STRUCTURE-CONSTRAINED MOLECULAR DESIGN WITH DISCRETE GRAPH BRIDGE DIFFUSION

Anonymous authors
Paper under double-blind review

ABSTRACT

Structure-constrained molecular design, such as decorating a scaffold, extending a motif, or linking fragments, requires generating around a fixed substructure while leaving it untouched. Diffusion models are a natural fit for this: unlike autoregressive language models, they can enforce such constraints directly by construction, simply by clamping the prescribed subgraph during sampling. However, much of the diffusion literature has focused instead on de novo generation or property optimization via guidance, with structural constraints more often imposed at inference time than trained for directly. We revisit this setting along two complementary directions. The first focuses on the role of preprocessing, examining how much of the apparent gap between graph diffusion and chemical language models can be attributed to preprocessing rather than architecture. This alone yields a substantially stronger constrained-generation baseline than prior graph diffusion work. The second, Graph Bridge Diffusion (GBD), goes further by training the model to see the constraint throughout the generative process rather than only at inference, closing a train–inference mismatch that every prior clamping-based approach leaves open. The result is a single model that serves a variety of structure-based generation tasks without task-specific retraining, and that better matches the reference chemical series in composition, not just in satisfying the constraint. Trained on 1.6M molecules with 16M parameters, GBD performs competitively with chemical language models with $5\times$ more parameters and over 1.1B molecular strings for pretraining, while outperforming graph diffusion baselines at matched budgets.

1 INTRODUCTION

Molecular generative models are mostly benchmarked on *de novo* design: sampling a molecule and checking that it is valid and drug-like. In practical drug discovery, once a hit compound is found, chemists edit it (such as decorating a scaffold, replacing a linker, or growing from a motif) while keeping the active substructure intact. The retained substructure is a requirement rather than a preference, and benchmarks such as SAFE (Noutahi et al., 2023) and MolGenBench (Cao et al., 2025) evaluate models on this form of constrained editing.

Graph diffusion models are conceptually well-suited to this task: because they build molecules atom by atom and bond by bond, substructure constraints can be expressed by holding specific values fixed. In practice, however, these constraints are external to training. Most known models are trained to reverse uniform corruption across all nodes and edges, with the substructure imposed only during sampling by *clamping* prescribed entries at each reverse step (Vignac et al., 2023; Qin et al., 2025). As a result, at time t, the denoiser receives a graph that is clean in anchored regions and noisy elsewhere, creating a state absent from its training distribution. This error is intrinsic to the forward process and persists even with an ideal denoiser (Wu et al., 2023), yet its empirical magnitude on molecular graphs has not been measured.

This mismatch leads to severe performance degradation: DeFoG (Qin et al., 2025)—a state-of-the-art model for de novo molecular generation, where it reaches strong validity (Appendix B.1)—achieves a validity of only 0.08 when the same model is instead sampled under a scaffold constraint on MolGenBench, compared to 0.52–0.96 achieved by models built for constrained generation. Existing workarounds either train a separate model per task (Xie et al., 2024; Igashov et al., 2024b), resulting in single-purpose tools, or abandon graphs for fragment-ordered strings (Noutahi et al., 2024; Lee et al., 2025), which requires pretraining on over 1.1B strings.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

We trace this performance gap to two independent sources and address both. The first is *preprocessing*: kekulising aromatic bonds (Nikitin et al., 2025) removes a bond category that the denoiser cannot handle consistently during constrained sampling. This adjustment alone raises MolGenBench validity from 0.08 to 0.93 without modifying the model architecture or training procedure, and applies directly to any graph diffusion model. The second is the *train–inference mismatch* itself: even after kekulisation, clamping queries the denoiser off-distribution. We address this issue by incorporating the substructure directly into the forward process.

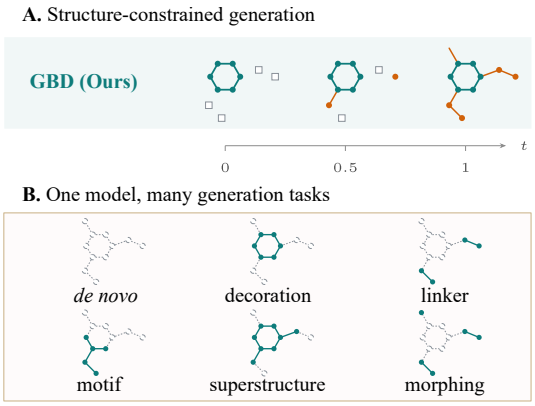


Figure 1: GBD: one model for the hit-to-lead task family. Generation starts from the given substructure, with every other atom and bond unassigned, and ends at a complete molecule containing it; the anchored part stays exact at every step by construction. The anchor alone selects the task.

Our model, Graph Bridge Diffusion (GBD), starts with the prescribed subgraph fixed and all unspecified positions in an absorbing state, then progressively generates the missing structure around it (Figure 1). By training on trajectories between these endpoints (partial-to-complete graph), the denoiser remains in-distribution at inference. Since tasks differ only in their anchor set, a single model handles diverse structure-constrained generation, while an empty anchor set recovers unconditional generation. Randomising the anchor set during training extends classifier-free dropout to substructure conditioning (Ho & Salimans, 2022), allowing one checkpoint to serve all editing patterns.

The bridge formulation delivers its main gains beyond raw validity, which preprocessing largely resolves. It improves functional-group fidelity and synthetic accessibility, and it degrades more gracefully under tight sampling budgets, requiring substantially fewer denoising steps than clamping-based alternatives. These metrics are critical for hit-to-lead optimization, where a scaffold-valid molecule with inappropriate functional groups or ring systems may be of little practical value.

While pretrained chemical language models achieve higher performance on some subtasks and metrics, they require substantially larger resources, with roughly $700\times$ more data and $5\times$ more parameters. Our focus is instead on *coverage and efficiency*: a single 16M-parameter model that handles diverse structure-constrained editing tasks without task-specific retraining, achieves strong structural fidelity, and does so with substantially fewer resources than competing approaches.

Our contributions are as follows:

(1) **We diagnose and address a failure mode of constrained graph diffusion.** We identify two key failure sources—molecular representation and the train–inference mismatch induced by sampling-time clamping—and show that preprocessing alone substantially improves validity.

(2) **We propose Graph Bridge Diffusion for structure-constrained generation.** We integrate the prescribed substructure directly into the generative process and prove that our objective recovers the exact conditional distribution, while inference-time clamping can produce a biased one.

(3) **We evaluate a single compact model across diverse tasks and budgets.** A 16M-parameter model trained on 1.6M molecules matches or exceeds chemical language models trained with roughly $700\times$ more data on several metrics, while bridge conditioning maintains strong performance with $10\times$ fewer steps than clamping-based graph diffusion.

2 Related Work

Fragment-constrained molecular design. The five editing tasks differ mainly in which part of the molecule is fixed. SAFE defines them over shared drug-derived fragments (Noutahi et al., 2024), while MolGenBench evaluates scaffold decoration against congeneric series (Cao et al., 2025). Existing approaches primarily follow two paradigms. The first uses 3D representations, requiring conformers and, typically, a protein pocket. DiffDec and DiffLinker target individual tasks, while Delete and DiffSBDD support multiple tasks from one checkpoint (Xie et al., 2024; Igashov et al., 2024b; Zhang et al., 2023; Schneuing et al., 2024). ShEPhERD requires a conformer but not a

pocket, while PGMG conditions on a pharmacophore rather than a substructure (Adams et al., 2025; Zhu et al., 2023). The second encodes constraints in token sequences and checks them after generation: SAFE filters samples by RDKit substructure matching (Noutahi et al., 2024), while GenMol's linker validity rose from 0.167 to 0.818 after revising its SAFE syntax to disambiguate attachment points (Lee et al., 2025). Thus, constraint satisfaction is a success rate rather than a guarantee, with violating samples discarded. In GBD, the anchored subgraph is fixed input rather than model output.

Sampling-time clamping. Overwriting constrained entries at each reverse step is an approximate way to sample conditionally from an unconditional model (Song et al., 2021), but can produce locally consistent yet globally inconsistent completions (Lugmayr et al., 2022; Ho et al., 2022). String models use the same approach for fragment constraints, which InVirtuoGen finds can hinder refinement (Kaech et al., 2026). For continuous diffusion, Trippe et al. (2023) show that the error persists with an exact denoiser and grows with target correlation, since it arises from the forward process. Graph models show the same symptom: Vignac et al. (2023) clamp a subgraph at every reverse step and report long-range consistency failures. The constraint itself is never violated, since the entries are simply overwritten, and sampling-time projection reaches near-total satisfaction for other constraint families too (Sharma et al., 2024). What stays open is how far completions drift from the true conditional. GBD removes its cause by training on the conditional it is later asked for.

