LOGICAL GUIDANCE OF STOCHASTIC INTERPOLANTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Guidance and composition of generative models have applications in visual design, chemistry, and biology. Logical guidance composes conditional scores exactly for diffusion models. However, Flow Matching (FM) and Stochastic Interpolants (SI) learn velocity fields. We show that negation and exclusive disjunction admit exact conditional velocity composition on any interpolant, whereas conjunction, and therefore any formula containing one, is exact on Gaussian paths but in general incurs a drift-interaction error, so composed velocities need not sample the logical target. LOGSI solves this by using score-based re-equilibration as our principled mechanism and falling back to the Gaussian-path velocity rule as an approximate transport predictor. In non-Gaussian bridge problems, generation alternates transport steps with score re-equilibration steps. On image transfer (EMNIST to CMNIST, CelebA) and QM9 source-to-target molecular editing with logical attribute constraints, LOGSI improves rule conformity at the same network-evaluation budget over composed-velocity baselines, while also improving fidelity on the image tasks. More broadly, LOGSI applies wherever stochastic interpolants need guidance and logical combinations of constraints are natural.

1 INTRODUCTION

Guidance and composition are central to generative AI, enabling controlled generation across domains ranging from visual design to chemistry and biology (Saharia et al., 2022; Hoogeboom et al., 2022; Watson et al., 2023; Skreta et al., 2025; Blohm & Garg, 2026). Guidance steers a model toward desired properties, while composition combines multiple conditions or concepts to produce outputs that satisfy more complex requirements. Most existing methods target diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song et al., 2021; 2020), where one endpoint is a fixed Gaussian, and the reverse dynamics depend only on the score. A natural extension is to lift logical guidance to stochastic interpolants (SI) (Albergo et al., 2025), which unify diffusion and flow matching (Lipman et al., 2023) and are not tied to a Gaussian endpoint. Classical guidance (Dhariwal & Nichol, 2021; Ho & Salimans, 2022) for SI has been studied (Feng et al., 2025; Jiang et al., 2026), but logical composition of atomic conditions in this regime remains open. We focus on domain transfer (Liu et al., 2023): an interpolant transports a source distribution ρ_0 to a target ρ_1, samples can be initialized from concrete source instances (Albergo et al., 2024), and guidance enforces a Boolean formula φ over target attributes. For example, we may edit a source molecule by adding a geometry (e.g. rings) and an atom-composition constraint (e.g. oxygen).

Throughout, we distinguish between unconditional velocity and score fields (or velocities and scores for short), denoted by $v_t(x)$ and $s_t(x)$, and conditional velocity and score fields, denoted by $v_t(x \mid c)$ and $s_t(x \mid c)$, which steer generation toward samples satisfying a constraint or property c.

The core challenge is that SIs learn conditional velocities, but it is not clear a priori how these fields should be combined to enforce a Boolean formula φ. For conditional scores, Alesiani et al. (2026) (LogDiff) derive exact recursive composition rules. We show that the analogous velocity composition is exact for Gaussian-path bridges, and that outside that regime it fails if the Boolean formula contains conjunctions or conditional-independent disjunctions (CI-disjunctions).

We introduce LOGSI to address this asymmetry between velocity and score compositionality (Figure 1). On general stochastic interpolants with a non-Gaussian starting distribution, both velocity and score are available at sampling time, but only the score admits exact logical composition for every formula (Alesiani et al., 2026). In LOGSI, composed *velocities* are exact on Gaussian-path

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.

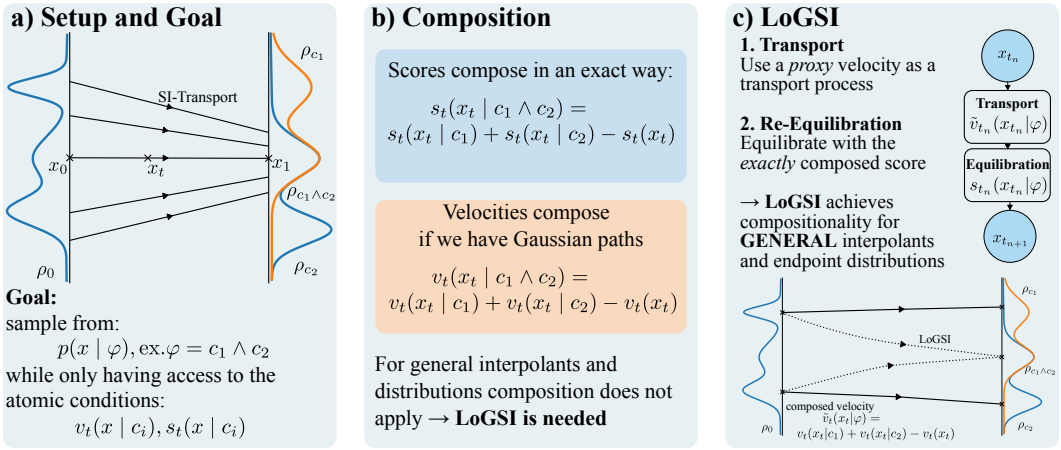

Figure 1: **Overview of LoGSI. (a)** Logical composition targets samples from ρ_1 satisfying $c_1 \wedge c_2$, transported from ρ_0 by a stochastic interpolant, sampling can only use the atomic conditioned scores and velocities. **(b)** Scores compose exactly for general stochastic interpolants, velocity composition matches the same rule on Gaussian-path / diffusion bridges, except at negation and exclusive-disjunction nodes, which match on any interpolant (Theorem 1). **(c)** Our method transports samples with a proxy velocity roughly pointing in the right direction, then re-equilibrates with the exactly composed conditional score.

bridges and can be used as a *proxy* transport field elsewhere. LoGSI therefore separates roles: an approximate transport step along the *proxy* transport field $\tilde{v}_t(\cdot \mid \varphi)$ advances mass along the interpolant, while exact composed conditional scores re-equilibrate samples toward $p_t(\cdot \mid \varphi)$ at each timestep. In the limit of sufficient score-correction steps at fixed t, the procedure is consistent with the logical target. Concretely, we realize this split as a transport–re-equilibration sampler: each outer step applies a transport update along $\tilde{v}_t$, followed by K overdamped Langevin steps with the exactly composed conditional score.

We evaluate our method on image and molecular transfer with non-Gaussian endpoints, testing general Boolean templates (conjunction, disjunction, negation) and randomly generated expression trees. At matched network-evaluation budget, score-guided re-equilibration with a proxy velocity improves rule conformity over composed-velocity integration. It also yields higher fidelity on the image experiments.

Our main contributions are as follows:

- **Score–velocity asymmetry for logical SI.** On Gaussian-path flow matching, LogDiff score composition induces an exact velocity composition rule. For general stochastic interpolants with non-Gaussian endpoints, score composition carries over unchanged, velocity rules fail for conjunction and CI-disjunction, and therefore for general logical statements φ.
- **Logical guidance by score correction.** We sample under Boolean formulas by re-equilibrating with exactly composed conditional scores at each timestep, avoiding any closed-form composed velocity field.
- **Proxy velocity transport.** We reuse the Gaussian-path velocity rule as a transport proxy, cutting the number of correction steps needed and improving conformity over the composed-velocity baseline.

2 Related Work

Generative methods and bridges. Diffusion models generate data by reversing a fixed noising process (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song & Ermon, 2020), unified by Song et al. (2021) in the notion of stochastic processes. Song et al. (2021) also propose to sample with a

predictor-corrector scheme, where a Langevin corrector uses the score to remove the predictor's discretization error while both target the same marginal. Our sampler is similar but structurally different: because only the score composes exactly for formulas containing conjunctions or general disjunctions (Remark 1), the score–velocity asymmetry (Section 4.2) forces the transport step to be a deliberate proxy $\tilde{v}_t \neq v_t(\cdot \mid \varphi)$, while the corrector is the exactness mechanism enforcing the logical constraint, needed beyond discretization error even with infinitesimal transport steps. Flow matching (Lipman et al., 2023) learns a velocity field between two distributions, and stochastic interpolants (Albergo et al., 2025) generalize both diffusion and flow matching while dropping the Gaussian-endpoint assumption, which is why we adopt them for our method. Finally, Schrödinger bridges (De Bortoli et al., 2021; Shi et al., 2023) transport between arbitrary distributions under an entropy-optimal-transport objective. Since our logical composition is orthogonal to how the bridge is trained, extending it to such samplers is an appealing direction for future work.

Guidance and compositional conditioning. Guidance of generative models was first achieved by training a separate classifier to steer diffusion sampling (Dhariwal & Nichol, 2021). The noise-aware classifiers we use for the posterior weights in our logical composition play an analogous role. Classifier-free guidance (Ho & Salimans, 2022) shortly after removed the need for a separately trained classifier by jointly modeling conditional and unconditional scores. Both approaches condition on a single condition and do not address how multiple conditions should be combined. Compositional generation, started in products of experts (Hinton, 2002) and revived more recently (Du et al., 2023; Gaudi et al., 2025), targets exactly this problem, and subsequent work developed compositional and logical conditioning for diffusion models (Liu et al., 2022). Closely related, Skreta et al. (2025) introduce SuperDiff, which composes pretrained diffusion models at inference time through logical `AND` and `OR` operators. We discuss the connection to our bridge setting in Section A.6. Most relevant to us, Alesiani et al. (2026) derive exact recursive rules for composing conditional scores under Boolean formulas. Guidance has also been extended to flow matching (Feng et al., 2025) and stochastic interpolants (Jiang et al., 2026), but to our knowledge logical composition in these settings remains unexplored.

3 Preliminaries

To build our method we rely on the framework of stochastic interpolants by Albergo et al. (2025) and the logical guidance work of Alesiani et al. (2026).

3.1 Stochastic Interpolants

In this work we use the framework of stochastic interpolants by Albergo et al. (2025), which unifies diffusion and flow matching models. The idea of this framework is to replace the diffusion or linear interpolation of diffusion and flow matching models between the two distributions ρ_0 and ρ_1 with an interpolant of the form

$$x_t = I(t, x_0, x_1) + \gamma(t)z, \quad t \in [0,1],\, z \sim \mathcal{N}(0, I), \tag{1}$$

where $(x_0, x_1) \sim \pi$ is a coupling with marginals $x_0 \sim \rho_0$ and $x_1 \sim \rho_1$, the interpolation I satisfies the boundary conditions $I(0, x_0, x_1) = x_0$ and $I(1, x_0, x_1) = x_1$, and the noise scheduler γ satisfies $\gamma(0) = \gamma(1) = 0$ and $\gamma(t) > 0$ for all $t \in (0, 1)$. One can think of this as an interpolation between the distributions ρ_0 and ρ_1, bridged by intermittently adding Gaussian noise to the process. The cases $\gamma \equiv 0$ (flow matching) and Gaussian-path diffusion are special cases (Albergo et al., 2025). We work with general endpoints ρ_0, ρ_1 and refer to Section 4.1 for the Gaussian-path theory used in our velocity composition result. In the bridge problem we regress two targets between the given target distributions: the velocity generated by the interpolation I and the score generated by the noise scheduler γ. The training objectives are

$$\mathcal{L}_v = \mathbb{E}\big\|v_t(x_t) - (\partial_t I(t, x_0, x_1) + \dot{\gamma}(t)z)\big\|^2, \quad \mathcal{L}_s = \mathbb{E}\big\|s_t(x_t) + \tfrac{z}{\gamma(t)}\big\|^2, \tag{2}$$

where the expectations are over t, $(x_0, x_1) \sim \pi$, and $z \sim \mathcal{N}(0, I)$. The minimisers define the marginal velocity and score fields

$$v_t(x) = \mathbb{E}\big[\partial_t I(t, x_0, x_1) + \dot{\gamma}(t)z \mid x_t = x\big], \quad s_t(x) = \nabla_x \log p_t(x), \tag{3}$$

where p_t is the probability density of x_t. For a later derivation we will also need that, by Tweedie's identity, the Gaussian noise component implies that $\mathbb{E}[z \mid x_t = x] = -\gamma(t)\, s_t(x)$. With these two fields trained, we can sample the stochastic interpolant deterministically (ODE) or stochastically (SDE). For ODE-based sampling only the velocity is needed:

$$\mathrm{d}x_t = v_t(x_t)\mathrm{d}t, \quad x_0 \sim \rho_0. \tag{4}$$

For SDE-based sampling the score is also needed explicitly to control the amount of injected noise:

$$\mathrm{d}x_t = \Big(v_t(x_t) + \tfrac{1}{2}\bar{\gamma}(t)s_t(x_t)\Big)\mathrm{d}t + \sqrt{\bar{\gamma}(t)}\,\mathrm{d}W_t, \quad x_0 \sim \rho_0, \tag{5}$$

where $\bar{\gamma}(t) \geq 0$ is a sampling noise scheduler, which is distinct from the training scheduler γ. Empirically choosing them to be equal $\gamma = \bar{\gamma}$, works well. All SDEs, regardless of the noise scheduler, share the same marginal p_t as the probability-flow ODE.

3.2 Exact compositions of scores

For diffusion models, exact logical composition was developed by Alesiani et al. (2026). There only the score enters the reverse-time dynamics, so it is natural to work with posterior logical scores rather than with conditional densities directly. Let $\mathcal{C} = \{c_1, \ldots, c_n\}$ be a set of atomic predicates (e.g. atomic conditions), and let φ be a Boolean formula over $\mathcal{C}$ built from conjunction, disjunction, and negation. For $t \in (0, 1]$ and $x \in \mathbb{R}^d$, define the backward truth probability and the logical score

$$p_t(\varphi \mid x) := \mathbb{P}\big(x_1 \models \varphi \mid x_t = x\big), \quad s_t(\varphi \mid x) := \nabla_x \log p_t(\varphi \mid x), \tag{6}$$

where $x_1 \models \varphi$ means that the terminal state satisfies φ. By Bayes' rule, the conditional score, used for generation, satisfies $s_t(x \mid \varphi) := \nabla_x \log p_t(x \mid \varphi) = s_t(x) + s_t(\varphi \mid x)$. For an atomic predicate c_i, the corresponding logical score can be obtained from a conditional and an unconditional score network as $s_t(c_i \mid x) = s_t(x \mid c_i) - s_t(x)$. Exact composition is possible whenever φ admits a probabilistic-circuit representation whose conjunction nodes combine conditionally independent subformulas and whose disjunction nodes combine conditionally independent or mutually exclusive subformulas (Alesiani et al., 2026). The exact recursive formulas are given in Appendix A.1. Under these conditions, the composed conditional score is an affine combination of the atomic conditional scores: letting k denote the number of atoms $c_1, \ldots, c_k$ appearing in φ, there exist coefficients $\alpha_i^{\varphi}(t, x)$, determined recursively from the posterior estimates $\widehat{\pi}(\cdot) := p_t(\cdot \mid x)$, such that

$$s_t(x \mid \varphi) = s_t(x) + \sum_{i=1}^{k} \alpha_i^{\varphi}(t, x)\big[s_t(x \mid c_i) - s_t(x)\big]. \tag{7}$$

In practice the atomic posteriors $\widehat{\pi}(c_i) = p_t(c_i \mid x)$ are estimated with a noise- and time-aware classifier, while the score terms come from the trained conditional and unconditional networks.

