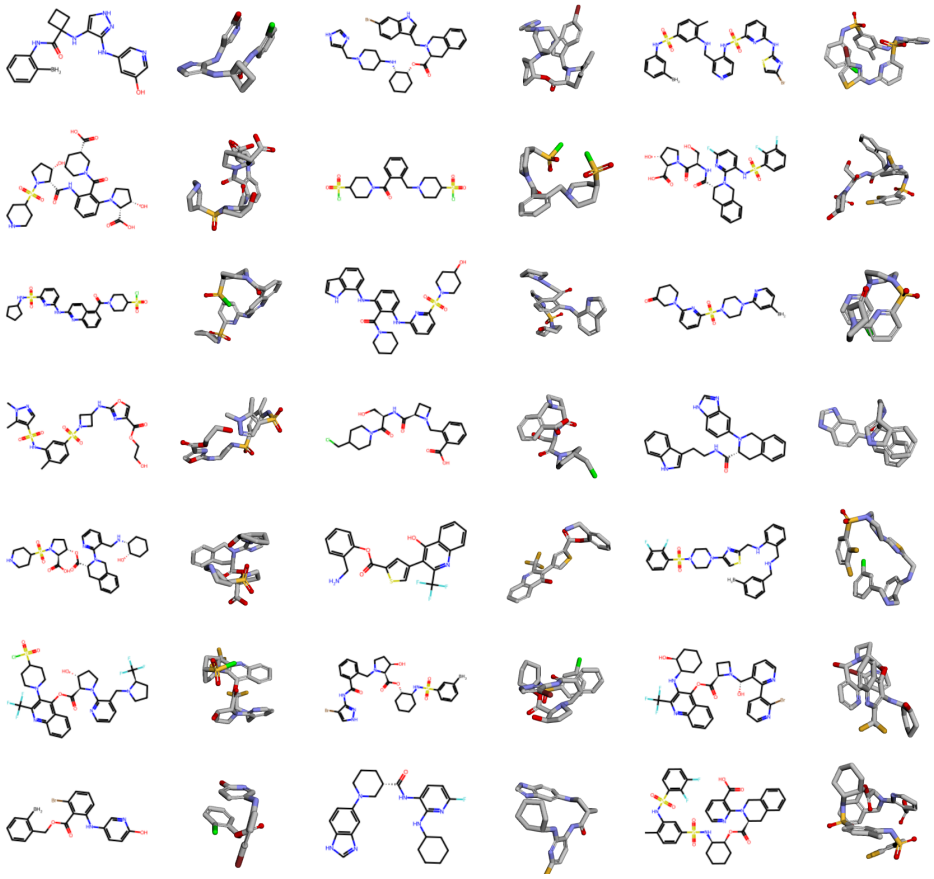


Figure 9: Unconditionally sampled random molecules from SYNCOGEN.

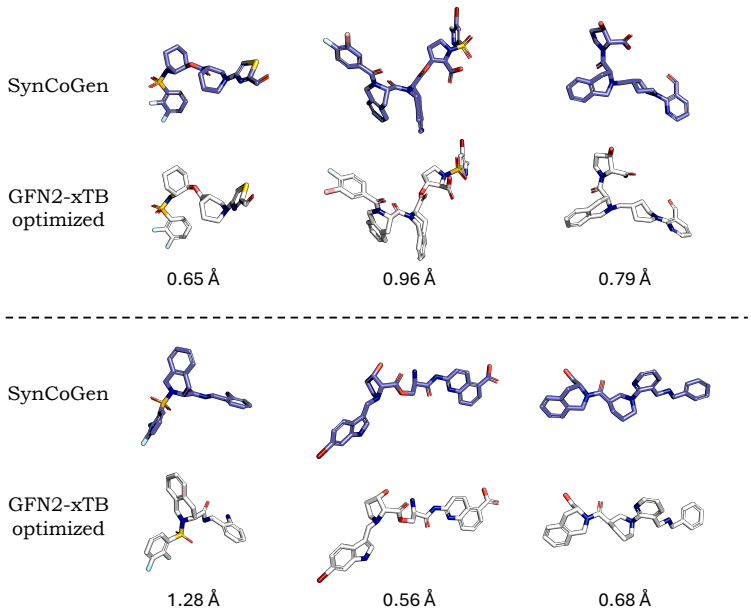


Figure 10: A subset of randomly sampled molecules from SYNCOGEN and further optimized by GFN2-xTB until convergence. Alignment RMSD is shown below the molecular structures.