Anchoring the forward process. Keeping selected coordinates clean during noising and training on this conditional is partial noising, introduced for text diffusion (Gong et al., 2023). Conditional 3D molecular diffusion uses this formulation in DiffLinker (Igashov et al., 2024b), while FLOWR.MULTI trains a single pocket-conditioned model for several editing modes (Cremer et al., 2026). Randomising held coordinates is also established, using uniform subsets in discrete sequence models (Hoogeboom et al., 2022) and structured masks in 3D (Lin et al., 2024; Xu et al., 2025). Putting the constraint in the process rather than the sampler is ConStruct's argument before ours (Madeira et al., 2024); its edge-absorbing process admits only constraints that survive edge deletion, so it cannot express a prescribed labelled subgraph. Anchoring a subset yields a data-dependent coupling (Albergo et al., 2024), which we call a *bridge* because it bridges noise and data conditioned on a prescribed subgraph, though it differs from classical diffusion bridges that pin the full state at both endpoints (Zhou et al., 2024; Igashov et al., 2024a). The mismatch also depends on the noise model: with absorbing noise, unmasked entries are clean, so clamping matches the reverse kernel (Austin et al., 2021; Sahoo et al., 2024; Ou et al., 2025), and GenMol conditions on fragments with no change to its process (Lee et al., 2025). With marginal transitions, clean and identically resampled entries are indistinguishable (Vignac et al., 2023; Qin et al., 2025). Lim et al. (2020) and Maziarz et al. (2022) train on constraints but grow from a connected seed. We are not aware of a graph model conditioning on a possibly disconnected anchor set over both nodes and edges, as GBD does.

3 METHODOLOGY

3.1 THE ANCHORED CONDITIONAL

A graph on n nodes is a pair $G = (X, E)$ with node types $x^{(i)} \in \mathcal{X}$ for $i \in [n]$ and edge types $e^{(ij)} \in \mathcal{E}$ for $i < j$, where $\mathcal{E}$ contains a *no-bond* symbol so that E is a complete labelling of the pairs. We write G_t for the state at time t and index coordinates by $\mathcal{C} = [n] \cup \{(i, j) : i < j\}$.

A constraint is a labelled subgraph S: a set of nodes $V_S \subseteq [n]$, a node type for each $i \in V_S$, and an edge type for each pair inside V_S. It partitions the coordinates into the *anchored* set

$$\mathcal{A} \;=\; V_S \cup \{(i, j) : i, j \in V_S\}, \qquad \mathcal{F} \;=\; \mathcal{C} \setminus \mathcal{A}, \tag{1}$$

and we write $G[\mathcal{A}] = S$ for the event that the anchored coordinates carry the prescribed labels. Among the free coordinates, the pairs with exactly one endpoint in V_S,

$$\mathcal{B} \;=\; \{(i, j) \in \mathcal{F} : |\{i, j\} \cap V_S| = 1\} \;\subset\; \mathcal{F}, \tag{2}$$

carry the attachment of the constraint to the rest of the molecule. They are free: a constraint fixes which atoms and which internal bonds are present, not where the molecule grows from them. Each task that we consider is an instance of the same object,

$$\pi_S \;=\; p_{\mathrm{data}}(\,\cdot \mid G[\mathcal{A}] = S\,), \tag{3}$$

under a different $\mathcal{A}$, and $\mathcal{A} = \emptyset$ recovers p_{data}.

Limitation of clamping. A denoising generative model is specified by a family of reverse kernels driven by a predicted posterior over clean graphs. Sampling from Equation 3 requires the posterior conditioned on the constraint, $p_{1|t}(G_1 \mid G_t, G_1[\mathcal{A}] = S)$. A model trained without constraints instead supplies $p_{1|t}(G_1 \mid G_t)$. Clamping enforces the prescribed values on $\mathcal{A}$ after each reverse step, but leaves the model's updates on the free coordinates $\mathcal{F}$ unchanged. These updates use the unconstrained posterior, rather than the posterior conditioned on $G_1[\mathcal{A}] = S$, so they need not follow the correct conditional distribution. Proposition 2 makes this precise, showing that the constrained and unconstrained posteriors agree for all t and states only when the anchored labels carry no information about the free coordinates, in which case the constraint would not be worth imposing.

3.2 Discrete flow matching on graphs

We build on DeFoG (Qin et al., 2025), a discrete flow matching model (Campbell et al., 2024; Gat et al., 2024) for graphs. Time runs from noise to data: $t = 0$ is the prior and $t = 1$ is the data distribution, the reverse of the convention usual in diffusion.

For a single discrete variable z with values in a finite set $\mathcal{Z}$ and a noise distribution $p_0 \in \Delta^{|\mathcal{Z}|}$, the conditional probability path is the linear interpolation

$$p_{t|1}(z_t \mid z_1) \;=\; t\,\delta(z_t, z_1) \;+\; (1-t)\,p_0(z_t). \tag{4}$$

A graph is noised coordinatewise: given G_1,

$$p_{t|1}(G_t \mid G_1) \;=\; \prod_{i\in[n]} p_{t|1}\Big(x_t^{(i)} \mid x_1^{(i)}\Big) \prod_{i<j} p_{t|1}\Big(e_t^{(ij)} \mid e_1^{(ij)}\Big)\,. \tag{5}$$

Generation is a continuous-time Markov chain run forward in t. For each coordinate there is a rate matrix $R_t^\star(\cdot \mid z_1)$ generating Equation 4 for a fixed endpoint (Qin et al., 2025, Equation 3), and the rate that generates the marginal path is its expectation under the posterior over clean values,

$$R_t(z_t, j) \;=\; \mathbb{E}_{z_1 \sim p_{1|t}(\cdot|z_t)}[\,R_t^\star(z_t, j \mid z_1)\,], \qquad p_{t+\Delta t|t}(j \mid z_t) = \delta(z_t, j) + R_t(z_t, j)\,\Delta t. \tag{6}$$

The posterior is the only unknown. A network f_θ predicts it for every node and edge, written $\boldsymbol{p}_{1|t}^\theta(\cdot \mid G_t)$, and is trained by cross-entropy against the clean graph,

$$\mathcal{L} \;=\; \mathbb{E}_{t,\,G_1\sim p_{\mathrm{data}},\,G_t\sim p_{t|1}(\cdot|G_1)}\,\mathrm{CE}_\lambda\Big(G_1,\,\boldsymbol{p}_{1|t}^\theta(\cdot \mid G_t)\Big)\,, \tag{7}$$

with CE_λ summing node and edge cross-entropies at relative edge weight $\lambda > 0$. Number of nodes is drawn from the empirical distribution of the training set and held fixed along the trajectory.

3.3 Anchoring the process

The constraint enters as a restriction on the coupling between the two endpoints. Instead of drawing G_0 from the prior independently of G_1, we draw the pair from

$$\pi(G_0, G_1 \mid S) \;=\; \underbrace{\pi_S(G_1)}_{\text{contains } S} \cdot \underbrace{\delta(G_0[\mathcal{A}], S)}_{\text{copied}} \cdot \prod_{c\in\mathcal{F}} p_0\Big(z_0^{(c)}\Big)\,, \tag{8}$$

a data-dependent coupling in the sense of Albergo et al. (2024), supported on pairs that agree on $\mathcal{A}$. The conditional path is Equation 5 on the free coordinates and constant on the anchored ones,

$$p_{t|1}(G_t \mid G_1, S) \;=\; \delta(G_t[\mathcal{A}], S) \prod_{c\,\in\,\mathcal{F}} p_{t|1}\Big(z_t^{(c)} \mid z_1^{(c)}\Big)\,, \tag{9}$$

so that $G_t[\mathcal{A}] = S$ holds at every t, including $t = 0$. The corresponding generator is Equation 6 on $\mathcal{F}$, with the posterior now conditioned on the constraint, and zero on $\mathcal{A}$:

$$R_t\Big(z_t^{(c)}, j \mid S\Big) = \begin{cases} \mathbb{E}_{G_1\sim p_{1|t}(\cdot|G_t,S)}\left[\,R_t^\star\Big(z_t^{(c)}, j \mid z_1^{(c)}\Big)\right], & c \in \mathcal{F}, \\ 0, & c \in \mathcal{A}. \end{cases} \tag{10}$$

The denoiser is trained under the same conditional, with the anchored coordinates supplied as input:

$$\mathcal{L}_{\mathrm{anc}} \;=\; \mathbb{E}_{t,\,S,\,G_1\sim\pi_S,\,G_t\sim p_{t|1}(\cdot|G_1,S)}\,\mathrm{CE}_\lambda\Big(G_1,\,\boldsymbol{p}_{1|t}^\theta(\cdot \mid G_t, S)\Big)\,. \tag{11}$$

The training and sampling distributions now coincide. The denoiser sees graphs whose anchored block is clean and whose free block is noised to level t during training, and is queried on graphs of that form at inference, for the same $\mathcal{A}$ and the same S.