4 Method

We first characterize when logical composition extends from scores to velocities (Section 4.1), identify exactly where this extension fails for general stochastic interpolants (Section 4.2), and then describe our sampling method (Sections 4.3 and 4.4) which resolves the problems.

4.1 Gaussian-path logical composition

We generalize LogDiff-style guidance (Alesiani et al., 2026) to Gaussian-path flow matching, where the source distribution ρ_0 is Gaussian, and show that velocity composition in the sense of LogDiff is then exact (Proposition 1). Throughout, we take $x_t = \alpha_t x_1 + \beta_t \epsilon$, $\epsilon \sim \mathcal{N}(0, I)$. On such a path the conditional velocity is an affine function of the conditional score, $v_t(x \mid c) = a_t x + b_t\, s_t(x \mid c)$, with coefficients (a_t, b_t) that depend on t but not on the conditioning. This is true for every terminal condition c determined by x_1 whose noising kernel is independent of c given x_1. This relation is standard, see e.g. Zheng et al. (2023). We restate it with the explicit coefficients as Lemma 1 in Appendix A.2. Combining it with the compositionality of the scores gives the logical composition rule for velocities. (Equation (7))

Proposition 1 (Exact logical velocity composition on Gaussian paths)**.** *Let φ be a Boolean formula whose score satisfies the linear combination form of Section 3.2 and α_i^φ the coefficients of LogDiff. On the Gaussian path above,*

$$v_t(x \mid \varphi) = v_t(x) + \sum_{i=1}^{k} \alpha_i^\varphi(t,x)\big[v_t(x \mid c_i) - v_t(x)\big]. \tag{8}$$

This is a sufficiency result: because (a_t, b_t) are independent of the conditioning, the composed score of Section 3.2 induces the matching composed velocity, whereas necessity of the affine relation is not established. Proofs are in Section A.2 and the recursive formulas for the $\alpha_i^\varphi(t,x)$ in Section A.1. For general stochastic interpolants this rule becomes invalid in general, while negation and exclusive disjunction remain exact, as we show next.

4.2 Velocity composition on general stochastic interpolants

We now drop the Gaussian-path assumption and return to the general stochastic-interpolant framework of Section 3.1: $(x_0, x_1) \sim \pi$ with $x_0 \sim \rho_0$, $x_1 \sim \rho_1$ (both possibly non-Gaussian), and

$$x_t = I(t, x_0, x_1) + \gamma(t)z, \qquad z \sim \mathcal{N}(0, I), \tag{9}$$

with the same boundary conditions on I and γ as in the preliminaries. As in Section 4.1, we seek a relation between the conditional velocity and score. For a general stochastic interpolant, the marginal velocity admits the decomposition

$$v_t(x \mid c) = \mu_t(x \mid c) - \dot\gamma(t)\gamma(t)s_t(x \mid c), \qquad \mu_t(x \mid c) = \mathbb{E}\big[\partial_t I(t, x_0, x_1) \mid x_t = x, c\big], \tag{10}$$

which separates a conditioning-dependent drift $\mu_t(x \mid c)$ from a score term whose coefficient depends on t only. This identity is established in the theoretical analysis of Albergo et al. (2025). It is then natural to test whether LogDiff composition, exact for scores, remains exact when applied directly to velocities. We define $\check v_t(x \mid \varphi)$ and $\check\mu_t(x \mid \varphi)$ as

$$\check\phi_t(x \mid \varphi) := \phi_t(x) + \sum_{i=1}^{k} \alpha_i^\varphi(t,x)\ \left[\phi_t(x \mid c_i) - \phi_t(x)\right], \quad \phi_t \in \{v_t, \mu_t\}. \tag{11}$$

Since the score term in Equation (10) composes exactly, i.e. $\check s_t(x \mid \varphi) = s_t(x \mid \varphi), \forall\varphi$ (under the assumptions given in Section 3.2), the question of when Equation (11) is exact reduces entirely to the drift: the two fields differ exactly by $\check\mu_t(x \mid \varphi) - \mu_t(x \mid \varphi)$. By writing $\mathbf{1}_\psi := \mathbf{1}\{x_1 \models \psi\}$, we define the logical drift moment of a subformula ψ as

$$m_t^\psi(x) := \mathbb{E}\big[\partial_t I(t, x_0, x_1)\, \mathbf{1}_\psi \;\big|\; x_t = x\big], \qquad \text{so that} \qquad \mu_t(x \mid \psi) = \frac{m_t^\psi(x)}{\widehat\pi(\psi)}. \tag{12}$$

Both m_t^ψ and $\widehat\pi(\psi)$ are linear in the indicator $\mathbf{1}_\psi$, so every Boolean operation is inherited exactly except conjunction (and therefore also for CI-disjunction, since $c_1 \vee c_2 = \neg(\neg c_1 \wedge \neg c_2)$), where $m_t^{\psi\wedge\chi} = \mathbb{E}[\partial_t I\, \mathbf{1}_\psi \mathbf{1}_\chi \mid x_t = x]$ weights the drift by a product of indicators that does not reduce to the atomic moments (see Lemma 2 in the Appendix). Two consequences follow, one positive (Theorem 1) and one negative (Theorem 2).

Theorem 1 (Exact velocity composition for negation and mutual exclusive disjunction)**.** *For any interpolant I, noise schedule γ, coupling π and endpoints ρ_0, ρ_1, the NOT rule (whenever $\widehat\pi(\psi) \in (0,1)$) and the OR-ME rule (whenever $\mathbf{1}_\psi \mathbf{1}_\chi \equiv 0$) of Section A.1 are exact for velocities. Consequently, if φ admits a probabilistic-circuit representation containing no conjunction or CI-disjunction node, then $\check v_t(\cdot \mid \varphi) = v_t(\cdot \mid \varphi)$ on every stochastic interpolant.*

For conjunction the obstruction can be computed in closed form, and it turns out to be an interaction between the atomic conditions.

Theorem 2 (Exact obstruction for conjunction)**.** *Let $\varphi = c_1 \wedge c_2$, assume the atoms are conditionally independent given x_t in a neighbourhood of x with $\widehat\pi(c_i) \in (0,1)$, so that LogDiff's AND rule ($\alpha_1^\varphi = \alpha_2^\varphi = 1$) gives the exact composed score and $\widehat\pi(\varphi) = \widehat\pi(c_1)\widehat\pi(c_2) > 0$, and let $d_t(a_1, a_2) :=$*

$\mu_t(x \mid c_1^{a_1} \wedge c_2^{a_2})$, $a \in \{0,1\}^2$, be the four cell-conditional drifts, where $c^1 = c$ and $c^0 = \neg c$. Then, for every interpolant I, noise schedule γ, coupling π and endpoints ρ_0, ρ_1,

$$\check{v}_t(x \mid \varphi) - v_t(x \mid \varphi) = -\big(1 - \widehat{\pi}(c_1)\big)\big(1 - \widehat{\pi}(c_2)\big)\,\Delta_t(x), \tag{13}$$

$$\Delta_t(x) := d_t(1,1) - d_t(1,0) - d_t(0,1) + d_t(0,0). \tag{14}$$

The bias thus factorises into a purely probabilistic prefactor and the drift interaction Δ_t. Composition at a conjunction is therefore exact if and only if $\big(1-\widehat{\pi}(c_1)\big)\big(1-\widehat{\pi}(c_2)\big)\Delta_t(x) = 0$, in particular whenever the conditional drift is additive $d_t(a_1, a_2) = \theta_0 + \theta_1 a_1 + \theta_2 a_2$ in the truth values of the atoms, $\Delta_t(x) = 0$, and asymptotically as $\widehat{\pi}(c_i) \to 1$.

The same computation applies at any conjunction node with conditionally independent children, so each such node contributes a residual of the form Equation (14). Composed velocities are exact only if all of these vanish or cancel, which does not hold in general, as the counterexample in Section A.3 shows. The same applies to CI-disjunction through $\psi \vee \xi = \neg(\neg\psi \wedge \neg\xi)$. Proofs are in Appendix A.3. Remark 2 shows that Gaussian-path structure forces $\Delta_t \equiv 0$. Since a logical statement containing conjunctions cannot be used to combine velocities, we need a sampler that still enforces conformity, even when the velocity composition is not exact.

Remark 1 (Scores still compose exactly)**.** *The obstruction in Theorem 2 is specific to general composed velocities. The composed conditional score $s_t(x \mid \varphi)$ from Section 3.2 remains exact on general stochastic interpolants under the same conditional-independence and exclusivity assumptions as in LogDiff.*

4.3 Guiding general stochastic interpolants

In the general bridge problem, the probability-flow ODE and its equivalent SDE family share the marginals p_t, but only the score can be composed exactly for general logical statements φ (Remark 1). We therefore split sampling into two roles: (i) a *transport* step advances mass toward ρ_1, and (ii) exact composed scores *re-equilibrate* each sample to the correct conditional marginal $p_t(\cdot \mid \varphi)$.

Transport-re-equilibration. On a time grid $0 = t_0 < \cdots < t_N = 1$ with step $\Delta t_n = t_{n+1} - t_n$, each outer iteration first takes a single transport step along $\tilde{v}_t(\cdot \mid \varphi)$, using either the ODE or SDE transport (Line 9 and 11 of Algorithm 1, a hyperparameter of our method). It then runs $K \geq 0$ overdamped Langevin correctors at fixed $t = t_{n+1}$ after transport,

$$x_{n+1}^{(k+1)} = x_{n+1}^{(k)} + \tfrac{1}{2}\bar{\gamma}(t_{n+1})\, s_{t_{n+1}}(x_{n+1}^{(k)} \mid \varphi)\,\Delta t_n + \sqrt{\bar{\gamma}(t_{n+1})\,\Delta t_n}\,\eta_n^{(k)}, \tag{15}$$

initialized at the transported state, with $\eta_n^{(k)} \sim \mathcal{N}(0, I)$ independent, for $k = 0, \ldots, K-1$. As in diffusion sampling, correctors are not applied at the endpoints, where the score is undefined. We have picked $\frac{1}{2}\gamma(t_{n+1})\Delta t_n$ as Langevin step size to not introduce a new hyperparameter. In principle one can choose other step sizes, and we discuss this in Appendix B.5. The conditional score is assembled from the atomic conditional scores via Section 3.2. For an algorithmic depiction, see Algorithm 1, Algorithm 2 and Algorithm 3 in the appendix. For an discussion about the consistency of this equilibration see Section A.4.

4.4 Proxy velocity for transport

A natural, easy to compute transport step uses the unconditional velocity, $\tilde{v}_t(x \mid \varphi) := v_t(x)$. However, this step advances the sample along trajectories that are typically misaligned with $p_t(\cdot \mid \varphi)$, therefore requiring many Langevin equilibration steps to re-equilibrate with the logical marginal at each noise level. We instead propose to use the Gaussian-path composition rule Equation (11) as a *proxy* velocity $\tilde{v}_t(x \mid \varphi) = \check{v}_t(x \mid \varphi)$, evaluated with the atomic conditional velocities of the trained stochastic interpolant. Outside the Gaussian-path regime this field is not equal to the true conditional velocity $v_t(x \mid \varphi)$ in general (Theorem 2), but it is the unique velocity that would be correct if the affine relation of Lemma 1 held. Additionally, by Theorem 1 it is exact at every negation and exclusive-disjunction node of φ, which is what makes Equation (11) a good transport field despite not being the conditional velocity in general. Empirically it reduces the burden on the corrector as shown in the experiment section.

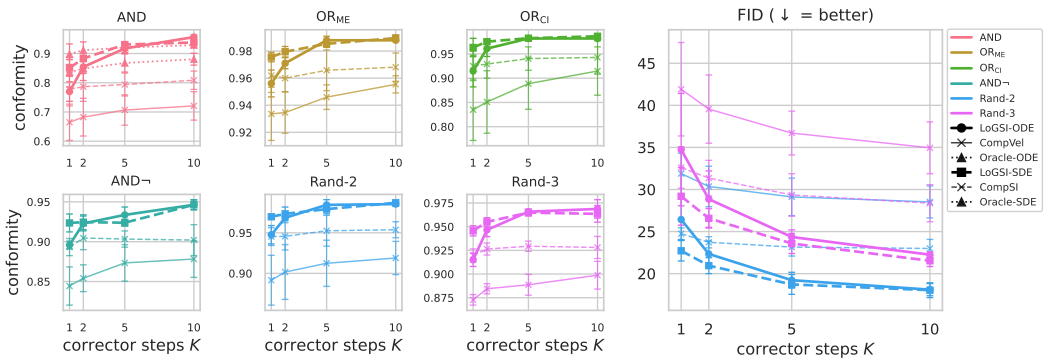

Figure 2: **EMNIST to CMNIST** logical sampling at matched NFE. **(left)** Conformity of LoGSI and of the compute-matched baselines, as a function of corrector steps K. Rand-2/Rand-3 are random depth-2/3 expressions. **(right)** FID as a function of corrector steps K at matched NFE. For overview reasons only Rand-2/Rand-3 are shown, for all of the experiments see Figure 8.

5 Experiments

We evaluate LoGSI on three domain-transfer benchmarks that test logical composition under non-Gaussian endpoints. In each setting we train a stochastic interpolant transporting a *source* distribution ρ_0 to a *target* distribution ρ_1, then sample from ρ_1 subject to a Boolean constraint φ over semantic attributes. Atoms are class-conditional velocity and score networks, plus noise-aware classifiers for the posterior weights in disjunctions and negations (Section B). All results are averaged over 3 training and evaluation runs.

Baselines and compute matching. We compare our method against two composition-only baselines without score re-equilibration ($K{=}0$): **CompVel**, deterministic ODE integration of the composed proxy velocity in Equation (11) (the analogue of LogDiff-style guidance (Alesiani et al., 2026)), and **CompSI**, SDE integration that combines that velocity with the exactly composed score. Our sampler is run as **LoGSI-ODE** and **LoGSI-SDE**, using the ODE or SDE predictor of Algorithm 1 with $K{>}0$ Langevin correctors at fixed t. For the logical `AND` we additionally report **Oracle-ODE/SDE**: compute-matched CFG on the joint label (guidance weight 1), without composition or score correction. All main-text comparisons are *compute-matched* in number of function evaluations (NFE): each LoGSI run with N outer steps and K correctors is matched against the corresponding $K{=}0$ baseline with enough extra integration steps to equalise the budget (LoGSI-ODE against CompVel, LoGSI-SDE against CompSI), which isolates score-based re-equilibration from simply spending more network calls on transport.

Metrics. Our primary metric is logical *conformity*: the fraction of generated samples whose attributes satisfy φ. On the image benchmarks attributes are read off by a held-out evaluator classifier, distinct from the noise-aware classifier that drives sampling. On QM9 they are recomputed from the generated structure with RDKit (RDKit contributors), along with chemical validity, and conformity is measured only over valid molecules. For images we additionally report FID against an expression-conditioned reference set (real images whose ground-truth attributes satisfy the same φ) and LPIPS to the source image. For QM9 we report validity$\times$conformity, source-to-sample Kabsch root mean square deviation (RMSD), and novelty. Full definitions are in Section B.4.