Table 4: Wasserstein-1 distance (W_1) and Jensen–Shannon divergence (JSD) for the generative models (lower is better). For bond lengths, angles, and dihedrals, we computed the average W_1 and JSD for the top 10 prevalent lengths/angles/dihedrals. Comparisons are made to the respective training set.

(a) Bond dihedrals

Method	W_1	JSD
SYNCOGEN	7.01	0.29
SEMLAFLOW SYNSPACE	6.50	0.22
SEMLAFLOW	7.76	0.28
EQGAT-Diff	8.48	0.29
MiDi	9.32	0.38
JODO	5.47	0.31
FlowMol-CTMC	13.69	0.35
FlowMol-Gauss	18.85	0.46

(b) Bond angles

Method	W_1	JSD
SYNCOGEN	1.36	0.22
SEMLAFLOW SYNSPACE	1.64	0.28
SEMLAFLOW	1.18	0.21
EQGAT-Diff	1.37	0.16
MiDi	1.41	0.21
JODO	0.59	0.12
FlowMol-CTMC	1.90	0.24
FlowMol-Gauss	3.68	0.30

(c) Bond lengths

Method	W_1	JSD
SYNCOGEN	0.0171	0.34
SEMLAFLOW SYNSPACE	0.0320	0.48
SEMLAFLOW	0.0200	0.38
EQGAT-Diff	0.0039	0.13
MiDi	0.0142	0.31
JODO	0.0034	0.12
FlowMol-CTMC	0.0089	0.20
FlowMol-Gauss	0.0152	0.28

(d) `GFN2-xTB` non-covalent E

Method	W_1	JSD
SYNCOGEN	0.0838	0.33
SEMLAFLOW SYNSPACE	0.0125	0.16
SEMLAFLOW	0.0249	0.16
EQGAT-Diff	0.0073	0.12
MiDi	0.0084	0.14
JODO	0.0031	0.11
FlowMol-CTMC	0.0605	0.26
FlowMol-Gauss	0.0322	0.19

(e) `GFN-FF` non-bonded E

Method	W_1	JSD
SYNCOGEN	1.37	0.28
SEMLAFLOW SYNSPACE	1.09	0.22
SEMLAFLOW	1.52	0.16
EQGAT-Diff	1.69	0.18
MiDi	1.80	0.19
JODO	1.33	0.12
FlowMol-CTMC	1.53	0.17
FlowMol-Gauss	2.13	0.17

(f) `MMFF` total E

Method	W_1	JSD
SYNCOGEN	6.59	0.089
SEMLAFLOW SYNSPACE	54.63	0.22
SEMLAFLOW	69.56	0.24
EQGAT-Diff	4.80	0.076
MiDi	19.00	0.11
JODO	22.07	0.11
FlowMol-CTMC	41.95	0.15
FlowMol-Gauss	26.96	0.14

Table 5: With given reaction graphs, comparison of mean coverage (COV) and matching accuracy (MAT) for RDKit ETKDG and zero-shot conformer generation using SYNCOGEN.