Proposition 1. *Fix S and let f_θ minimise Equation 11, so that $\boldsymbol{p}^\theta_{1|t}(\cdot \mid G_t, S) = p_{1|t}(\cdot \mid G_t, S)$ on $\mathcal{F}$ for p_t^S-almost every G_t. Then the chain generated by Equation 10 started from $G_0 \sim \pi(\cdot \mid S)$ satisfies $G_t[\mathcal{A}] = S$ for all $t \in [0,1]$ and has terminal law π_S.*

The proof is in Appendix A and is short: the anchored coordinates carry zero rate, so the process restricted to $\mathcal{F}$ is an ordinary discrete flow matching process with data distribution π_S and S as fixed context, and the correctness of that process is Campbell et al. (2024). The content of the proposition is not the argument but the contrast with Proposition 2: the same exactness does not hold for clamping, for any denoiser, and the discrepancy is present at every t.

3.4 One process, many anchors

Nothing above fixes $\mathcal{A}$. Training on a single anchor pattern is task-specific; instead, we randomize $\mathcal{A}$ during training to obtain a single model applicable across tasks. For each molecule, with probability p we take $\mathcal{A} = \emptyset$, with probability q the Bemis–Murcko scaffold, and otherwise a subset of its BRICS fragments, of a size drawn uniformly over the available fragments. The three cases are the unconditional model, the decoration tasks and the fragment tasks respectively. This is dropout over substructures rather than over labels, and it plays the role that dropping the class label plays in classifier-free guidance (Ho & Salimans, 2022): one checkpoint is trained on a family of conditionals and any member of the family can be requested at inference.

The family is broader than the benchmark: at sampling time, $\mathcal{A}$ can be any labelled subgraph, connected or not, with $\mathcal{A} = \emptyset$ recovering *de novo* generation from the same weights – though we only evaluate anchors of the types used during training (scaffolds and fragments; see limitations).

The mixture (p, q) is a choice and not a quantity to be maximised. Raising the mass on a single pattern improves the matching task and costs the others (Table 3); we use $p = 0.2$, $q = 0.4$.

Algorithm 1: GBD training

Input: dataset $\mathcal{D}$; anchor probabilities p, q with $p + q \le 1$
while *not converged* **do**
  $G \sim \mathcal{D}$; $r \sim \mathrm{Uniform}(0,1)$;
  $G \leftarrow \mathrm{Kekulize}(\mathrm{RemoveH}(G))$; `// preprocessing`
  if $r < p$ **then**
    $S \leftarrow \emptyset$; `// unconditional`
  else if $r < p + q$ **then**
    $S \leftarrow \mathrm{Scaffold}(G)$; `// decoration`
  else
    $S \leftarrow \mathrm{Sample}(\mathrm{BRICS}(G))$; `// fragment-based`
  $\mathcal{A} \leftarrow$ coordinates of S; $t \sim \mathcal{T}$;
  $G_t \sim p_{t|1}(\cdot \mid G, S)$; `// Equation 9`
  $h \leftarrow \mathrm{RRWP}(G_t)$; `// graph features`
  $\boldsymbol{p}^\theta_{1|t} \leftarrow f_\theta(G_t, h, t, \mathbf{1}_\mathcal{A})$;
  step on $\mathrm{CE}_\lambda(G, \boldsymbol{p}^\theta_{1|t})$; `// Equation 11`

Algorithm 2: GBD sampling

Input: anchor S (possibly empty); steps $1/\Delta t$
$n \sim \hat{p}(n \mid S)$; $G_0 \sim \pi(\cdot \mid S)$; `//  Equation 8`
for $t = 0$ **to** $1 - \Delta t$ ***step*** Δt **do**
  $h \leftarrow \mathrm{RRWP}(G_t)$; `// graph features`
  $\boldsymbol{p}^\theta_{1|t} \leftarrow f_\theta(G_t, h, t, \mathbf{1}_\mathcal{A})$;
  $G_{t+\Delta t} \sim \delta(G_t, \cdot) + R_t(\cdot \mid S)\,\Delta t$;
  `// Eq. Equation 10, A inert`
$G_1 \leftarrow \mathrm{AddH}(\mathrm{Aromatize}(G_1))$; `// postprocessing`
return G_1

3.5 Graph-specific issues

Exempting the conditioning variables from corruption is how conditional generation is done in sequence (Gong et al., 2023) and point-cloud (Igashov et al., 2024b) models. On attributed graphs the same idea meets a few obstacles that have no analogue there.

Boundary edges. An anchor fixes node and edges types inside $\mathcal{A}$, and leaves $\mathcal{B}$ free. These are the coordinates on which the constraint and the generated part meet, and they are generated under a prior in which *no-bond* is the dominant symbol, so the model must decide both where the molecule attaches and with what bond order. In our process $\mathcal{B}$ belongs to $\mathcal{F}$ in the coupling Equation 8, the path Equation 9 and the objective Equation 11, so attachment is learned under the conditional rather than prescribed alongside the constraint.

Graph size. The number of free nodes is not determined by S, and unlike image inpainting there is no canvas whose remaining area is known. We draw n once before sampling, from an empirical

conditional $\hat{p}(n \mid S)$ estimated as the distribution of atom counts among the reference molecules associated with the given anchor, and hold n fixed along the trajectory. This gives GBD access to the size statistics of the reference series, which autoregressive baselines determine themselves during decoding; we flag this as a comparison limitation, not a method limitation.

Aromatic bonds. Common graph diffusion models (Vignac et al., 2023; Qin et al., 2025) represent aromatic bonds as a separate type alongside single, double, and triple bonds. But aromaticity is a ring-level property, so the label is only well defined once the ring is complete, while standard validity checks use different valence contributions (Nikitin et al., 2025). Under clamping, prescribed aromatic bonds depend on atoms not yet generated, so the sampler cannot satisfy their valence constraints. Kekulising the dataset removes the category: aromatic rings are written as alternating bonds, aromaticity is restored by RDKit after generation, and the bond vocabulary drops to three types. This changes no part of the architecture, the objective or the sampler. It raises MolGenBench validity from 0.08 to 0.93. Nikitin et al. (2025) report the same preprocessing improving validity for unconstrained 3D generators; in the constrained setting the effect is an order of magnitude, because clamping is what turns an ambiguous label into an unsatisfiable constraint.

Training and sampling are summarised in Algorithm 1 and Algorithm 2, respectively.

4 Experiments

4.1 Setup

Benchmarks. We evaluate on two benchmarks for structure-constrained molecular generation. First, **MolGenBench** (Cao et al., 2025) provides a large-scale scaffold decoration benchmark covering 120 biological targets, with five congeneric series per target, yielding 600 distinct examples. Following Cao et al. (2025), we generate 200 completions per scaffold and evaluate using *Validity*, *QED*, *SA score*, *Uniqueness*, *Diversity*, and three distributional fidelity metrics measuring Jensen–Shannon divergence between generated and reference distributions of atom types (*AtomType*), ring systems (*RingType*), and functional groups (*FunctionalGroup*). Note that Validity measures whether a generated molecule is valid and contains desired substructure. Second, **SAFE** (Noutahi et al., 2023) evaluates five fragment-constrained generation tasks: scaffold decoration, motif extension, linker generation, superstructure generation, and scaffold morphing, using 26 starting molecules. We note that the SAFE examples are not necessarily structures encountered during training, even with fragment-based sampling, as the specific fragment combinations in the benchmark may be new to the model. The same core metrics are used, while distributional fidelity measures are unavailable for SAFE due to its smaller benchmark size.

Baselines. We compare three baseline families. **Chemical language models** represent molecules as strings and enforce fragment constraints via token masking. SAFE-GPT (Noutahi et al., 2023) and GenMol (V2) (Lee et al., 2025) are pretrained on large-scale corpora using the SAFE representation. **Graph diffusion models** operate directly on molecular graphs: DiGress (Vignac et al., 2023) and DeFoG (Qin et al., 2025) are our primary diffusion baselines which we run ourselves. **Chemistry-specific baselines**: PGMG (Zhu et al., 2022), Delete (Zhang et al., 2023), DiffSBDD (Schneuing et al., 2024), DiffDec (Xie et al., 2024), and ShEPhERD (Adams et al., 2025). For these methods, we report results published by Cao et al. (2025), evaluated under MolGenBench protocol. We treat these numbers as contextual reference points rather than a controlled comparison: they indicate the range of performance purpose-built, task-specific methods achieve on this benchmark.