EMNIST to CMNIST transfer. We train paired ResNet (He et al., 2016) velocity and score models transporting greyscale EMNIST (Cohen et al., 2017) letters (`a--e`, `A--E`) to Colored MNIST digits (LeCun et al., 1998; Gaudi et al., 2025), so each source letter is edited into a digit satisfying the logical rule. Atoms are `Digit`(d) and `Color`(c) for $d, c \in \{0, \ldots, 9\}$. We evaluate fixed templates (AND, OR_{ME}, OR_{CI}, AND$\neg$) and randomly generated expression trees of depth two and three over the same atoms. Section B.2 states which disjunction rule applies to which formula. Figure 2 summarizes the compute-matched comparison. Adding correctors improves conformity by up to 14 pp over CompSI in the AND case and up to 4 pp on random depth-three expressions, where the

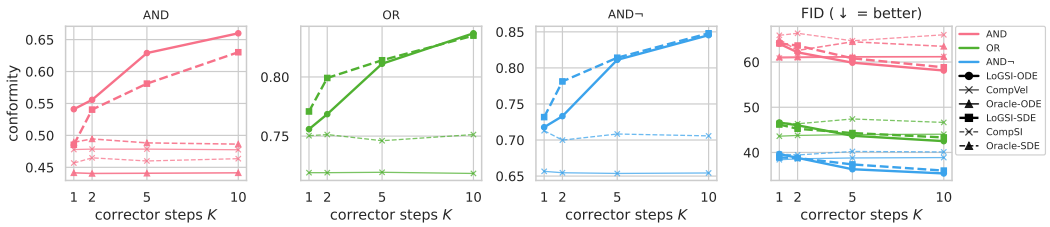

Figure 3: **CelebA latent transfer** with logical constraints on `Hat` and `Glasses`. **(left)** Conformity of LoGSI and the compute-matched baselines as a function of corrector steps K. **(right)** FID as a function of corrector steps K at matched NFE. Colours index the logical template (AND, OR, AND¬). For readability, the plots with the error bars are moved to the appendix in Figure 7.

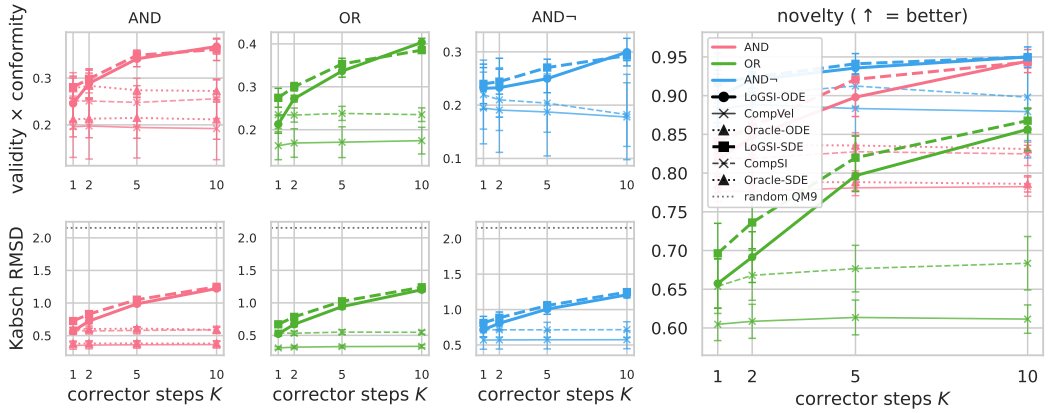

Figure 4: **QM9 molecular editing** with logical constraints on `rings` and `oxygen`, at matched NFE. **(left, top)** validity×conformity and **(left, bottom)** Kabsch-aligned RMSD to the source molecule, both as functions of corrector steps K. The dotted line marks the RMSD between random QM9 molecules. **(right)** Novelty of generated molecules vs. K.

smaller gain may be caused by the already high conformity $> 96\%$. FID also decreases with more correctors, falling below the baselines, so re-equilibration improves fidelity as well as conformity rather than trading one for the other. OR_{ME} still benefits from correctors because Theorem 1 is exact only for the true fields, whereas learned velocities, classifier posteriors, and discretization remain approximate. ODE and SDE transport are indistinguishable at larger K. Oracle-SDE matches the baselines while being stronger at small K, and Oracle-ODE is worse throughout. Further metrics are in Figure 8: LPIPS to the source grows with K but stays below the LPIPS of random source-target pairs, so edits remain close to their source.

CelebA transfer. Inspired by Shi et al. (2023), we transport CelebA (Liu et al., 2015) *female* images (ρ_0) to *male* images (ρ_1), training two SiT models (Ma et al., 2024) for velocity and score in the FLUX.1-schnell VAE (Black Forest Labs, 2024) latent space. Logical atoms are the two attributes `Hat` and `Glasses`, and we test the statements AND, OR_{CI}, and AND¬. Posterior weights for OR_{CI} and negation come from a time-conditioned latent SiT classifier (Section B). Figure 3 reports the compute-matched results. Conformity again rises with correctors, by up to 18 pp over the baseline. Similarly to CMNIST, FID also decreases past the baseline values, and the ODE and SDE predictors perform comparably. Additionally, both Oracle baselines perform worse. As on CMNIST, LPIPS to the source image grows with K but stays below the random source–target LPIPS (Figure 7), so edited faces remain close to their source seed.

QM9 molecular editing. On QM9 (Ramakrishnan et al., 2014; Ruddigkeit et al., 2012) we train EGNN (Satorras et al., 2021) (as in Hoogeboom et al. (2022)) velocity and score models that bridge *source* to *target* molecular graphs, both drawn from QM9. Unlike mirror interpolants that connect

Table 1: **Transport-velocity ablation.** Conformity (validity×conformity for QM9) if the transport step uses the Gaussian-path proxy velocity (LoGSI) vs. the unconditional velocity (uncd.), at 5 corrector steps. Furthermore, we report also a baseline where no transport step was used. We report the AND case only, to avoid confounding from the noise-aware classifier used in OR/negation.

Method	CMNIST	CelebA	QM9
LoGSI 5 step	$\mathbf{0.904 \pm 0.003}$	$\mathbf{0.63 \pm 0.09}$	$\mathbf{0.34 \pm 0.02}$
uncd. 5 step	0.150 ± 0.004	0.07 ± 0.03	0.27 ± 0.02
$\frac{1}{2}v_t(x\|c_1) + \frac{1}{2}v_t(x\|c_2)$ 5 step	0.650 ± 0.007	0.29 ± 0.08	0.31 ± 0.02
only score 5 step	0.034 ± 0.001	0.030 ± 0.002	0.053 ± 0.007

a sample to itself, we use a random coupling so that any molecule can be transported to any other, enabling editing. Atoms are binary molecular attributes. Our main statement pairs `has_rings`$= 1$ with `has_oxygen`$= 1$ under AND, OR_{CI}, and $\text{AND}^{\neg}$. Each sample is initialized from a concrete source graph whose labels *violate* φ, so conformity measures whether the sampler can *create* the constraint from a non-conforming start (Section B). Figure 4 reports the compute-matched results, a stricter test than the image benchmarks since a sample must be chemically valid *and* rule-conforming. Validity×conformity rises with the corrector budget by up to 16 pp over the baseline. Decomposing the product (Figure 9) reveals a trade-off: conformity among valid molecules rises while validity loses. Since exact re-equilibration targets $p_t(\cdot \mid \varphi)$, whose samples are valid by construction, we read this loss as finite-K approximation error rather than as a property of the guidance target. As expected, the Kabsch RMSD to the source grows with K, reaching around 1.4 Å, but stays well below the RMSD between random QM9 molecules (dotted line), so edits remain local. Novelty likewise grows with K. As before, ODE and SDE transport differ little at large K, and both Oracle baselines perform worse than the proposed method.

Transport-velocity ablation. It is a priori unclear which transport velocity $\tilde{v}_t$ is best, so we compare our Gaussian-path proxy velocity Equation (11) against the unconditional velocity $\tilde{v}_t = v_t$, against no transport step at all, and against the equal-weight average $\frac{1}{2}v_t(x \mid c_1) + \frac{1}{2}v_t(x \mid c_2)$, i.e. Liu et al. (2022)'s weighted score composition applied to velocities instead of scores (Tables 1 and 4). The proxy velocity yields higher conformity across all three benchmarks, confirming the design choice of Section 4.4. We report ODE transport here, and the SDE variant (Table 5) shows the same trend.

6 Conclusion

We studied logical composition for stochastic interpolants that transport between non-Gaussian source and target distributions. While conditional scores compose exactly in the LogDiff sense on general interpolants, the analogous velocity rule survives only in part: negation and exclusive disjunction remain exact on any interpolant, whereas conjunction, and hence any formula containing one, is exact only on Gaussian-path bridges. LoGSI therefore separates roles: a proxy composed velocity advances samples along the interpolant, and exactly composed conditional scores re-equilibrate them toward $p_t(\cdot \mid \varphi)$ at each timestep. Across EMNIST-to-CMNIST, CelebA, and QM9 editing benchmarks, this transport–re-equilibration procedure improves logical conformity at matched network-evaluation budget over composed-velocity baselines, for both fixed Boolean templates and randomly generated formulas. It additionally improves fidelity (FID) on the image benchmarks and novelty on QM9. More broadly, the same score-based viewpoint may extend to other bridge learners and to settings where logical combinations of constraints arise together with non-Gaussian transport.

Limitations. Like prior logical score composition, our rules assume conditional independence (or mutual exclusivity) of the composed subformulas given x_t, which is generally approximate but enables tractable exact composition and was adequate in our experiments. Conformity is further limited by the finite corrector budget K and by the calibration of the noise-aware classifiers used for disjunction and negation. Finally, re-equilibration trades off conformity against fidelity to the source sample x_0: as K grows, samples approach $p(\cdot \mid \varphi)$ rather than $p(\cdot \mid \varphi, x_0)$, though moderate K already improves conformity while retaining substantial source information.

REPRODUCIBILITY STATEMENT

All training hyperparameters, architectures, and evaluation protocols are reported in Section B. An anonymized implementation of LOGSI is included in the supplementary material for reviewers. Upon acceptance, we will publicly release the full source code, configuration files, and scripts needed to reproduce the experiments in this paper.

AI USE STATEMENT

In this work, we used generative AI tools to help formalize mathematical claims, assist in the writing of proofs, and implement methods. Concretely, large language models were used to help formalize and prove Theorems 1 and 2 and the explicit counterexample in Section A.3. Coding agents were used to help implement LOGSI and the experimental pipelines. We have not used generative AI tools to generate synthetic data sets, propose or refine hypotheses, interpret results, or assist with translation, and cleaning or reformatting datasets as well as qualitative and thematic data analysis are not applicable to this work. We have not used generative AI tools to generate synthetic data sets, propose or refine hypotheses, interpret results, translation, and cleaning or reformatting datasets. Qualitative or thematic data analysis are not applicable to this work. Additionally, we used generative AI tools to draft parts of the manuscript and edit the text for readability. We have reviewed all AI-assisted work: the statements and proofs of Theorems 1 and 2 and the counterexample were checked by the authors, and LLM-generated code was verified for correctness against the stated algorithms and experimental protocol. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Michael Albergo, Nicholas M Boffi, and Eric Vanden-Eijnden. Stochastic interpolants: A unifying framework for flows and diffusions. *Journal of Machine Learning Research*, 26(209):1–80, 2025.

Michael Samuel Albergo, Mark Goldstein, Nicholas Matthew Boffi, Rajesh Ranganath, and Eric Vanden-Eijnden. Stochastic interpolants with data-dependent couplings. In *International Conference on Machine Learning*, pp. 921–937. PMLR, 2024.

Francesco Alesiani, Jonathan H Warrell, Tanja Bien, Henrik Christiansen, Matheus Ferraz, and Mathias Niepert. Logical guidance for the exact composition of diffusion models. In *Forty-third International Conference on Machine Learning*, 2026.

Black Forest Labs. Flux.1 [schnell]. `https://github.com/black-forest-labs/flux`, 2024.

Peter Blohm and Vikas K Garg. Composition of pretrained diffusion models: A logic-based calculus. In *The Fourteenth International Conference on Learning Representations*, 2026. URL `https://openreview.net/forum?id=ADLiUSC7Qm`.

Gregory Cohen, Saeed Afshar, Jonathan Tapson, and Andre Van Schaik. Emnist: Extending mnist to handwritten letters. In *2017 international joint conference on neural networks (IJCNN)*, pp. 2921–2926. IEEE, 2017.

Valentin De Bortoli, James Thornton, Jeremy Heng, and Arnaud Doucet. Diffusion schrödinger bridge with applications to score-based generative modeling. *Advances in neural information processing systems*, 34:17695–17709, 2021.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

Yilun Du, Conor Durkan, Robin Strudel, Joshua B Tenenbaum, Sander Dieleman, Rob Fergus, Jascha Sohl-Dickstein, Arnaud Doucet, and Will Sussman Grathwohl. Reduce, reuse, recycle: Compositional generation with energy-based diffusion models and mcmc. In *International conference on machine learning*, pp. 8489–8510. PMLR, 2023.

Ruiqi Feng, Chenglei Yu, Wenhao Deng, Peiyan Hu, and Tailin Wu. On the guidance of flow matching. In *International Conference on Machine Learning*, pp. 16993–17029. PMLR, 2025.

Sachit Gaudi, Gautam Sreekumar, and Vishnu Boddeti. Coind: Enabling logical compositions in diffusion models. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Representation Learning*, volume 2025, pp. 32159–32191, 2025.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Geoffrey E Hinton. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800, 2002.

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

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Emiel Hoogeboom, Vıctor Garcia Satorras, Clément Vignac, and Max Welling. Equivariant diffusion for molecule generation in 3d. In *International conference on machine learning*, pp. 8867–8887. PMLR, 2022.

Puming Jiang, Meiyi Wang, Kelvin Lin, Ce Hao, and Harold Soh. Guided streaming stochastic interpolant policy. *arXiv preprint arXiv:2605.10051*, 2026.

Wolfgang Kabsch. A solution for the best rotation to relate two sets of vectors. *Foundations of Crystallography*, 32(5):922–923, 1976.

Yann LeCun, Corinna Cortes, and Christopher J.C. Burges. The MNIST database of handwritten digits, 1998. URL http://yann.lecun.com/exdb/mnist/.

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.

Guan-Horng Liu, Arash Vahdat, De-An Huang, Evangelos Theodorou, Weili Nie, and Anima Anandkumar. I2sb: Image-to-image schrödinger bridge. In *International Conference on Machine Learning*, pp. 22042–22062. PMLR, 2023.

Nan Liu, Shuang Li, Yilun Du, Antonio Torralba, and Joshua B Tenenbaum. Compositional visual generation with composable diffusion models. In *European conference on computer vision*, pp. 423–439. Springer, 2022.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pp. 3730–3738, 2015.

Nanye Ma, Mark Goldstein, Michael S Albergo, Nicholas M Boffi, Eric Vanden-Eijnden, and Saining Xie. Sit: Exploring flow and diffusion-based generative models with scalable interpolant transformers. In *European Conference on Computer Vision*, pp. 23–40. Springer, 2024.

Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

Raghunathan Ramakrishnan, Pavlo O Dral, Matthias Rupp, and O Anatole von Lilienfeld. Quantum chemistry structures and properties of 134 kilo molecules. *Scientific Data*, 1, 2014.

RDKit contributors. RDKit: Open-source cheminformatics. https://www.rdkit.org.

Lars Ruddigkeit, Ruud Van Deursen, Lorenz C Blum, and Jean-Louis Reymond. Enumeration of 166 billion organic small molecules in the chemical universe database gdb-17. *Journal of chemical information and modeling*, 52(11):2864–2875, 2012.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. In *Advances in Neural Information Processing Systems*, 2022.

Vıctor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E (n) equivariant graph neural networks. In *International conference on machine learning*, pp. 9323–9332. PMLR, 2021.

Yuyang Shi, Valentin De Bortoli, Andrew Campbell, and Arnaud Doucet. Diffusion schrödinger bridge matching. *Advances in neural information processing systems*, 36:62183–62223, 2023.

Marta Skreta, Lazar Atanackovic, Joey Bose, Alexander Tong, and Kirill Neklyudov. The superposition of diffusion models using the itô density estimator. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=2o58Mbqkd2.

Jascha Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics, 2015. URL https://arxiv.org/abs/1503.03585.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution, 2020. URL https://arxiv.org/abs/1907.05600.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021.

Joseph L Watson, David Juergens, Nathaniel R Bennett, Brian L Trippe, Jason Yim, Helen E Eisenach, Woody Ahern, Andrew J Borst, Robert J Ragotte, Lukas F Milles, et al. De novo design of protein structure and function with rfdiffusion. *Nature*, 620(7976):1089–1100, 2023.

Qinqing Zheng, Matt Le, Neta Shaul, Yaron Lipman, Aditya Grover, and Ricky TQ Chen. Guided flows for generative modeling and decision making. *arXiv preprint arXiv:2311.13443*, 2023.

A METHOD DETAILS

A.1 RECURSIVE COMPOSITION RULES

Scores. As stated in the main text, exact composition is possible whenever φ admits a probabilistic-circuit representation whose conjunction nodes combine conditionally independent subformulas and whose disjunction nodes combine either conditionally independent or mutually exclusive subformulas (Alesiani et al., 2026). In that case, $p_t(\varphi \mid x)$ and $s_t(\varphi \mid x)$ can be computed recursively from the atomic quantities. Writing $\widehat{\pi}(\psi) = p_t(\psi \mid x)$ for posterior estimates of subformulas ψ, the main rules are

$$s_t(c_i \mid x) = \nabla_x \log p_t(x \mid c_i) - \nabla_x \log p_t(x), \qquad \text{(Atomic)} \qquad (16)$$

$$s_t(\neg\psi \mid x) = -\frac{\widehat{\pi}(\psi)}{1-\widehat{\pi}(\psi)}\, s_t(\psi \mid x), \qquad \text{(NOT)} \qquad (17)$$

$$s_t(\psi \wedge \chi \mid x) = s_t(\psi \mid x) + s_t(\chi \mid x), \qquad \text{(AND)} \qquad (18)$$

$$s_t(\psi \vee \chi \mid x) = \frac{\widehat{\pi}(\psi)\widehat{\pi}(\neg\chi)\, s_t(\psi \mid x) + \widehat{\pi}(\chi)\widehat{\pi}(\neg\psi)\, s_t(\chi \mid x)}{\widehat{\pi}(\psi) + \widehat{\pi}(\chi) - \widehat{\pi}(\psi)\widehat{\pi}(\chi)}, \qquad \text{(OR-CI)} \qquad (19)$$

$$s_t(\psi \vee \chi \mid x) = \frac{\widehat{\pi}(\psi)\, s_t(\psi \mid x) + \widehat{\pi}(\chi)\, s_t(\chi \mid x)}{\widehat{\pi}(\psi) + \widehat{\pi}(\chi)}, \qquad \text{(OR-ME)} \qquad (20)$$

with matching recursive rules for $\widehat{\pi}(\varphi)$ given in Alesiani et al. (2026). For each disjunction one uses the OR-CI rule if the subformulas are conditionally independent given $x_t = x$, and the OR-ME rule if they are mutually exclusive.

Velocities. Under the same conditions as above, the velocities of Gaussian path FM combine in a very similar way as the scores. Here $v(x \mid c_i)$ is the conditional velocity field and $v(x)$ is the unconditional velocity field. By Theorem 1 the Atomic, NOT and OR-ME rules below are exact for velocities on any stochastic interpolant. Only the AND rule, and through inclusion–exclusion the OR-CI rule, require the Gaussian-path assumption, with the error of the AND rule given in closed form by Theorem 2.

$$v_t(c_i \mid x) = v_t(x \mid c_i) - v_t(x), \qquad \text{(Atomic)} \qquad (21)$$

$$v_t(\neg\psi \mid x) = -\frac{\widehat{\pi}(\psi)}{1-\widehat{\pi}(\psi)}\, v_t(\psi \mid x), \qquad \text{(NOT)} \qquad (22)$$

$$v_t(\psi \wedge \chi \mid x) = v_t(\psi \mid x) + v_t(\chi \mid x), \qquad \text{(AND)} \qquad (23)$$

$$v_t(\psi \vee \chi \mid x) = \frac{\widehat{\pi}(\psi)\widehat{\pi}(\neg\chi)\, v_t(\psi \mid x) + \widehat{\pi}(\chi)\widehat{\pi}(\neg\psi)\, v_t(\chi \mid x)}{\widehat{\pi}(\psi) + \widehat{\pi}(\chi) - \widehat{\pi}(\psi)\widehat{\pi}(\chi)}, \qquad \text{(OR-CI)} \qquad (24)$$

$$v_t(\psi \vee \chi \mid x) = \frac{\widehat{\pi}(\psi)\, v_t(\psi \mid x) + \widehat{\pi}(\chi)\, v_t(\chi \mid x)}{\widehat{\pi}(\psi) + \widehat{\pi}(\chi)}. \qquad \text{(OR-ME)} \qquad (25)$$

The recursive rules for the noise aware classifiers $\widehat{\pi}(\varphi)$ stay also the same as above.

A.2 PROOF OF GAUSSIAN-PATH RESULTS

Lemma 1 (Affine velocity–score relation)**.** *Let $x_t = \alpha(t)x_1 + \beta(t)x_0$ with $x_0 \sim \mathcal{N}(0, I)$ independent of x_1, and let c be a terminal condition determined by x_1 whose noising kernel is independent of c given x_1. Then for every $t \in (0, 1)$*

$$v_t(x \mid c) = a_t x + b_t s_t(x \mid c), \quad \textit{with} \quad a_t = \frac{\dot{\alpha}(t)}{\alpha(t)}, \quad b_t = \frac{\dot{\alpha}(t)}{\alpha(t)}\beta_t^2 - \dot{\beta}(t)\beta(t), \qquad (26)$$

where the coefficients (a_t, b_t) depend on t only, not on c. In particular $v_t(x) = a_t x + b_t s_t(x)$.

Proof of Lemma 1. Let $x_t = \alpha(t)x_1 + \beta(t)x_0$ with $x_0 \sim \mathcal{N}(0, I)$, and let c be any conditioning event under which the noising kernel does not depend on c directly, only through the law of x_1:

$$p_t(x \mid x_1, c) = p_t(x \mid x_1) = \mathcal{N}\big(x;\, \alpha(t)x_1,\, \beta(t)^2 I\big). \qquad (27)$$

Since $x_0 = (x_t - \alpha(t)x_1)/\beta(t)$ holds surely, taking the conditional expectation defining the marginal velocity field,

$$v_t(x \mid c) = \mathbb{E}\big[\dot\alpha(t)x_1 + \dot\beta(t)x_0 \,\big|\, x_t = x,\, c\big] = \frac{\dot\beta(t)}{\beta(t)}\, x \;+\; \left(\dot\alpha(t) - \frac{\dot\beta(t)}{\beta(t)}\alpha(t)\right)\mathbb{E}[x_1 \mid x_t = x, c], \tag{28}$$

using that x_1 enters linearly, so the substitution can be pushed through the expectation term by term.

By equation 27 and Bayes' rule, $p(x_1 \mid x_t = x, c) \propto p_t(x \mid x_1)\, q(x_1 \mid c)$, and differentiating $p_t(x \mid c) = \int p_t(x \mid x_1) q(x_1 \mid c)\, dx_1$ under the integral sign with $\nabla_x \log p_t(x \mid x_1) = -(x - \alpha(t)x_1)/\beta(t)^2$ gives the Tweedie-type identity

$$\mathbb{E}[X_1 \mid X_t = x, c] = \frac{x + \beta(t)^2\, s_t(x \mid c)}{\alpha(t)}, \qquad s_t(x \mid c) := \nabla_x \log p_t(x \mid c). \tag{29}$$

Substituting equation 29 into equation 28:

$$v_t(x \mid c) = \frac{\dot\beta(t)}{\beta(t)}\, x + \left(\dot\alpha(t) - \frac{\dot\beta(t)}{\beta(t)}\alpha(t)\right)\frac{x + \beta(t)^2 s_t(x \mid c)}{\alpha(t)} \tag{30}$$

$$= \frac{\dot\alpha(t)}{\alpha(t)}\, x + \left(\frac{\dot\alpha(t)}{\alpha(t)}\beta(t)^2 - \dot\beta(t)\beta(t)\right) s_t(x \mid c). \tag{31}$$

Thus $v_t(x \mid c) = a_t x + b_t s_t(x \mid c)$ with a_t, b_t as in the statement, independent of c. Taking c to be the trivial conditioning event gives $v_t(x) = a_t x + b_t s_t(x)$. □

Proof of Proposition 1. By Lemma 1, for any conditioning label c,

$$v_t(x \mid c) = a_t x + b_t s_t(x \mid c). \tag{32}$$

Hence

$$\begin{aligned} v_t(x \mid \varphi) &= a_t x + b_t \nabla_x \log p_t(x \mid \varphi) \\ &= a_t x + b_t \left(s_t(x) + \sum_i \alpha_i^{\varphi}(t,x)\big(s_t(x \mid c_i) - s_t(x)\big)\right) \\ &= a_t x + b_t s_t(x) + \sum_i \alpha_i^{\varphi}(t,x)\, b_t\big(s_t(x \mid c_i) - s_t(x)\big) \\ &= v_t(x) + \sum_i \alpha_i^{\varphi}(t,x)\big(v_t(x \mid c_i) - v_t(x)\big), \end{aligned} \tag{33}$$

because we used equation 7 in the first step, and the $a_t x$ terms cancel identically in the last substitution. □

A.3 Logical rules for velocities

We first record the reduction of the problem to the drift, then state and prove Lemma 2, prove Theorem 1 including its circuit consequence, prove Theorem 2 in both of its forms, and finally give the explicit counterexample.

Obstruction in the drift. Using equation 10, expand $\check v_t$ from equation 11 as

$$\check v_t(x \mid \varphi) = \check\mu_t(x \mid \varphi) - \dot\gamma(t)\gamma(t)\, s_t(x \mid \varphi), \tag{34}$$

where $\check\mu_t(x \mid \varphi) := \mu_t(x) + \sum_i \alpha_i^{\varphi}(t,x)\big(\mu_t(x \mid c_i) - \mu_t(x)\big)$ and we used the exact score identity equation 7. Comparing with $v_t(x \mid \varphi) = \mu_t(x \mid \varphi) - \dot\gamma(t)\gamma(t)\, s_t(x \mid \varphi)$, the score terms cancel and

$$\check v_t(x \mid \varphi) - v_t(x \mid \varphi) = \check\mu_t(x \mid \varphi) - \mu_t(x \mid \varphi). \tag{35}$$

Thus $\check v_t = v_t(\cdot \mid \varphi)$ iff

$$\mu_t(x \mid \varphi) = \mu_t(x) + \sum_i \alpha_i^{\varphi}(t,x)\big(\mu_t(x \mid c_i) - \mu_t(x)\big). \tag{36}$$

Therefore, the property of velocity compositionality reduces to the property of the compositionality of drifts.

Drift moments and Boolean algebra.

Lemma 2 (Drift moments and Boolean algebra)**.** *Let $t \in (0,1)$ and x with $p_t(x) > 0$, and let ψ, χ be subformulas with $p_t(\psi \mid x) = \widehat{\pi}(\psi) > 0$. Then, with m_t^{ψ} as in Equation* (12)*,*

$$\mu_t(x \mid \psi) = \frac{m_t^{\psi}(x)}{\widehat{\pi}(\psi)}, \quad m_t^{\neg\psi}(x) = \mu_t(x) - m_t^{\psi}(x), \quad m_t^{\psi\vee\chi}(x) = m_t^{\psi}(x) + m_t^{\chi}(x) - m_t^{\psi\wedge\chi}(x), \tag{37}$$

together with $\widehat{\pi}(\neg\psi) = 1 - \widehat{\pi}(\psi)$ and $\widehat{\pi}(\psi \vee \chi) = \widehat{\pi}(\psi) + \widehat{\pi}(\chi) - \widehat{\pi}(\psi \wedge \chi)$. These identities hold for any interpolant I, noise schedule γ, coupling π and endpoints ρ_0, ρ_1, with no independence or exclusivity assumption.

Proof of Lemma 2. Fix $t \in (0,1)$ and x with $p_t(x) > 0$, and let $\pi_t(\cdot \mid x)$ be the conditional density of (x_0, x_1) given $x_t = x$, so that $m_t^{\psi}(x) = \int \partial_t I \, \mathbf{1}_{\psi} \, d\pi_t(\cdot \mid x)$ and $\widehat{\pi}(\psi) = \int \mathbf{1}_{\psi} \, d\pi_t(\cdot \mid x)$. For $\widehat{\pi}(\psi) > 0$, the conditional law of (x_0, x_1) given $\{x_t = x\} \cap \{x_1 \models \psi\}$ is $\pi_t(\cdot \mid x)$ restricted to $\{x_1 \models \psi\}$ and renormalised by $\widehat{\pi}(\psi)$, hence

$$\mu_t(x \mid \psi) = \frac{\int \partial_t I \, \mathbf{1}_{\psi} \, d\pi_t(\cdot \mid x)}{\widehat{\pi}(\psi)} = \frac{m_t^{\psi}(x)}{\widehat{\pi}(\psi)}. \tag{38}$$

Both $\psi \mapsto m_t^{\psi}(x)$ and $\psi \mapsto \widehat{\pi}(\psi)$ are linear in the indicator $\mathbf{1}_{\psi}$, so the remaining identities follow from the pointwise Boolean identities

$$\mathbf{1}_{\neg\psi} = 1 - \mathbf{1}_{\psi}, \qquad \mathbf{1}_{\psi\wedge\chi} = \mathbf{1}_{\psi}\mathbf{1}_{\chi}, \qquad \mathbf{1}_{\psi\vee\chi} = \mathbf{1}_{\psi} + \mathbf{1}_{\chi} - \mathbf{1}_{\psi}\mathbf{1}_{\chi}. \tag{39}$$

□

Exact rules: negation and exclusive disjunction.