Method	**COV (%)** ↑	**MAT (Å)** ↓
RDKit	0.692	0.657
SYNCOGEN	0.614	0.693

D.5 MOLECULAR INPAINTING EXPERIMENTS

Three protein–ligand complexes (PDB IDs 7N7X[3], 5L2S[4] and 4EYR[5]) were selected for molecular inpainting of the ligand structures. These ligands were chosen because they are prominent FDA-approved drugs, and they are typically challenging to synthesize, but the key functional groups are present in our building blocks. Specifically, 4EYR contains ritonavir, a prominent HIV protease inhibitor on the World Health Organization's List of Essential Medicines; 5L2S contains abemaciclib, an anti-cancer kinase inhibitor that is amongst the largest selling small molecule drugs; 7N7X contains berotralstat, a recently approved drug that prevents hereditary angioedema. Note that for 4EYR, the inpainting was done using the ligand geometry from the PDB entry 3NDX[6], but docking was performed with 4EYR because the protein structure in 3NDX contained issues – nonetheless, both entries contain the same protease and ligand.

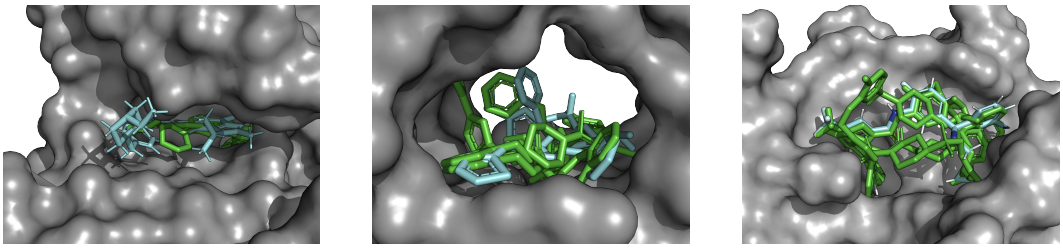
In addition to the experiments in Section 5.2, we evaluate SYNCOGEN's conditional sampling performance for the fragment linking framework against the state-of-the-art model DiffLinker (Igashov et al., 2024). While DiffLinker is trained for fragment-linking, our model performs zero-shot fragment linking without any finetuning. For both models, the size of the linker was chosen so that it matches

[3] `https://www.rcsb.org/structure/7N7X`
[4] `https://www.rcsb.org/structure/5L2S`
[5] `https://www.rcsb.org/structure/4EYR`
[6] `https://www.rcsb.org/structure/3NDX`

that of the original ligand: 2 extra nodes were sampled for SYNCOGEN and 15 linking atoms for DiffLinker in the case of 5L2S, while 3 extra nodes and 25 linking atoms were sampled for 4EYR and 7N7X. We specified leaving groups (for SYNCOGEN) and anchor points (for DiffLinker) so that the fragments are linked at the same positions as in the ligand. Results are shown in Table 6. No retrosynthetic pathways were found for the molecules in DiffLinkers, while SYNCOGEN models synthetic pathways and synthetic pathways can be easily drawn, with examples for 4EYR shown in Figure 12. This out-of-distribution task for SYNCOGEN leads to fewer valid molecules; however, for the valid candidates, SYNCOGEN has lower interaction energies and achieves 100% connectivity as it uses reaction-based assembly, whereas DiffLinker can sample disconnected fragments.

Table 6: Molecular inpainting task. Results are averaged over 1000 generated samples, except retrosynthesis solve rate (out of 100). SYNCOGEN-FT denotes a light fine-tuning the model for 5 epochs on in-painting with randomly fixed fragments.

Method	Target	AiZyn. ↑	Synth. ↑	Valid. ↑	Connect. ↑	MMFF ↓	GFN-FF ↓	GFN2-xTB ↓	Diversity ↑	PB ↑
DiffLinker	5L2S	0	0	95.8	95.09	14.22	7.52	-0.95	0.60	49.3
	4EYR	0	0	93.7	81.86	20.01	8.49	-1.03	0.81	35.0
	7N7X	0	0	95.8	74.65	20.51	7.99	-1.09	0.78	37.5
SYNCOGEN	5L2S	73	79	57.6	100	10.11	6.77	-0.78	0.62	27.3
	4EYR	72	58	46.9	100	12.80	6.58	-0.86	0.64	32.0
	7N7X	53	69	50.6	100	4.243	6.60	-0.80	0.67	56.1
SYNCOGEN-FT	5L2S	77	84	75.3	100	4.25	6.58	-0.81	0.632	56.2
	4EYR	42	78	62.0	100	10.13	5.33	-0.78	0.604	19.8
	7N7X	57	77	73.6	100	4.09	6.86	-0.83	0.664	47.9