Training data. All models trained by us use the MOSES dataset (Polykovskiy et al., 2020), a standard collection of approximately 1.9M drug-like molecules filtered from ZINC, with 1.6M molecules in the training split. For the text-based models, we use the publicly available pretrained checkpoints, which were trained on more than 1.1B SAFE strings, as models trained exclusively on MOSES perform substantially worse, as shown in experiments below.

Preprocessing. We restrict the atom and bond vocabulary to 11 atom types and 3 bond types, covering all benchmark molecules and nearly all training data; molecules outside this vocabulary are discarded. Molecules are Kekulized using RDKit[1] (Section 3.5), with explicit hydrogens omitted

[1]RDKit: Open-source cheminformatics. `https://www.rdkit.org`

during training. After generation, aromaticity and implicit hydrogens are restored. Molecular scaffolds are extracted using the Bemis–Murcko decomposition, and BRICS fragmentation is applied to obtain fragment-based subgraphs for training.

Training. We follow the hyperparameters of DeFoG. Models are trained for 10 epochs (at which point training loss has converged), using the AdamW optimiser with learning rate 2×10^{-4}, batch size 64, and weight decay 10^{-12}. Training takes around 2 days on a RTX 20-series GPU. The parameter λ in Equation 7 weights the cross-entropy losses for nodes and edges by 1 and 5, respectively. Subgraph dropout probabilities p and q (Algorithm 1) are set to $p = 0.2$, $q = 0.4$ (Section 4.4).

Sampling. We follow DeFoG's (Qin et al., 2025) sampling strategy and use stochasticity parameter $\eta = 10$, which controls the amount of re-noising injected at each step. We tested other values but observed no improvement, so we retain DeFoG's default setting for a fair comparison.

Code available at `https://anonymous.4open.science/r/gbd-F868`.

4.2 MolGenBench: Quantitative Evaluation of Scaffold Decoration

Results are reported in Table 1. Each entry averages over 200 generated graphs per scaffold across 600 scaffolds; given this scale, we report single-seed results. Three findings stand out.

Table 1: MolGenBench scaffold decoration results, with higher scores indicating better performance. Note that GBD w/o bridge is equivalent to DeFoG with our preprocessing. Best score in bold; note that the remaining metrics are conditioned on validity.

			Molecular Quality					Structural Fidelity		
Category	**Data**	**Model**	**Validity**	**Uniqueness**	**Diversity**	**QED**	**SA**	**Atom**	**Ring**	**FG**
Autoregressive	**Text**	Pretrained SAFE-GPT	0.887	0.584	0.456	0.574	0.758	0.898	0.902	0.676
		trained from scratch	0.364	0.258	0.243	0.713	0.801	0.877	0.885	0.650
		Pretrained GenMol	**0.958**	0.791	0.586	0.610	0.727	**0.920**	0.918	0.709
		trained from scratch	0.774	0.025	0.016	0.665	**0.837**	0.873	0.870	0.612
	Graphs	PGMG*	0.939	**1.000**	0.752	0.474	0.808	0.889	0.873	0.635
		Delete*	0.728	0.925	0.499	0.484	0.68	0.894	0.907	0.711
Diffusion	**Molecular Graphs**	DiffSBDD-M*	0.768	0.656	0.600	0.571	0.714	0.913	0.911	0.698
		DiffSBDD-C*	0.761	0.751	0.675	0.554	0.696	0.913	0.922	0.705
		DiffDec*	0.678	0.746	0.513	0.565	0.685	0.893	0.903	0.678
		ShEPhERD*	0.521	0.976	0.812	0.533	0.593	0.899	0.911	0.64
	Graphs	DiGress, $T = 500$	0.087	0.981	**0.828**	0.706	0.691	0.881	0.899	0.650
		DeFoG, $T = 500$	0.081	0.645	0.597	**0.982**	0.832	0.861	0.897	0.642
		GBD w/o bridge, $T = 500$	0.926	0.917	0.749	0.539	0.615	0.900	0.968	0.718
		GBD, $T = 500$	0.945	0.927	0.706	0.540	0.748	**0.920**	**0.985**	**0.777**
		GBD, $T = 50$	0.900	0.934	0.723	0.535	0.724	0.919	0.984	0.773

Graph diffusion without our preprocessing fails outright. Under standard preprocessing and clamping, DiGress and DeFoG achieve only 0.087 and 0.081 validity, respectively, despite high validity in de novo generation (Appendix B.1). As discussed in Section 3.5, aromatic bonds combined with sampling-time clamping cause near-total failure. GBD instead achieves 0.945 validity with matched training data and model size. Benefits of our bridge conditioning, independent of preprocessing choices, are further supported by the no-bridge ablation discussed in Section 4.4.

GBD matches or exceeds purpose-built and large-scale pretrained baselines on validity and structural fidelity. Against pretrained text models, GBD is comparable in validity and achieves higher structural fidelity, despite using 1.6M molecules and 16M parameters versus 1.1B SAFE strings and 87–89M parameters. Against the chemistry-specific baselines, GBD falls within their validity range and exceeds them on every structural fidelity metric, despite lacking their 3D or pocket information – though we caution this comparison is not run under matched splits or protocols, and should be read as indicative rather than conclusive. Fidelity measures capture whether generated molecules preserve structural characteristics of the reference chemical series, rather than merely producing valid and drug-like molecules. Additionally, GBD requires much fewer denoising steps than diffusion baselines, approaching the step counts of autoregressive models (Section 4.4).

MOSES-trained text models underperform their pretrained counterparts. Training SAFE-GPT and GenMol on MOSES substantially reduces performance versus their pretrained counterparts. We tested several model sizes and training configurations but observed little difference in performance. On de novo generation (Appendix B.1) and several SAFE tasks, these models achieve

non-trivial validity, indicating that they learn a usable molecular distribution from MOSES alone; we therefore do not believe that their performance is primarily limited by undertraining. These results highlight the importance of large-scale pretraining for text-based models in this setting, while GBD achieves competitive performance using substantially less resources.

4.3 SAFE Benchmark: Various Fragment-Constrained Tasks

Results across the five SAFE tasks are shown in Figure 2, with examples of generated molecules and intermediate states illustrated in Figure 3.

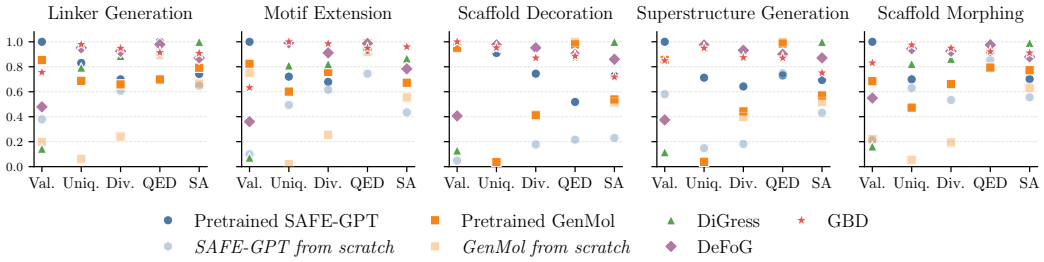
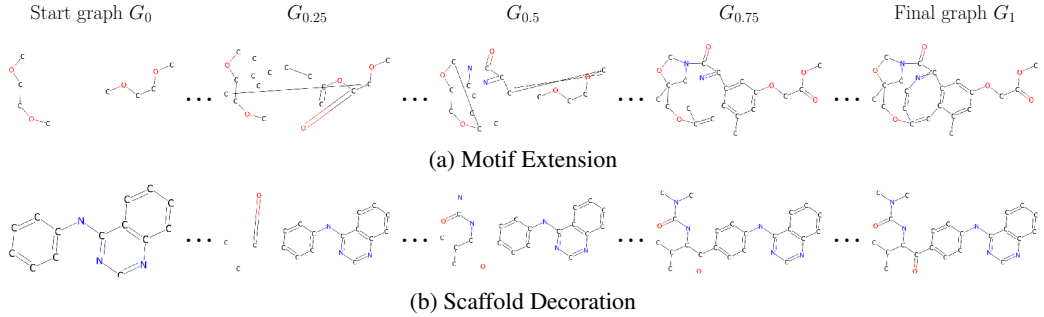


Figure 2: Performance on SAFE tasks averaged over 5 runs; numerical results in Appendix B.2



(a) Motif Extension

(b) Scaffold Decoration

Figure 3: Example generations from GBD on SAFE; other tasks are shown in Appendix B.3.