Proof of Theorem 1. Writing $v_t(\psi \mid x) := v_t(x \mid \psi) - v_t(x)$ for the logical velocity of a subformula, the two rules to be proved are

$$v_t(\neg\psi \mid x) = -\frac{\widehat{\pi}(\psi)}{1 - \widehat{\pi}(\psi)} \, v_t(\psi \mid x) \qquad \text{if } \widehat{\pi}(\psi) \in (0,1), \tag{40}$$

$$v_t(\psi \vee \chi \mid x) = \frac{\widehat{\pi}(\psi)\, v_t(\psi \mid x) + \widehat{\pi}(\chi)\, v_t(\chi \mid x)}{\widehat{\pi}(\psi) + \widehat{\pi}(\chi)} \qquad \text{if } \mathbf{1}_{\psi}\mathbf{1}_{\chi} \equiv 0,\ \widehat{\pi}(\psi) + \widehat{\pi}(\chi) > 0, \tag{41}$$

which are the NOT and OR-ME rules of Section A.1. Write $q := \widehat{\pi}(\psi)$, $r := \widehat{\pi}(\chi)$ and $b_t := -\dot{\gamma}(t)\gamma(t)$, so that Equation (10) reads $v_t(\cdot \mid c) = \mu_t(\cdot \mid c) + b_t \, s_t(\cdot \mid c)$ with b_t independent of the conditioning event. It suffices to show that the linear combinations of the drifts is the same as the scores (Section A.1).

Negation. By Lemma 2, $m_t^{\neg\psi}(x) = \mu_t(x) - q\,\mu_t(x \mid \psi)$ and $\widehat{\pi}(\neg\psi) = 1 - q$, so

$$\mu_t(x \mid \neg\psi) - \mu_t(x) = \frac{\mu_t(x) - q\,\mu_t(x \mid \psi)}{1 - q} - \mu_t(x) = -\frac{q}{1-q}\big[\mu_t(x \mid \psi) - \mu_t(x)\big]. \tag{42}$$

The coefficients agree with the rules for the score (Section A.1), which gives the NOT rule.

Exclusive disjunction. If $\mathbf{1}_{\psi}\mathbf{1}_{\chi} \equiv 0$ then $m_t^{\psi\vee\chi} = m_t^{\psi} + m_t^{\chi}$ and $\widehat{\pi}(\psi \vee \chi) = q + r$ by Lemma 2 and Equation (39), hence

$$\mu_t(x \mid \psi \vee \chi) - \mu_t(x) = \frac{q\big[\mu_t(x \mid \psi) - \mu_t(x)\big] + r\big[\mu_t(x \mid \chi) - \mu_t(x)\big]}{q + r}. \tag{43}$$

Here, the weights are again the same as the scores (Section A.1). Adding b_t times the score identities to the drift identities yields Equations (40) and (41).

Conjunction-free circuits. The second claim follows by induction over the circuit. At a leaf, $v_t(c_i \mid x) = v_t(x \mid c_i) - v_t(x)$ holds by definition and $\widehat{\pi}(c_i)$ in practice is supplied by the noise-aware classifier. If the children of a node are exact, then Equations (40) and (41) show that a negation

or exclusive-disjunction node returns the exact logical velocity of its subformula, and the accompanying posterior recursions $\widehat{\pi}(\neg\psi) = 1 - \widehat{\pi}(\psi)$ and $\widehat{\pi}(\psi \vee \chi) = \widehat{\pi}(\psi) + \widehat{\pi}(\chi)$ are exact by Lemma 2. The value at the root is therefore $v_t(x \mid \varphi) - v_t(x)$, i.e. $\check{v}_t(\cdot \mid \varphi) = v_t(\cdot \mid \varphi)$. A conditionally independent disjunction is not covered, because $\mathbf{1}_{\psi\vee\chi} = \mathbf{1}_\psi + \mathbf{1}_\chi - \mathbf{1}_\psi\mathbf{1}_\chi$ contains a conjunction moment. This obstruction, which this creates is covered in Theorem 2. □

The obstruction at conjunction.

Proof of Theorem 2. Define the following function,

$$R_t(x) := \mathbb{E}\Big[\partial_t I(t, x_0, x_1)\big(\mathbf{1}_{c_1} - \widehat{\pi}(c_1)\big)\big(\mathbf{1}_{c_2} - \widehat{\pi}(c_2)\big) \;\Big|\; x_t = x\Big]. \tag{44}$$

We first show that the difference of proxy and true velocity equals $-R_t(x)/\big(\widehat{\pi}(c_1)\widehat{\pi}(c_2)\big)$, and then convert this into the form given in the main text Equation (13).

For $S \subseteq \{1, 2\}$ put $M_S := m_t^{\bigwedge_{i\in S} c_i}(x)$ and $Q_S := \widehat{\pi}(\bigwedge_{i\in S} c_i)$, with $M_\emptyset = \mu_t(x)$ and $Q_\emptyset = 1$, and abbreviate $M_i := M_{\{i\}}$, $Q_i := Q_{\{i\}}$, $M_{12} := M_{\{1,2\}}$, $Q_{12} := Q_{\{1,2\}}$; by Lemma 2, $M_i = Q_i\,\mu_t(x \mid c_i)$. Expanding the product in Equation (44) and using linearity of the conditional expectation,

$$R_t(x) = M_{12} - Q_2M_1 - Q_1M_2 + Q_1Q_2M_\emptyset. \tag{45}$$

Conditional independence gives $Q_{12} = Q_1Q_2 > 0$, so dividing Equation (45) by Q_1Q_2 and using $\mu_t(x \mid \varphi) = M_{12}/Q_{12}$ from Lemma 2,

$$\mu_t(x \mid \varphi) = \frac{R_t(x)}{Q_1Q_2} + \frac{M_1}{Q_1} + \frac{M_2}{Q_2} - M_\emptyset = \frac{R_t(x)}{Q_1Q_2} + \check{\mu}_t(x \mid \varphi), \tag{46}$$

because the AND coefficients are $\alpha_1^\varphi = \alpha_2^\varphi = 1$, for which $\check{\mu}_t(x \mid \varphi) = \mu_t(x \mid c_1) + \mu_t(x \mid c_2) - \mu_t(x)$. Conditional independence also makes the composed score exact, so Equation (35) applies and

$$\check{v}_t(x \mid \varphi) - v_t(x \mid \varphi) = \check{\mu}_t(x \mid \varphi) - \mu_t(x \mid \varphi) = -\frac{R_t(x)}{Q_1Q_2}. \tag{47}$$

Next we show how to convert this into the statement in the main text.

Write $A_i := \mathbf{1}_{c_i}$ and $p_i := \widehat{\pi}(c_i) \in (0, 1)$, and let all expectations be under $\pi_t(\cdot \mid x)$. All four cells then carry positive mass, so the cell-conditional drifts $d_t(a_1, a_2)$ are well defined. Conditioning additionally on the pair (A_1, A_2) and using the tower property,

$$R_t(x) = \mathbb{E}\big[d_t(A_1, A_2)\,(A_1 - p_1)(A_2 - p_2)\big]. \tag{48}$$

Since $(a_1, a_2) \mapsto d_t(a_1, a_2)$ is a function on $\{0, 1\}^2$, it has the exact multilinear expansion

$$d_t(a_1, a_2) = \theta_\emptyset + \theta_1a_1 + \theta_2a_2 + \theta_{12}a_1a_2, \qquad \theta_{12} = \Delta_t(x), \tag{49}$$

with $\theta_\emptyset = d_t(0, 0)$, $\theta_1 = d_t(1, 0) - d_t(0, 0)$ and $\theta_2 = d_t(0, 1) - d_t(0, 0)$; the coefficient of the product term is the mixed second difference Equation (14). Conditional independence of A_1, A_2 together with $A_i^2 = A_i$ gives

$$\mathbb{E}[(A_1 - p_1)(A_2 - p_2)] = 0, \qquad \mathbb{E}[A_i(A_1 - p_1)(A_2 - p_2)] = 0 \quad (i = 1, 2), \tag{50}$$

$$\mathbb{E}[A_1A_2(A_1 - p_1)(A_2 - p_2)] = \mathbb{E}[A_1(A_1 - p_1)]\,\mathbb{E}[A_2(A_2 - p_2)] = p_1(1 - p_1)\,p_2(1 - p_2), \tag{51}$$

so only the interaction term of Equation (49) survives,

$$R_t(x) = p_1(1 - p_1)\,p_2(1 - p_2)\,\Delta_t(x). \tag{52}$$

Substituting into Equation (47) gives $\check{v}_t - v_t(\cdot \mid \varphi) = -R_t(x)/(p_1p_2) = -(1 - p_1)(1 - p_2)\Delta_t(x)$, which is Equation (13). Since $p_1p_2 > 0$, the bias vanishes if and only if $R_t(x) = 0$, equivalently $(1 - p_1)(1 - p_2)\Delta_t(x) = 0$, and the two limiting cases in the statement follow by inspection of the prefactor. □

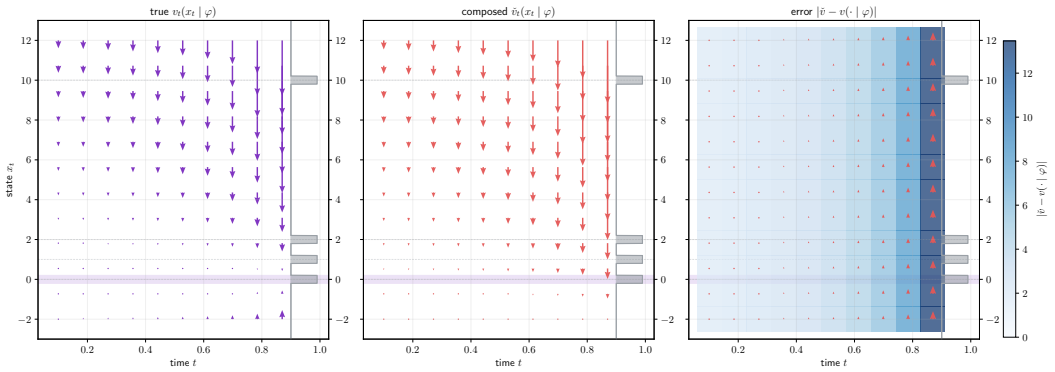

Figure 5: **Explicit counterexample.** Here we plot the situation depicted in the counterexample Appendix A.3. On the left of each plot the source distribution and on the right the target distribution is shown. The first plot depicts the velocity field at different times of the AND composition. (Since the problem is one dimensional the velocity field is also one dimensional and has no component across time) The second plot depicts the linear combined atomic velocity fields. And the right shows the difference between these two, proving that velocities can not be combined together in the same way scores can. The depicted arrows of the velocity are downscaled by a factor 0.03 for readability.

Remark 2 (Gaussian paths force the residual to vanish)**.** *On the Gaussian path of Section 4.1,* $x_t = \alpha(t)x_1 + \beta(t)x_0$ *with* $x_0 \sim \mathcal{N}(0, I)$ *independent of* x_1 *and* $\gamma \equiv 0$*, the relation* $x_0 = (x_t - \alpha(t)x_1)/\beta(t)$ *holds surely, so*

$$\partial_t I(t, x_0, x_1) = \frac{\dot{\beta}(t)}{\beta(t)}\, x_t + \Big(\dot{\alpha}(t) - \frac{\dot{\beta}(t)}{\beta(t)}\alpha(t)\Big)x_1. \tag{53}$$

Given $x_t = x$ *the drift is thus an affine function of* x_1 *with coefficients that do not depend on the conditioning, and by the Tweedie identity equation 29 the conditional mean* $\mathbb{E}[x_1 \mid x_t = x, c]$ *is an affine function of* $s_t(x \mid c)$ *with* c*-independent coefficients. Exact score composition equation 7 therefore propagates to the drifts,* $\breve{\mu}_t(x \mid \varphi) = \mu_t(x \mid \varphi)$*, i.e.* $R_t \equiv 0$ *in Theorem 2. This is the mechanism behind Proposition 1. Since the two indicator variances are strictly positive for* $\widehat{\pi}(c_i) \in (0, 1)$*, Equation* (52) *then also gives* $\Delta_t \equiv 0$*: on a Gaussian path, conditional independence of the atomic* events *already forces additivity of the cell-conditional* drifts*. It is exactly this implication that non-Gaussian endpoints destroy.*

Now that we have seen that the problem is inside the drift part of the velocity field, we can construct an explicit FM counterexample.

Explicit counterexample. We now give a construction where the atoms are *exactly* conditionally independent given x_t, so LogDiff's CI-AND score rule with $\alpha_1^{\varphi} = \alpha_2^{\varphi} = 1$ is the exact composed score, while the LogDiff velocity composition still fails.

We use the flow-matching bridge ($\gamma \equiv 0$, $I(t, x_0, x_1) = (1-t)x_0 + tx_1$) with independent coupling. Fix cluster centres $v_1 = 0$, $v_2 = 1$, $v_3 = 2$, $v_4 = 10$ and a small cluster half-width $\ell/2$ with $\ell \in (0, 1)$ (so the intervals $B_j = [v_j - \ell/2, v_j + \ell/2]$ are disjoint). Take

$$\rho_1(x_1) = \tfrac{1}{4\ell} \sum_{j=1}^{4} \mathbf{1}_{B_j}(x_1), \qquad \rho_0(x_0) = \tfrac{1}{2R}\mathbf{1}_{[-R,R]}(x_0), \tag{54}$$

with R large (concretely $R = 100$). The source is non-Gaussian, so we are not on a Gaussian-path bridge. Define atoms and formula

$$c_1 = \{x_1 \in B_1 \cup B_2\}, \qquad c_2 = \{x_1 \in B_1 \cup B_3\}, \qquad \varphi = c_1 \wedge c_2 = \{x_1 \in B_1\}, \tag{55}$$

and evaluate at $(t, x_t) = (\frac{1}{2}, 5)$. At $t = \frac{1}{2}$, $x_t = (x_0 + x_1)/2$, so a change of variables gives the conditional density of x_t given x_1:

$$f_{x_t|x_1}(x_t \mid x_1) = 2\,\rho_0(2x_t - x_1) = \tfrac{1}{R}\,\mathbf{1}_{[-R,R]}(2x_t - x_1). \tag{56}$$

Define

$$J := \left(\frac{-R + 10 + \ell/2}{2}, \frac{R - \ell/2}{2}\right). \tag{57}$$

For every $x_t \in J$ and every $x_1 \in \bigcup_j B_j \subset [-\ell/2, 10 + \ell/2]$, one has $2x_t - x_1 \in [-R, R]$, so the indicator is 1 and $f_{x_t|x_1}(x_t \mid x_1) = 1/R$ is constant in x_1. By Bayes' rule,

$$f_{x_1|x_t}(x_1 \mid x_t) \;\propto\; \rho_1(x_1)\, f_{x_t|x_1}(x_t \mid x_1) \;\propto\; \rho_1(x_1), \qquad x_t \in J, \tag{58}$$

so the posterior of x_1 given x_t equals ρ_1 throughout J, with mass $\frac{1}{4}$ on each cluster B_j. In particular, $5 \in J$, and throughout this neighborhood

$$P(c_1 \mid x_t) = \tfrac{1}{2}, \qquad P(c_2 \mid x_t) = \tfrac{1}{2}, \qquad P(c_1 \wedge c_2 \mid x_t) = \tfrac{1}{4}. \tag{59}$$

Thus $c_1 \perp c_2 \mid x_t$ holds locally around $x_t = 5$. The corresponding probability identity can therefore be differentiated with respect to x_t, and LogDiff's CI-AND rule gives the exact composed score at $x_t = 5$.