(a) 5L2S (b) 3NDX/4EYR (c) 7N7X

Figure 11: Structural overlays of the native protein (gray) and its native ligand (blue) with AlphaFold3-predicted folds of a subset of generated ligands (green) for (a) 5L2S, (b) 3NDX/4EYR, and (c) 7N7X.

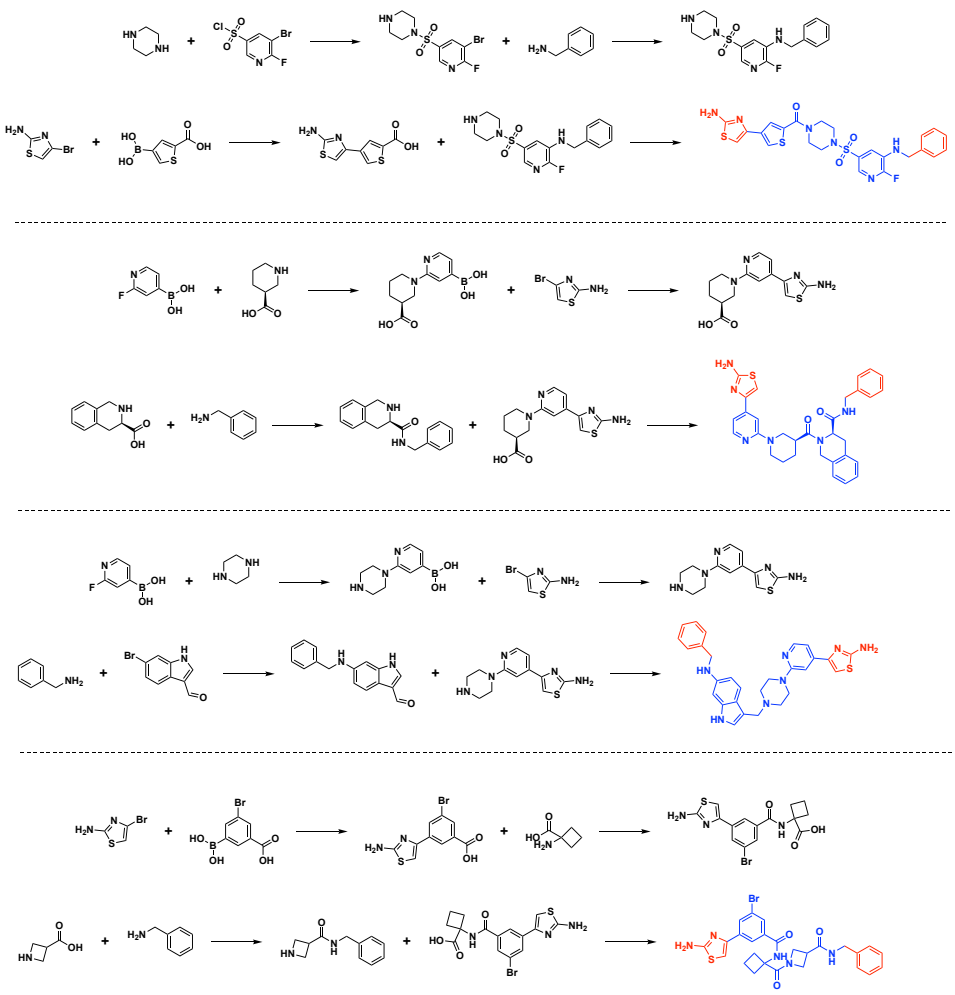


Figure 12: Synthetic pathways for molecules generated in the molecular inpainting task for target 3NDX/4EYR. The final product is shown in blue, and the inpainted fragments are shown in red.