GBD leads in the more structurally open tasks and remains competitive elsewhere. For scaffold decoration and superstructure generation, GBD achieves 0.825 and 0.782 validity, respectively, outperforming the graph baselines and MOSES-trained text models and approaching the pretrained text models. On linker generation, motif extension, and scaffold morphing, GBD trails the pretrained text models in validity but consistently outperforms DiGress and is competitive with DeFoG. It also achieves overall the highest uniqueness and diversity among models. Thus, GBD is the strongest graph-based model and narrows the gap to text models pretrained on substantially larger datasets.

QED and SA trade off against structural exploration. GBD does not lead on QED or SA, typically trailing the pretrained text models by a moderate margin. This coincides with the higher uniqueness and diversity achieved by GBD, suggesting broader exploration around the constrained fragment, while the pretrained text models favor more drug-like completions.

4.4 Ablations: Bridge Conditioning, Number of Steps, and Subgraph Sampling

Effect of bridge training. To isolate the contribution of bridge-specific training, we evaluate GBD (w/o bridge), which uses the same architecture and preprocessing but is trained without subgraph conditioning and applies the constraint only at inference through clamping, as in DeFoG and DiGress. Table 2 shows that bridge conditioning consistently improves validity, SA, and structural fidelity, with the largest gaps at low sampling budgets. The no-bridge model instead achieves higher uniqueness and diversity, particularly at low T. The main gains are in the fidelity measures, which are unavailable for the SAFE tasks and are therefore omitted. Moreover, we observe that conditioning enables strong performance with fewer denoising steps, consistent with benefits observed for conditional diffusion in the image domain (Pandey et al., 2022; Proszewska et al., 2025).

Number of inference steps T. Figure 4 shows the effect of the inference budget. We find $T = 50$ to offer a practical balance between generation quality and inference cost. Although we initially

Table 2: Effect of bridge conditioning on MolGenBench. The better of the two scores is in bold.

Model	T	Molecular Quality					Structural Fidelity		
		Validity	Uniqueness	Diversity	QED	SA	Atom	Ring	FG
GBD	500	**0.945**	**0.927**	0.706	**0.540**	**0.748**	**0.920**	**0.985**	**0.777**
	50	**0.900**	0.934	0.723	0.535	**0.724**	**0.919**	**0.984**	**0.773**
	10	**0.693**	0.949	0.767	0.523	**0.645**	**0.913**	**0.974**	**0.749**
GBD w/o bridge	500	0.926	0.917	**0.749**	0.539	0.615	0.900	0.968	0.718
	50	0.885	**0.965**	**0.796**	**0.538**	0.597	0.898	0.965	0.721
	10	0.583	**0.970**	**0.815**	**0.528**	0.562	0.898	0.959	0.696

used $T = 500$, matching DiGress and DeFoG, our conditional formulation maintains strong performance with substantially fewer denoising steps, consistent with work on efficient conditional diffusion (Pandey et al., 2022; Proszewska et al., 2025) in image domain. We therefore use $T = 50$ as the default, bringing the number of sequential model evaluations closer to typical autoregressive decoding budgets; we focus here on this sequential inference depth rather than wall-clock time, which depends on hardware, batching, and parallelization.

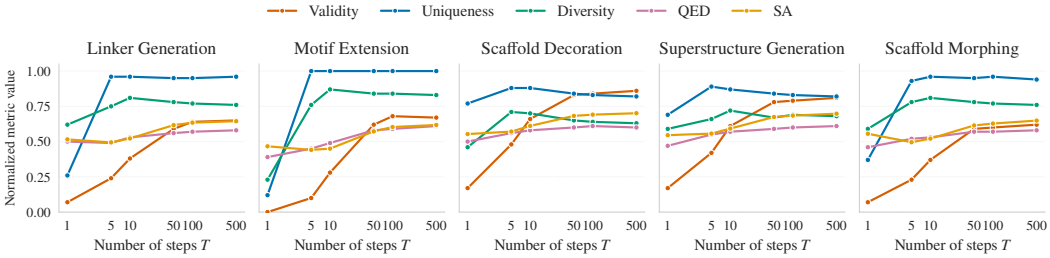


Figure 4: Effect of the number of inference steps T on GBD performance on SAFE benchmark.

Subgraph sampling probabilities. We ablate the subgraph sampling probabilities (p, q) in Table 3 (Section 3.4). Performance is insensitive over a broad range, hence we fix $(.20, .40)$.

Table 3: Validity for anchor sampling probabilities (p, q), corresponding to no subgraph, scaffold, and fragment-based sampling.

	$(.10, .45)$	$(.20, .40)$	$(.33, .33)$	$(.50, .25)$
MolGenBench	0.909	0.901	0.909	0.888
Linker Generation	0.570	0.593	0.587	0.560
Motif Extension	0.604	0.618	0.617	0.596
Scaffold Decoration	0.825	0.825	0.830	0.800
Superstructure Generation	0.785	0.782	0.776	0.766
Scaffold Morphing	0.580	0.585	0.580	0.557
Avg.	**0.712**	**0.717**	**0.716**	**0.694**

5 Conclusions

We revisited structure-constrained molecular design and traced poor graph diffusion performance to two causes: ambiguous aromatic bond representation and train-inference mismatch from sampling-time clamping. Fixing the former substantially improves validity, while GBD addresses the latter by incorporating the constraint into the forward process and recovering the true conditional distribution exactly, something clamping cannot guarantee.

A single 16M-parameter GBD checkpoint, trained on only 1.6M molecules, supports five structure-constrained editing tasks as well as de novo generation without task-specific retraining. Despite its substantially smaller pretraining corpus, it is competitive with chemical language models pretrained on over 1.1B molecular strings, while outperforming purpose-built 3D and pocket-conditioned baselines without access to their additional structural information. Moreover, bridge conditioning degrades more gracefully than clamping under tight inference budgets, closing much of the step-count gap to autoregressive models. Together, these results show that training-time conditioning offers a simple, general alternative to building specialized structure-constrained generative models.

Limitations. GBD performs worse than pretrained text models on some tasks, which is expected given their substantially larger pretraining corpora. Closing this gap without comparable pretraining scale remains open. Our training anchors are limited to scaffolds and BRICS fragments, and we have not evaluated anchors substantially different from those seen during training. Finally, our evaluation is entirely 2D; extending the bridge formulation to 3D-conditioned settings is a natural next step.

AI USE STATEMENT

In this work, we used generative AI tools to provide feedback on research and identify relevant literature. We have not used generative AI tools to develop research ideas, formulate mathematical claims, write proofs, design experiments, or write code used to produce or interpret our results, and generating synthetic datasets and translations are not applicable to this work. Additionally, we used generative AI tools to edit and polish the manuscript, assist with LaTeX formatting, and write code for creating plots. We have reviewed all AI-assisted work. In particular, we reviewed AI-assisted text for accuracy and verified that it faithfully represents our findings and claims; all reported results were produced through our own experiments. We take responsibility for the final content of this work, including all text, claims, and artifacts produced with the assistance of generative AI.

ETHICS STATEMENT

This work aims to improve computational tools for structure-constrained molecular design, a step in early-stage drug discovery. As with other generative models for molecular design, there is a general dual-use consideration: methods for generating candidate molecules with desired properties could in principle be misused to design harmful compounds. We do not believe our specific contribution meaningfully changes this risk profile relative to existing published constrained-generation methods, since our method does not introduce new capabilities for optimizing toxicity or other harmful properties, and property-guided generation is not the focus of this work. All datasets used (MOSES, SAFE) are publicly available and derived from previously published, non-sensitive chemical data.

REPRODUCIBILITY STATEMENT

The code required to reproduce our results is available at `https://anonymous.4open.science/r/gbd-F868` and will be made publicly available upon publication. We provide the key details necessary for reproduction in Section 4.

REFERENCES

Keir Adams, Kento Abeywardane, Jenna C. Fromer, and Connor W. Coley. Shepherd: Diffusing shape, electrostatics, and pharmacophores for bioisosteric drug design. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=KSLkFYHlYg.

Michael S. Albergo, Mark Goldstein, Nicholas Matthew Boffi, Rajesh Ranganath, and Eric Vanden-Eijnden. Stochastic interpolants with data-dependent couplings. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=FFILRGD0jG.

Jacob Austin, Daniel D. Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. Structured denoising diffusion models in discrete state-spaces. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 17981–17993, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/958c530554f78bcd8e97125b70e6973d-Abstract.html.