Velocities and LogDiff velocity composition. For $\gamma \equiv 0$ and linear I,

$$v_t(x_t \mid c) = \mathbb{E}[x_1 - x_0 \mid x_t, c] = \tfrac{1}{1-t}\big(\mathbb{E}[x_1 \mid x_t, c] - x_t\big), \tag{60}$$

which at $t = \frac{1}{2}$, $x_t = 5$ becomes $2(\mathbb{E}[x_1 \mid x_t, c] - 5)$. Since each cluster B_j is symmetric around its centre v_j and receives uniform weight, $\mathbb{E}[x_1 \mid x_t = 5,\, x_1 \in B_j] = v_j$ exactly, hence:

$$\begin{aligned}
&\mathbb{E}[x_1 \mid x_t, c_1] = \tfrac{v_1+v_2}{2} = 0.5, && v_t(5 \mid c_1) = -9,\\
&\mathbb{E}[x_1 \mid x_t, c_2] = \tfrac{v_1+v_3}{2} = 1, && v_t(5 \mid c_2) = -8,\\
&\mathbb{E}[x_1 \mid x_t] = \tfrac{v_1+v_2+v_3+v_4}{4} = 3.25, && v_t(5) = -3.5,\\
&\mathbb{E}[x_1 \mid x_t, \varphi] = v_1 = 0, && v_t(5 \mid \varphi) = -10.
\end{aligned} \tag{61}$$

The LogDiff velocity composition (with the correct CI-AND coefficients $\alpha_1^{\varphi} = \alpha_2^{\varphi} = 1$) evaluates to

$$\begin{aligned}
\check{v}_t(5 \mid \varphi) &= v_t(5) + [v_t(5 \mid c_1) - v_t(5)] + [v_t(5 \mid c_2) - v_t(5)]\\
&= -3.5 + (-5.5) + (-4.5) = -13.5 \;\neq\; -10 = v_t(5 \mid \varphi).
\end{aligned} \tag{62}$$

Interpretation. Consistent with equation 35, $\check{v}_t(5 \mid \varphi) - v_t(5 \mid \varphi) = \check{\mu}_t(5 \mid \varphi) - \mu_t(5 \mid \varphi)$, and the residual is

$$\check{\mu}_t(5 \mid \varphi) - \mu_t(5 \mid \varphi) = \tfrac{v_2+v_3-v_1-v_4}{2} = \tfrac{1+2-0-10}{2} = -3.5, \tag{63}$$

matching $-13.5 - (-10) = -3.5$. The drift residual vanishes exactly when the four values form a parallelogram, $v_2 + v_3 = v_1 + v_4$, a condition independent of the CI condition on the atomic probabilities. Choosing $\{v_j\}$ to satisfy uniform posterior (which yields CI at x_t) but to violate $v_2 + v_3 = v_1 + v_4$ therefore isolates the failure of velocity composition from the score composition, which is exact at this point. □

Remark 3 (The counterexample and Theorem 2)**.** *The four clusters are exactly the four cells of the atom pair, $B_1 = c_1 \wedge c_2$, $B_2 = c_1 \wedge \neg c_2$, $B_3 = \neg c_1 \wedge c_2$ and $B_4 = \neg c_1 \wedge \neg c_2$, with cell-conditional drifts $d_t(a_1, a_2) = 2(v_j - 5)$ for the corresponding cluster j. The mixed second difference Equation* (14) *is therefore*

$$\Delta_t(5) = 2\big(v_1 - v_2 - v_3 + v_4\big) = 2\,(0 - 1 - 2 + 10) = 14, \tag{64}$$

and since $\widehat{\pi}(c_1) = \widehat{\pi}(c_2) = \frac{1}{2}$, Equation (13) *predicts the residual $-\big(1 - \frac{1}{2}\big)\big(1 - \frac{1}{2}\big) \cdot 14 = -3.5$, matching the direct computation above. Equivalently, by Equation* (52)*, $R_t(5) = \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot 14 = 0.875$ and $-R_t(5)/\big(\widehat{\pi}(c_1)\widehat{\pi}(c_2)\big) = -3.5$. The "parallelogram" condition $v_2 + v_3 = v_1 + v_4$ is thus precisely the vanishing of the drift interaction Δ_t, which by Theorem 2 is the exact criterion for velocity composition at a conjunction, and which is logically independent of the conditional independence of the atoms holding here.*

A.4 CONSISTENCY OF THE EQUILIBRATION

For fixed $t_{n+1} \in (0,1)$, the continuous-time Langevin dynamics corresponding to Equation (15), driven by the exact composed score $s_{t_{n+1}}(\cdot \mid \varphi) = \nabla_x \log p_{t_{n+1}}(\cdot \mid \varphi)$, admit $p_{t_{n+1}}(\cdot \mid \varphi)$ as an invariant distribution. Under standard regularity and ergodicity assumptions, these dynamics therefore re-equilibrate the sample toward the desired logical marginal, independently of the preceding transport along $\tilde{v}$. The updates $x_{n+1}^{(k)}$ in Equation (15) form the corresponding Euler-Maruyama approximation. Their accuracy improves as Δt_n decreases and the effective equilibration time $K\,\bar{\gamma}(t_{n+1})\,\Delta t_n$ increases. In the joint limit $\Delta t_n \to 0$ and $K\,\bar{\gamma}(t_{n+1})\,\Delta t_n \to \infty$, the corrector recovers $p_{t_{n+1}}(\cdot \mid \varphi)$ at that fixed time. At finite K, the choice of $\tilde{v}$ remains important because a well-aligned transport step provides a better initialization for re-equilibration. Taking $\tilde{v} \equiv 0$ and $K \to \infty$ yields score-only Langevin dynamics at fixed t. In principle, writing Equation (15) as a Langevin step with step size $\varepsilon_t = \frac{1}{2}\bar{\gamma}(t)\,\Delta t$ shows that this coefficient is a free hyperparameter of our method, and could be chosen differently. Unless stated otherwise we use this schedule-tied default. Reasons for this choice and an ablation are given in Appendix B.5.

A.5 ALGORITHMS

Algorithm 1 LOGSI: logical sampling with transport–re-equilibration guidance

Require: Boolean formula φ; initial state $x \sim \rho_0$; grids $0 = t_0 < \cdots < t_N = 1$; equilibration budget K; flags ODE, PROXY

for $n = 0, \ldots, N-1$ **do**
  $t \leftarrow t_n$; $\Delta t \leftarrow t_{n+1} - t_n$
  if PROXY **then**
    $\alpha_i \leftarrow$ COMPOSEWEIGHTS(φ, x, t) ▷ Algorithm 2
    $\tilde{v}_t(x \mid \varphi) \leftarrow v_t(x) + \sum_i \alpha_i^{\varphi}(t,x)\big(v_t(x \mid c_i) - v_t(x)\big)$ ▷ Equation (11)
  else
    $\tilde{v}_t(x \mid \varphi) \leftarrow v_t(x)$
  if ODE **then**
    $x \leftarrow x + \tilde{v}_t(x \mid \varphi)\,\Delta t$ ▷ ODE transport
  else
    $s_t(x \mid \varphi) \leftarrow$ COMPOSESCORE(φ, x, t) ▷ Algorithm 3
    $x \leftarrow x + \big(\tilde{v}_t(x \mid \varphi) + \frac{1}{2}\bar{\gamma}(t)\,s_t(x \mid \varphi)\big)\Delta t + \sqrt{\bar{\gamma}(t)\Delta t}\,\xi,\ \xi \sim \mathcal{N}(0, I)$ ▷ SDE transport
  if $t_{n+1} \in (0,1)$ **then**
    $t \leftarrow t_{n+1}$
    for $k = 1, \ldots, K$ **do**
      $s_t(x \mid \varphi) \leftarrow$ COMPOSESCORE(φ, x, t) ▷ Algorithm 3
      $x \leftarrow x + \frac{1}{2}\bar{\gamma}(t)\,s_t(x \mid \varphi)\,\Delta t + \sqrt{\bar{\gamma}(t)\Delta t}\,\eta,\ \eta \sim \mathcal{N}(0, I)$ ▷ corrector, Equation (15)
return x

Algorithm 2 COMPOSEWEIGHTS: recursive LogDiff coefficients $(\alpha_i^{\varphi})_{i=1}^{k}$ (Section 3.2)

Require: Boolean formula φ over atoms $\{c_i\}_{i=1}^{k}$; state x; time t; classifier posteriors $\widehat{\pi}(\cdot) = p_t(\cdot \mid x)$
Ensure: coefficients $\{\alpha_i^{\varphi}(t,x)\}_i$ such that $s_t(\varphi \mid x) = \sum_i \alpha_i^{\varphi}\, s_t(c_i \mid x)$

$(\alpha^{\varphi},\, \widehat{\pi}(\varphi)) \leftarrow$ LOGICALWEIGHTS(φ)
return α^{φ}

function LOGICALWEIGHTS(ψ) ▷ returns $(\alpha^{\psi}, \widehat{\pi}(\psi))$ with $\alpha^{\psi} \in \mathbb{R}^k$
  if $\psi = c_j$ is atomic **then**
    $\alpha \leftarrow e_j$ ▷ one-hot: $\alpha_j = 1$, $\alpha_{i \neq j} = 0$
    return $(\alpha,\ \widehat{\pi}(c_j))$
  else if $\psi = \neg\chi$ **then**
    $(\alpha_\chi, \pi_\chi) \leftarrow$ LOGICALWEIGHTS(χ)
    return $\big(-\frac{\pi_\chi}{1-\pi_\chi}\,\alpha_\chi,\ 1-\pi_\chi\big)$ ▷ NOT
  else if $\psi = \chi \wedge \xi$ **then**
    $(\alpha_\chi, \pi_\chi) \leftarrow$ LOGICALWEIGHTS(χ); $(\alpha_\xi, \pi_\xi) \leftarrow$ LOGICALWEIGHTS(ξ)
    return $(\alpha_\chi + \alpha_\xi,\ \pi_\chi \pi_\xi)$ ▷ AND
  else if $\psi = \chi \vee \xi$ and mutually exclusive **then**
    $(\alpha_\chi, \pi_\chi) \leftarrow$ LOGICALWEIGHTS(χ); $(\alpha_\xi, \pi_\xi) \leftarrow$ LOGICALWEIGHTS(ξ)
    $\pi \leftarrow \pi_\chi + \pi_\xi$
    return $\big(\frac{\pi_\chi}{\pi}\alpha_\chi + \frac{\pi_\xi}{\pi}\alpha_\xi,\ \pi\big)$ ▷ OR-ME
  else if $\psi = \chi \vee \xi$ and conditionally independent **then**
    $(\alpha_\chi, \pi_\chi) \leftarrow$ LOGICALWEIGHTS(χ); $(\alpha_\xi, \pi_\xi) \leftarrow$ LOGICALWEIGHTS(ξ)
    $\pi \leftarrow \pi_\chi + \pi_\xi - \pi_\chi \pi_\xi$
    $\alpha \leftarrow \frac{\pi_\chi(1-\pi_\xi)}{\pi}\alpha_\chi + \frac{\pi_\xi(1-\pi_\chi)}{\pi}\alpha_\xi$
    return $(\alpha,\ \pi)$ ▷ OR-CI

Algorithm 3 COMPOSESCORE: LogDiff score composition via composed weights (Sections A.1 and 3.2)

Require: Boolean formula φ; state x; time t; atomic networks $s_t(x)$, $s_t(x \mid c_i)$
Ensure: composed conditional score $s_t(x \mid \varphi) = \nabla_x \log p_t(x \mid \varphi)$

$\{\alpha_i^{\varphi}(t,x)\}_i \leftarrow$ COMPOSEWEIGHTS(φ, x, t) ▷ Algorithm 2
return $s_t(x) + \sum_{i=1}^{k} \alpha_i^{\varphi}(t,x)\big[s_t(x \mid c_i) - s_t(x)\big]$ ▷ Equation (7)

A.6 Relation to SuperDiff

SuperDiff Skreta et al. (2025) introduce a method to track changes in the conditional densities $p_t(x \mid c_i)$ along a sample path without training a noise-aware classifier. Their Theorem 1 gives an Itô estimator for $\mathrm{d}\log p_t(x \mid c_i)$. In principle, this construction could also be applied to stochastic interpolants and bridges as in our setting, although they only develop it for diffusion models.

For logical AND, they choose weights κ_i (with $\sum_i \kappa_i = 1$) by requiring equal infinitesimal density updates, $\mathrm{d}\log p_t(x \mid c_i) = \mathrm{d}\log p_t(x \mid c_j)$, and transport the state with a κ-weighted linear combination of the atomic SDE drifts, so that in expectation the endpoint is equally likely under each conditional model, $p_1(x \mid c_i) = p_1(x \mid c_j)$ for all i, j. Crucially, matching the *increments* $\mathrm{d}\log p_t(x \mid c_i)$ along the trajectory only pins down terminal densities *up to the initial gap*: integrating from 0 to 1 gives

$$\log p_1(x_1 \mid c_i) - \log p_1(x_1 \mid c_j) \;=\; \log p_0(x_0 \mid c_i) - \log p_0(x_0 \mid c_j), \tag{65}$$

so the construction lands on the AND manifold $\{x : p_1(x \mid c_i) = p_1(x \mid c_j)\ \forall i, j\}$ only when the initial laws already agree, $p_0(x \mid c_i) = p_0(x \mid c_j)$ for all i, j. This is true for standard diffusion models, where generation begins from a shared Gaussian independent of the condition, but fails in our bridge setting: we do not start from noise but from explicit samples $x_0 \sim \rho_0$, and in general $p_0(x \mid c_i) \neq p_0(x \mid c_j)$. By equation 65, the endpoint gap then equals the initial gap and the resulting sample is not on the AND manifold.

A further practical difference concerns the divergence term in their density update. For VP/OU diffusion, $\langle \nabla, f_t \rangle$ in Theorem 1 is available in closed form because the forward drift f_t is linear. For general stochastic interpolants one would instead need div of each atomic drift $v_t(\cdot \mid c_i) + \frac{1}{2}\bar{\gamma}(t)\, s_t(\cdot \mid c_i)$, which for neural v_t, s_t must be estimated stochastically at every step (e.g. via Hutchinson's estimator), adding per-step cost and variance to the density updates rather than being a fundamental obstruction.

For these reasons we do not compare against SuperDiff in our experiments. Adapting their density-tracking construction to bridge interpolants, either by re-deriving the AND weights so that equation 65 is closed by the choice of κ, or by initialising from a coupling under which the initial gap vanishes, is a natural future direction.

B Experimental Details

B.1 Training hyperparameters

Training hyperparameters for the velocity and score networks are summarised in Table 2, architecture and evaluation-metric details follow in subsequent subsections.

Table 2: **Training hyperparameters** for our experiments. CFG dropout means the probability with which the condition is exchanged with the unconditional symbol.

Hyperparameter	CMNIST	CelebA	QM9
Epochs	500	500	500
Batch size	1024	64	128
Learning rate	10^{-3}	2×10^{-4}	10^{-4}
Weight decay	10^{-5}	0.01	0.01
Noise scheduler $\gamma(t)$	$4\,t(1-t)$	$8\,t(1-t)$	$t(1-t)$
Optimizer / scheduler	AdamW + cosine annealing to 0 over training		
CFG dropout	20% per-label / per-atom unconditional tokens		

B.2 Datasets and transfer protocols

EMNIST to CMNIST. The source distribution ρ_0 is the EMNIST dataset restricted to ten letter classes (a–e, A–E), the target distribution ρ_1 is the Colored MNIST dataset with ten digit and ten

Table 3: **Classifier accuracies** for the metrics used in the CMNIST and CelebA experiments.

Dataset	Attribute	Accuracy
CMNIST	Digit	0.996
	Color	1.000
	Joint	0.996
CelebA	Glasses	0.993
	Hat	0.986
	Joint	0.979

colour classes (10 colours are multiplied by the greyscale MNIST samples). Images are rendered as 28×28 RGB tensors in $[0, 1]$. Each sample is initialised from a concrete source image and transported toward CMNIST while enforcing a Boolean formula φ over $\texttt{Digit}(d)$ and $\texttt{Color}(c)$ atoms. We evaluate fixed templates (AND, OR_{ME}, OR_{CI}, AND$\neg$) and randomly generated expression trees of depth two–three over the same atom vocabulary, while ensuring that conditional independence and mutual exclusivity hold. Cross-axis formulas such as $\texttt{Digit}(d) \vee \texttt{Color}(c)$ use the OR-CI rule of Section A.1, while same-axis pairs such as $\texttt{Digit}(d_1) \vee \texttt{Digit}(d_2)$ (or the analogue for colours) use OR-ME.

CelebA. We train a stochastic interpolant from CelebA *female* to *male* images in the latent space of the FLUX.1-schnell VAE (16 channels, 16×16 spatial resolution). Logical atoms are two binary CelebA attributes: `Glasses` and `Hat`. We test the logical statements AND, OR-CI, and AND$\neg$. Decoded outputs are evaluated at 128×128 in pixel space.

QM9. We bridge noisy *source* molecules drawn from the QM9 dataset (matched in atom count to each target) to target molecules from QM9. Each graph carries 3D coordinates and a 5-way atom-type vector over $\{\text{H}, \text{C}, \text{N}, \text{O}, \text{F}\}$, coordinates are center-of-mass centred. Atoms are binary molecular attributes (ring presence, oxygen presence, etc.) composed with the same LogDiff rules as in Section 3.2. The main logical statement pairs `has_rings`$= 1$ with `has_oxygen`$= 1$ under AND, OR-CI, and AND$\neg$. Bridge sources are restricted to molecules whose ground-truth labels *violate* the rule, so conformity measures whether the sampler can *create* the constraint from a non-conforming start.

B.3 Model architectures

Each benchmark uses a separate trained velocity network v_θ, and score network s_θ. A time-aware classifier for posterior weights in OR / negation, and a held-out evaluator for the conformity metric is also trained (the molecular experiments uses rdkit as conformity metric). All the models were trained in a CFG way, to directly predict unconditioned and conditioned output with the same model.

B.3.1 EMNIST to CMNIST

Velocity and score fields. Both fields are U-Net-style ResNets with base width 96, conditioning dimension 64, and 3-channel 28×28 inputs. The encoder-decoder has three downsample stages (average pooling) and skip connections; residual blocks use FiLM-style modulation(Perez et al., 2018) from the combined $(\text{digit}, \text{colour}, t)$ embedding. The velocity head outputs a 3-channel field, the score head outputs a 3-channel score and, on its bottleneck features, two auxiliary classifiers (digit and colour logits) used at sampling time.

Noise-aware classifier (guidance). During sampling, atom log-probabilities $\widehat{\pi}(c_i \mid x_t)$ for disjunction and negation are read from the *score* network's auxiliary heads: the model is queried with the unconditional wildcard label on both axes, and log-softmax probabilities are taken for the atom's digit or colour value at the current (x_t, t). No separate guidance classifier is trained for CMNIST.

Conformity evaluator. Rule conformity on generated images is measured with a held-out classifier: ResNet18 adapted for 28×28 inputs (3×3 stem, no initial max-pool) with separate linear heads for digit and colour. This model is trained only on clean Colored MNIST and is *not* used inside the sampler. Accuracy of the classifier is reported in Table 3.

B.3.2 CELEBA

Velocity and score fields. Both fields are class-conditional SiT transformers trained on 16×16 VAE latents with 16 input channels, patch size 1, hidden size 384, depth 12, and 6 attention heads. CelebA's 40 binary attributes are embedded with a multi-label encoder (per-attribute 3-way tables: negative / positive / wildcard) and summed with a sinusoidal timestep embedding. Outputs are unpatchified to a 16-channel velocity or score field in latent space.

Noise-aware classifier (guidance). OR and AND$\neg$ require time-conditioned atom posteriors on the *noisy latent* x_t. We train a SiT model on the same latent geometry (patch size 2, hidden size 192, depth 6, 6 heads) with timestep conditioning only. The model outputs 40 independent logits; at step t the probability that attribute a equals value v is $\sigma(\ell_a)$ if $v{=}1$ and $1-\sigma(\ell_a)$ if $v{=}0$. With probability 0.1 during classifier training, $t{=}0$ and clean latents are shown so the network is calibrated near the source endpoint.

Conformity evaluator. After VAE decoding to 128×128 RGB, conformity uses a ResNet classifier trained on male CelebA samples with a 40-way sigmoid head (as a training starting point the ResNet classifier was initialized with the ImageNet trained weights). Accuracy of the classifier is reported in Table 3 (Only the tested attributes are reported).

B.3.3 QM9

Velocity and score fields. Both fields are EGNNs built on fully connected molecular graphs (all ordered atom pairs within each molecule). We use the EDM-stabilised EGNN variant with hidden size 256, depth 9, and 5 atom-type channels. Node inputs are continuous one-hot atom vectors and 3D coordinates; a conditioning module adds a sinusoidal time embedding plus embeddings for the binary graph labels (`has_rings`$=1$, `has_oxygen`$=1$ etc.), each with an extra "unknown" index for classifier-free guidance. The network predicts equivariant position velocities and invariant atom-type velocities; discrete types are recovered by argmax at the end of sampling.

Noise-aware classifier (guidance). The classifier shares the same graph construction but is a classification-only stack (hidden size 256, depth 6): equivariant message-passing blocks, mean pooling over nodes, layer norm, and a linear head producing one logit per binary attribute. It is trained on noisy graphs (x_t, h_t) with the same bridge noise as the generator; atom posteriors for OR / AND$\neg$ are sigmoids of the corresponding logits at the current t.

Validity and conformity. Chemical validity and rule satisfaction are evaluated with RDKit (not with a learned classifier like in the image cases). Conformity is defined only on RDKit-valid molecules.

B.4 EVALUATION METRICS

B.4.1 IMAGE BENCHMARKS

Rule conformity. For a batch with logical formula φ, conformity is the fraction of generated images whose *evaluator* predictions satisfy φ. On CMNIST and CelebA the evaluator is the ResNet18 classifier, see last section. This is distinct from the noise-aware classifiers used inside the sampler.

FID. We report Fréchet Inception Distance (FID) with Inception-v3 `pool3` features (2048-dimensional, `torchmetrics` implementation, `normalize=True`). Generated and reference images are bilinearly resized to 128×128 and clamped to $[0, 1]$ before feature extraction.

For CMNIST, FID is computed *per expression*: for each φ in the evaluation schedule, the reference set is the subset of real Colored MNIST training images whose ground-truth digit and colour satisfy

φ; generated samples for that batch are compared to this expression-conditioned reference (up to $\min(10{,}000, |\text{subset}|)$ real images). The reported FID is the mean across expression batches.

For CelebA, it is similar. We report the FID scores with the expression-conditioned references (up to $5{,}000$ reals).

B.4.2 QM9 METRICS

RDKit validity. For each molecule we build an RDKit conformer from predicted coordinates and atomic numbers, infer bonds, and run `SanitizeMol`. The *validity* rate is the fraction of samples for which this pipeline succeeds.

Rule conformity. On valid molecules only, we recompute binary attributes from the perceived structure: ring-based labels via RDKit ring perception (`has_rings`, etc.) and element-presence labels by atomic number. Conformity is the fraction of valid molecules whose recomputed attributes satisfy φ (AND / OR / AND$\neg$ over the two chosen atoms).

Joint success. The primary success metric reported in the main figures is validity $\times$ conformity (equivalently, the fraction of *all* samples that are both valid and rule-conforming).

Structural edit distance (RMSD). To calculate the distance between source and final molecule, we first apply the Kabsch algorithm (Kabsch, 1976) after centering both structures at zero centre of mass, removing rigid rotation ambiguity, and second calculate the RMSD between the positions of the atoms. As a baseline, we calculate the mean Kabsch-RMSD between two random QM9 molecules. This is purely a geometric metric, the atomtypes are not taken into account.

Novelty. Valid molecules are converted to canonical SMILES. *Novelty* is the fraction of *unique* valid SMILES not present in a reference set of canonical SMILES for the full QM9 dataset.

B.4.3 COMPUTE MATCHING AND ERROR BARS

Network-evaluation (NFE) budgets for LOGSI vs. matched baselines follow Section 5: outer steps N with K corrector substeps are matched against a $K{=}0$ baseline with additional integration steps, such that the number of evaluations of the velocity and score networks match. When multiple training or evaluation seeds are run, main-text error bars aggregate across seeds for training and evaluation.

B.5 ADDITIONAL EXPERIMENTS

Langevin step size ablation The Langevin corrector step size ε is in principle a free hyperparameter of LOGSI. In the main experiments we use the schedule-tied choice

$$\varepsilon_t = \tfrac{1}{2}\bar{\gamma}(t)\,\Delta t, \tag{66}$$

which reuses the SI noise scheduler, stays consistent with the Δt-based equilibration limit in the main text, and introduces no extra hyperparameter. As an ablation we instead use the variance-exploding (VE) signal-to-noise (SNR) rule of Song et al. (2021), originally proposed for Langevin correctors in score-based *diffusion* sampling:

$$\varepsilon = 2\Big(r\,\frac{\|z\|}{\|s_t(x \mid \varphi)\|}\Big)^2, \qquad x \leftarrow x + \varepsilon\, s_t(x \mid \varphi) + \sqrt{2\varepsilon}\, z, \qquad z \sim \mathcal{N}(0, I), \tag{67}$$

with target SNR $r > 0$.

We apply the same rule in our bridge / stochastic-interpolant sampler, where the corrector targets the logically composed marginal $p_t(\cdot \mid \varphi)$ rather than a reverse diffusion. Because this choice introduces the additional hyperparameter r, we grid-search $r \in \{0.05, 0.1, 0.16, 0.2\}$ and report the value $r = 0.1$ with highest conformity in the case of the QM9 dataset. Figure 6 shows that SNR tuning can raise logical conformity, but the resulting steps are often large enough that source identity is effectively lost (Kabsch RMSD approaching the level of unrelated source–target pairs). In the QM9 editing setting this is undesirable, so we keep the schedule-tied step size as the default: it trades a possible conformity gain for edits that remain local to the source. Designing step sizes that improve conformity without erasing source identity is left to future work.

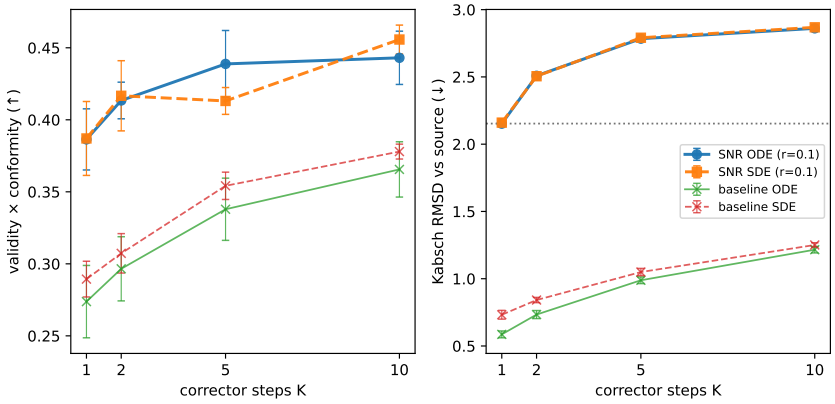

Figure 6: **Langevin step size ablation on QM9.** Left: conformity, right: Kabsch distance to the source as a function of correction steps. We compare schedule-tied steps Equation (66) to the SNR rule of Song et al. (2021) Equation (67), for both ODE and SDE transport.

Table 4: **Transport velocity ablation.** 1 step ablation of the experiment given in Table 1

Method	CMNIST	CelebA	QM9
LOGSI 1 step	$\mathbf{0.79 \pm 0.03}$	$\mathbf{0.54 \pm 0.12}$	$\mathbf{0.27 \pm 0.03}$
uncd. 1 step	0.020 ± 0.001	0.016 ± 0.009	0.15 ± 0.02
$\frac{1}{2}v_t(x\|c_1) + \frac{1}{2}v_t(x\|c_2)$ 1 step	0.44 ± 0.02	0.16 ± 0.05	0.20 ± 0.01
only score 1 step	0.033 ± 0.002	0.018 ± 0.006	0.016 ± 0.005

Table 5: **Transport velocity ablation.** Ablations on each of the proposed experiments, where the transport step velocity is exchanged by unconditional velocity, mean velocity between the two conditions and not using a transport step.