Andrew Campbell, Jason Yim, Regina Barzilay, Tom Rainforth, and Tommi S. Jaakkola. Generative flows on discrete state-spaces: Enabling multimodal flows with applications to protein co-design. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=kQwSbv0BR4.

Duanhua Cao, Zhehuan Fan, Jie Yu, Mingan Chen, Xinyu Jiang, Xia Sheng, Xingyou Wang, Xiaomin Luo, Dan Teng, and Mingyue Zheng. Benchmarking real-world applicability of molecular generative models from de novo design to lead optimization with molgenbench. *bioRxiv*, 2025. doi: 10.1101/2025.11.03.686215. URL https://www.biorxiv.org/content/early/2025/11/04/2025.11.03.686215.

Julian Cremer, Ross Irwin, Alessandro Tibo, Jon Paul Janet, Simon Olsson, and Djork-Arné Clevert. Flowr: flow matching for structure-aware de novo, interaction-and fragment-based ligand generation. *Nature Computational Science*, pp. 1–10, 2026.

Itai Gat, Tal Remez, Neta Shaul, Felix Kreuk, Ricky T. Q. Chen, Gabriel Synnaeve, Yossi Adi, and Yaron Lipman. Discrete flow matching. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/f0d629a734b56a642701bba7bc8bb3ed-Abstract-Conference.html.

Shansan Gong, Mukai Li, Jiangtao Feng, Zhiyong Wu, and Lingpeng Kong. Diffuseq: Sequence to sequence text generation with diffusion models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/pdf?id=jQj-_rLVXsj.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance, 2022. URL https://arxiv.org/abs/2207.12598.

Jonathan Ho, Tim Salimans, Alexey A. Gritsenko, William Chan, Mohammad Norouzi, and David J. Fleet. Video diffusion models. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/39235c56aef13fb05a6adc95eb9d8d66-Abstract-Conference.html.

Emiel Hoogeboom, Alexey A. Gritsenko, Jasmijn Bastings, Ben Poole, Rianne van den Berg, and Tim Salimans. Autoregressive diffusion models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. URL https://openreview.net/forum?id=Lm8T39vLDTE.

Ilia Igashov, Arne Schneuing, Marwin H. S. Segler, Michael M. Bronstein, and Bruno E. Correia. Retrobridge: Modeling retrosynthesis with markov bridges. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024a. URL https://openreview.net/forum?id=770DetV8He.

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*, 6(4):417–427, 2024b.

Benno Kaech, Luis Wyss, Karsten Borgwardt, and Gianvito Grasso. Refine drugs, don't complete them: Uniform-source discrete flows for fragment-based drug discovery. In *International Conference on Learning Representations*, volume 2026, pp. 120938–120963, 2026.

Seul Lee, Karsten Kreis, Srimukh Prasad Veccham, Meng Liu, Danny Reidenbach, Yuxing Peng, Saee Gopal Paliwal, Weili Nie, and Arash Vahdat. Genmol: A drug discovery generalist with discrete diffusion. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/lee25o.html.

Jaechang Lim, Sang-Yeon Hwang, Seokhyun Moon, Seungsu Kim, and Woo Youn Kim. Scaffold-based molecular design with a graph generative model. *Chemical science*, 11(4):1153–1164, 2020.

Yeqing Lin, Minji Lee, Zhao Zhang, and Mohammed AlQuraishi. Out of many, one: Designing and scaffolding proteins at the scale of the structural universe with genie 2, 2024. URL https://arxiv.org/abs/2405.15489.

Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 11451–11461. IEEE, 2022.

Manuel Madeira, Clément Vignac, Dorina Thanou, and Pascal Frossard. Generative modelling of structurally constrained graphs. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/f82385b8804009f9a81e1a30f1ff14e3-Abstract-Conference.html.

Krzysztof Maziarz, Henry Richard Jackson-Flux, Pashmina Cameron, Finton Sirockin, Nadine Schneider, Nikolaus Stiefl, Marwin H. S. Segler, and Marc Brockschmidt. Learning to extend molecular scaffolds with structural motifs. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. URL https://openreview.net/forum?id=ZTsoE8G3GG.

Filipp Nikitin, Ian Dunn, David Ryan Koes, and Olexandr Isayev. Geom-drugs revisited: toward more chemically accurate benchmarks for 3d molecule generation. *Digital Discovery*, 4(11): 3282–3291, 2025. ISSN 2635-098X. doi: 10.1039/d5dd00206k. URL https://doi.org/10.1039/d5dd00206k.

Emmanuel Noutahi, Cristian Gabellini, Michael Craig, Jonathan S. C Lim, and Prudencio Tossou. Gotta be safe: A new framework for molecular design, 2023. URL https://arxiv.org/abs/2310.10773.

Emmanuel Noutahi, Cristian Gabellini, Michael Craig, Jonathan SC Lim, and Prudencio Tossou. Gotta be safe: a new framework for molecular design. *Digital Discovery*, 3(4):796–804, 2024.

Jingyang Ou, Shen Nie, Kaiwen Xue, Fengqi Zhu, Jiacheng Sun, Zhenguo Li, and Chongxuan Li. Your absorbing discrete diffusion secretly models the conditional distributions of clean data. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=sMyXP8Tanm.

Kushagra Pandey, Avideep Mukherjee, Piyush Rai, and Abhishek Kumar. Diffusevae: Efficient, controllable and high-fidelity generation from low-dimensional latents, 2022. URL https://arxiv.org/abs/2201.00308.

Daniil Polykovskiy, Alexander Zhebrak, Benjamin Sanchez-Lengeling, Sergey Golovanov, Oktai Tatanov, Stanislav Belyaev, Rauf Kurbanov, Aleksey Artamonov, Vladimir Aladinskiy, Mark Veselov, Artur Kadurin, Simon Johansson, Hongming Chen, Sergey Nikolenko, Alan Aspuru-Guzik, and Alex Zhavoronkov. Molecular sets (moses): A benchmarking platform for molecular generation models, 2020. URL https://arxiv.org/abs/1811.12823.

Magdalena Proszewska, Nikolay Malkin, and N. Siddharth. On designing diffusion autoencoders for efficient generation and representation learning, 2025. URL https://arxiv.org/abs/2506.00136.

Yiming Qin, Manuel Madeira, Dorina Thanou, and Pascal Frossard. Defog: Discrete flow matching for graph generation. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/qin25d.html.

Subham S. Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T. Chiu, Alexander Rush, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information*

Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/eb0b13cc515724ab8015bc978fdde0ad-Abstract-Conference.html.

Arne Schneuing, Charles Harris, Yuanqi Du, Kieran Didi, Arian Jamasb, Ilia Igashov, Weitao Du, Carla Gomes, Tom Blundell, Pietro Lio, Max Welling, Michael Bronstein, and Bruno Correia. Structure-based drug design with equivariant diffusion models, 2024. URL https://arxiv.org/abs/2210.13695.

Kartik Sharma, Srijan Kumar, and Rakshit Trivedi. Diffuse, sample, project: Plug-and-play controllable graph generation. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=ia0Z8d1DbY.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=PxTIG12RRHS.

Brian L. Trippe, Jason Yim, Doug Tischer, David Baker, Tamara Broderick, Regina Barzilay, and Tommi S. Jaakkola. Diffusion probabilistic modeling of protein backbones in 3d for the motif-scaffolding problem. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/pdf?id=6TxBxqNME1Y.

Clément Vignac, Igor Krawczuk, Antoine Siraudin, Bohan Wang, Volkan Cevher, and Pascal Frossard. Digress: Discrete denoising diffusion for graph generation. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/pdf?id=UaAD-Nu86WX.

Luhuan Wu, Brian L. Trippe, Christian A. Naesseth, David M. Blei, and John P. Cunningham. Practical and asymptotically exact conditional sampling in diffusion models. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/63e8bc7bbf1cfea36d1d1b6538aecce5-Abstract-Conference.html.

Junjie Xie, Sheng Chen, Jinping Lei, and Yuedong Yang. Diffdec: structure-aware scaffold decoration with an end-to-end diffusion model. *Journal of Chemical Information and Modeling*, 64(7): 2554–2564, 2024.

Dong Xu, Zhangfan Yang, Sisi Yuan, Jenna Xinyi Yao, Jiangqiang Li, and Junkai Ji. Moda: A unified 3d diffusion framework for multi-task target-aware molecular generation, 2025. URL https://arxiv.org/abs/2507.07201.

Haotian Zhang, Huifeng Zhao, Xujun Zhang, Qun Su, Hongyan Du, Chao Shen, Zhe Wang, Dan Li, Peichen Pan, Guangyong Chen, et al. Delete: Deep lead optimization enveloped in protein pocket through unified deleting strategies and a structure-aware network, 2023. URL https://arxiv.org/abs/2308.02172.

Linqi Zhou, Aaron Lou, Samar Khanna, and Stefano Ermon. Denoising diffusion bridge models. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=FKksTayvGo.

Huimin Zhu, Renyi Zhou, Jing Tang, and Min Li. Pgmg: A pharmacophore-guided deep learning approach for bioactive molecular generation, 2022. URL https://arxiv.org/abs/2207.00821.

Huimin Zhu, Renyi Zhou, Dongsheng Cao, Jing Tang, and Min Li. A pharmacophore-guided deep learning approach for bioactive molecular generation. *Nature Communications*, 14(1):6234, 2023.

A PROOFS

Throughout, S is a fixed labelled subgraph with anchored coordinate set $\mathcal{A}$ and free set $\mathcal{F} = \mathcal{C} \setminus \mathcal{A}$, the number of nodes n is fixed, and $\pi_S = p_{\mathrm{data}}(\cdot \mid G[\mathcal{A}] = S)$ is assumed to have non-empty support. We write $z^{(c)}$ for the value of G at coordinate c, and p_t^S for the marginal law at time t of the anchored process, that is

$$p_t^S(G_t) \;=\; \sum_{G_1} \pi_S(G_1)\, p_{t|1}(G_t \mid G_1, S), \tag{12}$$

with $p_{t|1}(\cdot \mid G_1, S)$ as in Equation 9.

A.1 EXACTNESS OF THE ANCHORED PROCESS

Lemma 1 (The objective identifies the conditional posterior)**.** *The functional Equation 11 is minimised over all measurable $\boldsymbol{p}^\theta_{1|t}$ exactly at $\boldsymbol{p}^\theta_{1|t}(\cdot \mid G_t, S) = p_{1|t}(\cdot \mid G_t, S)$, coordinatewise on $\mathcal{F}$, for p_t^S-almost every G_t and almost every t.*

Proof. For fixed t and G_t, the inner expectation of Equation 11 is $-\sum_{c\in\mathcal{F}} \lambda_c \sum_v p_{1|t}(z_1^{(c)} = v \mid G_t, S) \log \boldsymbol{p}^\theta_{1|t}(v \mid G_t, S)$, a sum of cross-entropies between the true coordinatewise posteriors and the predicted ones, with non-negative weights. Each term is minimised uniquely at equality by Gibbs' inequality, and the terms are independent across coordinates because the prediction is coordinatewise. Taking the expectation over $G_t \sim p_t^S$ and t preserves the minimiser on the support. □

Proof of Proposition 1. Two steps.

The anchored coordinates are inert. By Equation 8 the chain starts at $G_0[\mathcal{A}] = S$ with probability one, and by Equation 10 the rate out of any state is zero on every $c \in \mathcal{A}$. A continuous-time Markov chain with zero rate on a coordinate leaves that coordinate unchanged almost surely, so $G_t[\mathcal{A}] = S$ for all $t \in [0, 1]$.

The free coordinates follow an exact discrete flow matching process with data distribution π_S. Condition on $G[\mathcal{A}] = S$ throughout and regard S as a fixed context of the network. Restricted to $\mathcal{F}$, the coupling Equation 8 draws $G_1|_{\mathcal{F}} \sim \pi_S|_{\mathcal{F}}$ and $G_0|_{\mathcal{F}}$ from the product prior; the path Equation 9 is precisely the factorised interpolation Equation 5 on those coordinates; and the generator Equation 10 is the posterior expectation Equation 6 of the per-coordinate rate $R_t^\star$ that generates Equation 4. These are the hypotheses of the discrete flow matching construction of Campbell et al. (2024, Proposition 3.2), whose conclusion is that the Kolmogorov forward equation $\partial_t p_t(G) = \sum_{G'} R_t(G', G)\, p_t(G') - \sum_{G'} R_t(G, G')\, p_t(G)$ holds for the marginal path, here Equation 12. On $\mathcal{A}$ the marginal is constant and the rate is zero, so the equation holds there trivially. Hence the chain has marginals p_t^S for all t; by Lemma 1 the posterior used in Equation 10 is the true one, so no approximation is introduced; and at $t = 1$, $p_1^S = \pi_S$ by Equation 4. □

Remark. Three hypotheses are worth making explicit. First, the statement is a continuous-time, exact-denoiser result, as is Proposition 2; the comparison between them lives in that limit, and neither addresses the error introduced by finite Δt or by factorised per-coordinate updates, which affects both samplers. Second, it fixes the number of nodes n: since Algorithm 2 draws $n \sim \hat{p}(n \mid S)$, the terminal law of the sampler is $\sum_n \hat{p}(n \mid S)\, \pi_S^{(n)}$, which equals π_S only insofar as $\hat{p}$ matches the true conditional. Third, S is assumed to lie in the support of the training anchor distribution, and π_S is the conditional of the *training* distribution; for an anchor absent from the training set this conditional is empty, and what the model does instead is generalise across anchors, which is an empirical claim (Section 4) rather than one covered here.

A.2 WHERE CLAMPING DIFFERS

The clamped sampler runs the unconditional generator Equation 6, driven by the unconditional posterior $p_{1|t}(\cdot \mid G_t)$, and after each step overwrites $\mathcal{A}$ with S. The following isolates what this costs on $\mathcal{F}$, where no overwriting takes place.

For reference, the per-coordinate rate generating Equation 4 is

$$R_t^\star(z_t, j \mid z_1) \;=\; \frac{\mathrm{ReLU}\big[\partial_t p_{t|1}(j \mid z_1) - \partial_t p_{t|1}(z_t \mid z_1)\big]}{Z_t^{>0}\, p_{t|1}(z_t \mid z_1)}, \qquad j \neq z_t,$$

with $Z_t^{>0} = |\{z : p_{t|1}(z \mid z_1) > 0\}|$, the diagonal fixed by $R_t(z_t, z_t) = -\sum_{j \neq z_t} R_t(z_t, j)$, and $\partial_t p_{t|1}(j \mid z_1) = \delta(j, z_1) - p_0(j)$ for the linear path.

Proposition 2. *Let the denoiser be exact for the unconditional process, and consider a state G_t with $G_t[\mathcal{A}] = S$, reached by clamping. The clamped update on $\mathcal{F}$ coincides with the anchored update Equation 10 for every such G_t, every $t \in (0, 1)$ and every $c \in \mathcal{F}$ if and only if*

$$p_{1|t}\Big(z_1^{(c)} \mid G_t\Big) \;=\; p_{1|t}\Big(z_1^{(c)} \mid G_t,\; G_1[\mathcal{A}] = S\Big) \qquad \textit{for all } c \in \mathcal{F}, \tag{13}$$

that is, if and only if the event that the sample satisfies the constraint is uninformative about every free coordinate given the current state.

Proof. Both updates have the form Equation 6 with the same per-coordinate rate $R_t^\star(\cdot \mid z_1^{(c)})$, and differ only in the distribution against which $R_t^\star$ is averaged: the clamped update uses $p_{1|t}(\cdot \mid G_t)$ and the anchored one uses $p_{1|t}(\cdot \mid G_t, G_1[\mathcal{A}] = S)$. Sufficiency of Equation 13 is immediate. For necessity, fix c and assume p_0 has full support, as it does for the marginal prior, so that $Z_t^{>0} = |\mathcal{Z}|$ for every $t < 1$ and every endpoint. Evaluating $R_t^\star$ on the linear path Equation 4 at a state z_t gives, on the off-diagonal $j \neq z_t$,

$$R_t^\star\big(z_t, \cdot \mid z_t\big) \;=\; \mathbf{0}, \qquad R_t^\star\big(z_t, \cdot \mid v\big) \;=\; \kappa_t\big(u + \beta_v e_v\big) \quad (v \neq z_t),$$

where $\kappa_t = \big(|\mathcal{Z}|(1-t)p_0(z_t)\big)^{-1}$, $u_j = \mathrm{ReLU}\big(p_0(z_t) - p_0(j)\big)$ and $\beta_v = 1 + \min\big(p_0(z_t) - p_0(v),\, 0\big) \in (0, 1]$. The clamped and anchored updates average $R_t^\star$ against two *probability* vectors, so their difference d satisfies $\sum_v d_v = 0$. Since the $v = z_t$ term contributes nothing, equality of the averaged rates reads $s\,u + \sum_{v \neq z_t} d_v \beta_v e_v = 0$ with $s = \sum_{v \neq z_t} d_v = -d_{z_t}$. Coordinate $w \neq z_t$ gives $d_w = -s\,u_w/\beta_w$, and summing over w gives $s\big(1 + \sum_{w \neq z_t} u_w/\beta_w\big) = 0$. As $u \geq 0$ and $\beta_w > 0$, the bracket is at least 1, so $s = 0$ and hence $d = 0$. Equality of the averages therefore forces equality of the coordinate marginals. Note that the family $\{R_t^\star(z_t, \cdot \mid v)\}_v$ is *not* linearly independent, since it contains the zero vector at $v = z_t$: a coordinate already at its endpoint carries no rate. The argument uses the simplex constraint on the two posteriors instead.

□

Remark. Condition Equation 13 is an independence requirement between the constraint and the part of the molecule being generated. It holds when S provides no additional information about any free coordinate given the current state, and generally fails whenever conditioning on S changes one of these coordinatewise posteriors. The constrained generation tasks exist because a scaffold does say a great deal about its decorations, which is closely related to the mechanism identified by Trippe et al. (2023), specialised here to the discrete factorised setting.

Remark: the noise model matters. Proposition 2 is a statement about substitutive noise, which is what DiGress and DeFoG use (Vignac et al., 2023; Qin et al., 2025). Under an absorbing kernel a non-mask symbol is its own clean value (Austin et al., 2021; Sahoo et al., 2024), every subset of observed coordinates is a state the forward process produces, and the constrained update is the reverse kernel rather than an intervention on it. This is why masked models over fragment strings condition on substructures natively (Lee et al., 2025).

B Supplementary Results

B.1 De Novo Generation

We provide additional results for de novo generation in Table 4.

B.2 SAFE-Based Fragment-Constrained Generation

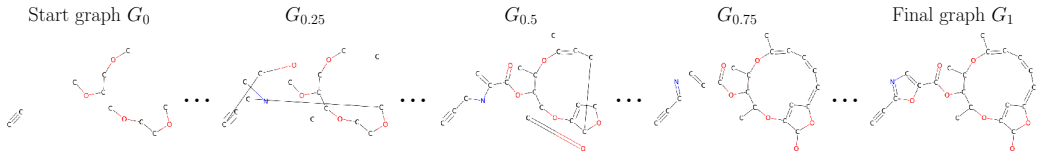
We provide numerical results for SAFE-based fragment-constrained generation in Table 5.

Table 4: De novo molecular generation performance on 1,000 generated samples. For diffusion-based models, $T = 500$ diffusion steps are used.

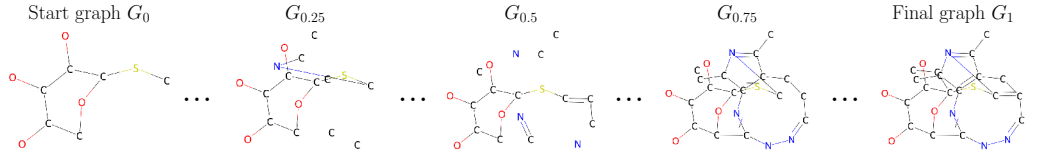
Model	Validity	Uniqueness	Diversity	QED	SA
Pretrained SAFE-GPT	0.933	1.000	0.879	0.691	3.415
trained from scratch	0.442	0.999	0.866	0.800	2.668
Pretrained GenMol	1.000	0.961	0.828	0.842	2.493
trained from scratch	0.106	0.073	0.985	0.360	5.424
DiGress	0.812	1.000	0.860	0.809	2.710
DeFoG	0.914	1.000	0.862	0.818	2.479
GBD	0.857	1.000	0.887	0.764	3.700

B.3 Qualitative Examples

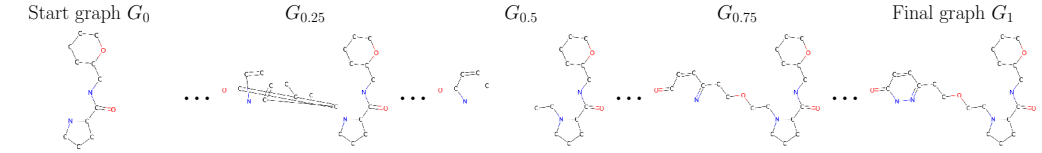
Additional qualitative examples are shown in Figure 5.



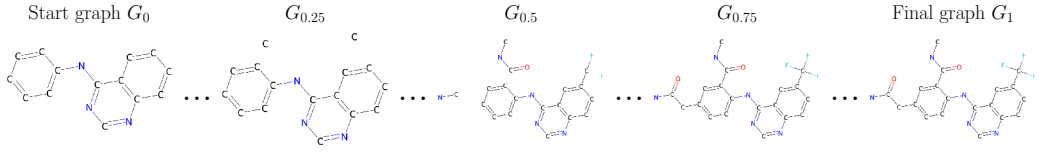
(a) Linker Generation



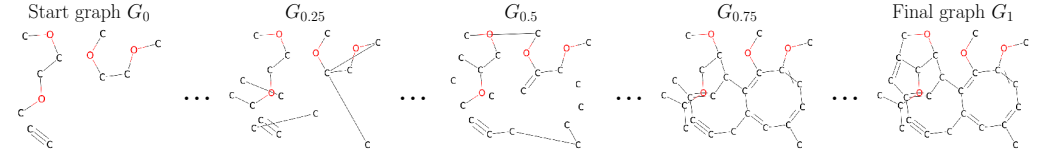
(b) Motif Extension



(c) Scaffold Decoration



(d) Superstructure Generation



(e) Scaffold Morphing

Figure 5: Example generations across the five fragment-constrained tasks in the SAFE benchmark.

Table 5: Performance across fragment-constrained generation SAFE tasks averaged over 5 runs. For DiGress and DeFoG, we use $T = 500$ which is their default setting, while for GBD we use $T = 50$. Higher scores indicate better performance. Best values within each task are shown in bold.

Model	Validity	Uniqueness	Diversity	QED	SA
Linker Generation					
Pretrained SAFE-GPT	**0.785**	0.809	0.572	**0.617**	0.706
trained from scratch	0.297	0.665	0.497	0.584	**0.758**
Pretrained GenMol	0.670	0.669	0.539	0.431	0.679
trained from scratch	0.156	0.061	0.197	0.552	0.748
DiGress	0.113	0.773	0.726	0.606	0.566
DeFoG	0.376	0.927	0.756	0.604	0.636
GBD	0.593	**0.954**	**0.776**	0.564	0.616
Motif Extension					
Pretrained SAFE-GPT	**0.976**	0.721	0.581	0.573	0.736
trained from scratch	0.097	0.494	0.527	0.455	**0.866**
Pretrained GenMol	0.804	0.601	0.650	0.605	0.733
trained from scratch	0.734	0.020	0.218	0.561	0.799
DiGress	0.071	0.811	0.704	**0.611**	0.624
DeFoG	0.352	0.992	0.780	0.603	0.671
GBD	0.618	**1.000**	**0.843**	0.580	0.571
Scaffold Decoration					
Pretrained SAFE-GPT	0.808	0.803	0.554	0.353	0.676
trained from scratch	0.040	0.008	0.132	0.147	**0.973**
Pretrained GenMol	0.786	0.032	0.308	0.667	0.789
trained from scratch	0.783	0.033	0.304	**0.681**	0.803
DiGress	0.107	**0.876**	**0.733**	0.599	0.512
DeFoG	0.335	0.860	0.708	0.613	0.596
GBD	**0.825**	0.840	0.648	0.603	0.681
Superstructure Generation					
Pretrained SAFE-GPT	**0.921**	0.632	0.492	0.497	0.707
trained from scratch	0.534	0.131	0.139	0.509	**0.860**
Pretrained GenMol	0.788	0.034	0.339	0.670	0.779
trained from scratch	0.787	0.034	0.304	**0.680**	0.810
DiGress	0.107	0.847	0.711	0.607	0.530
DeFoG	0.345	**0.864**	**0.714**	0.612	0.605
GBD	0.782	0.842	0.669	0.592	0.674
Scaffold Morphing					
Pretrained SAFE-GPT	**0.704**	0.682	0.536	0.572	0.728
trained from scratch	0.149	0.614	0.437	0.531	**0.808**
Pretrained GenMol	0.482	0.461	0.541	0.493	0.69
trained from scratch	0.156	0.054	0.159	0.571	0.766
DiGress	0.114	0.803	0.707	**0.621**	0.571
DeFoG	0.387	0.921	0.757	0.606	0.631
GBD	0.585	**0.950**	**0.777**	0.572	0.614