Method	CMNIST	CelebA	QM9
LOGSI 1 step	$\mathbf{0.84 \pm 0.02}$	$\mathbf{0.49 \pm 0.12}$	$\mathbf{0.29 \pm 0.01}$
uncd. 1 step	0.022 ± 0.002	0.02 ± 0.01	0.19 ± 0.02
$\frac{1}{2}v_t(x\|c_1) + \frac{1}{2}v_t(x\|c_2)$ 1 step	0.51 ± 0.01	0.14 ± 0.05	0.246 ± 0.008
only score 1 step	0.033 ± 0.002	0.018 ± 0.006	0.016 ± 0.005
LOGSI 5 step	$\mathbf{0.912 \pm 0.004}$	$\mathbf{0.58 \pm 0.09}$	$\mathbf{0.354 \pm 0.009}$
uncd. 5 step	0.150 ± 0.007	0.05 ± 0.02	0.294 ± 0.004
$\frac{1}{2}v_t(x\|c_1) + \frac{1}{2}v_t(x\|c_2)$ 5 step	0.686 ± 0.004	0.25 ± 0.05	0.327 ± 0.006
only score 5 step	0.034 ± 0.001	0.030 ± 0.002	0.053 ± 0.007

B.6 SAMPLES

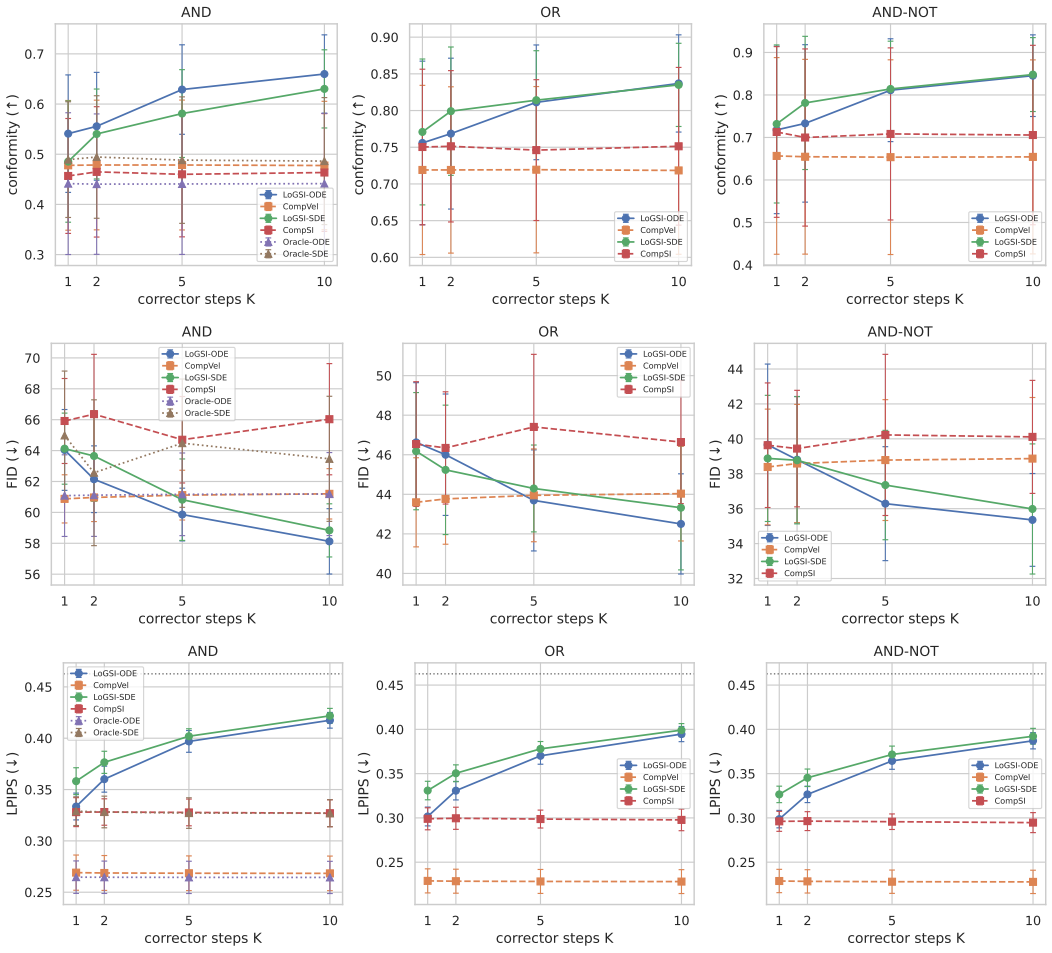

Figure 7: **Additional CelebA experiments.** Further metrics of the experiment regarding transfer learning on CelebA. We report conformity, FID score and additionally we report LPIPS distance to the starting sample. The dotted horizontal is the mean random LPIPS score between female source and male target sample in the CelebA dataset.

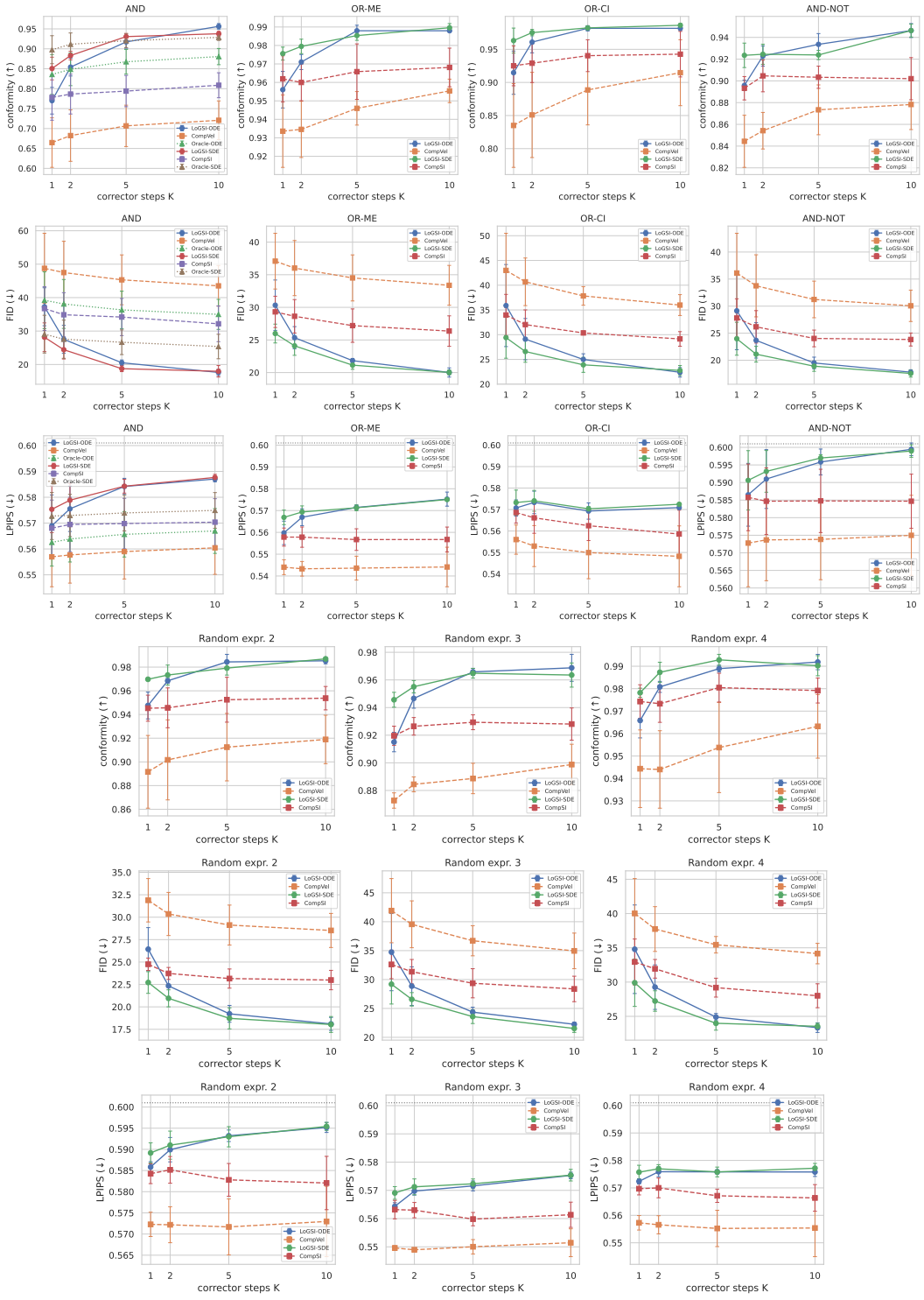

Figure 8: **Additional CMNIST experiments.** Further metrics of the experiment regarding transfer learning from EMNIST to CMNIST. We report FID score and additionally we report LPIPS distance to the starting sample. The dotted horizontal is the mean random LPIPS score between source sample in EMNIST and target sample in the CMNIST dataset.

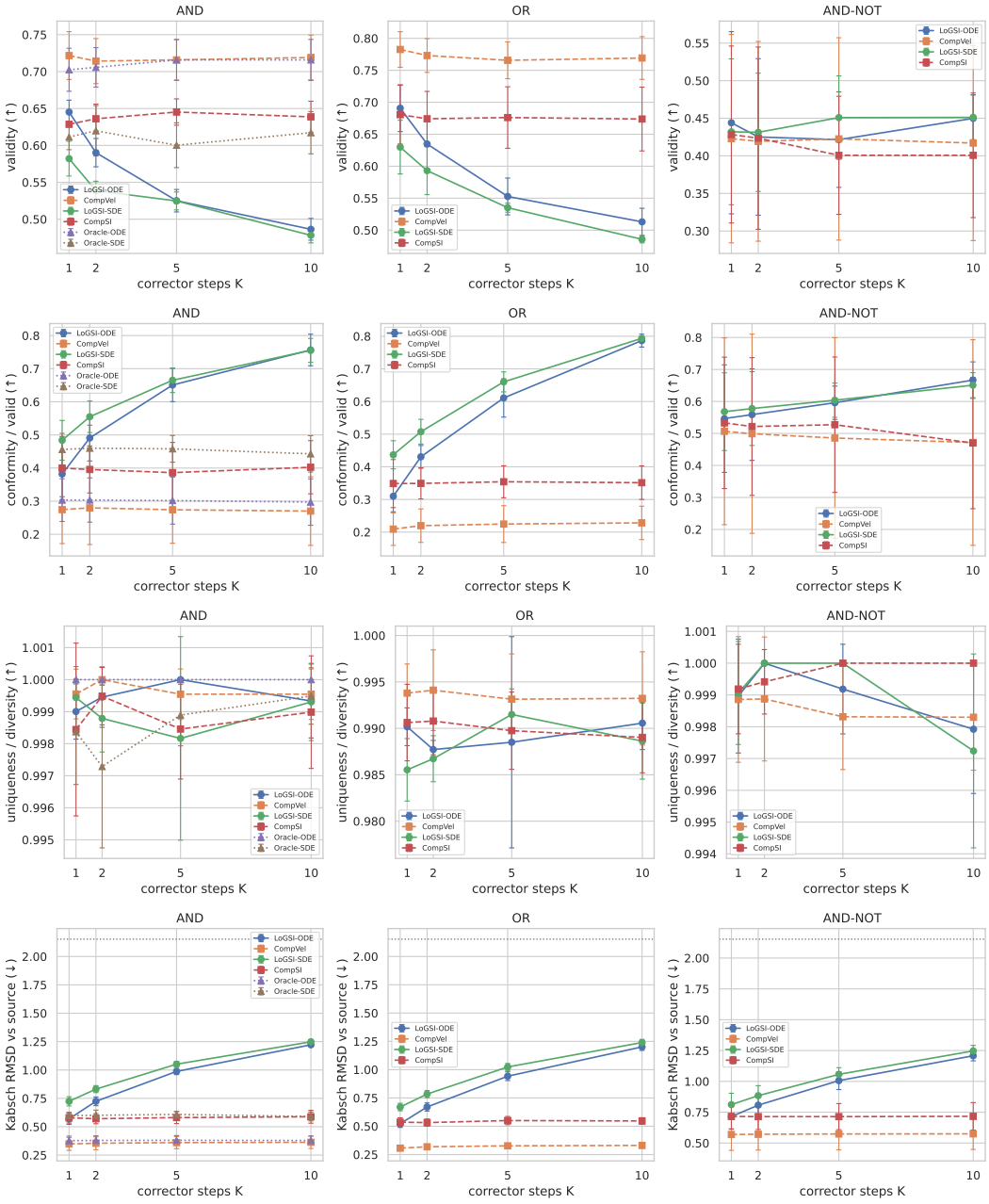

Figure 9: **Additional QM9 experiments.** Further metrics of the experiment regarding QM9 editing. We report validity and conformity separately. Furthermore, we report uniqueness and Kabsch RMSD. The dotted horizontal is the mean random Kabsch RMSD score between QM9 samples.

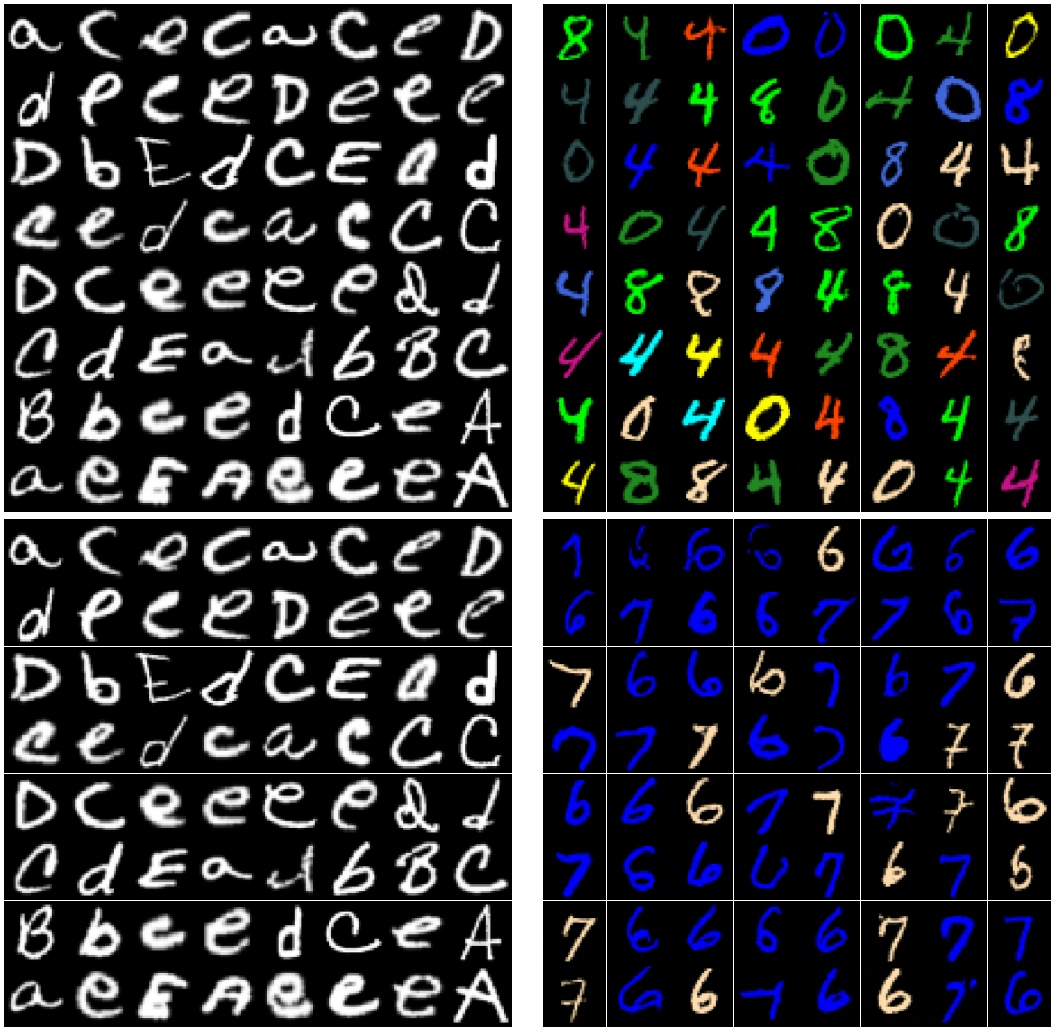

Figure 10: **Samples of the CMNIST experiments.** We plot samples from different randomly generated logical expressions. On the top we use the expression $(8 \vee 4) \vee (4 \vee 0)$ and on the bottom we use the expression $(7 \vee 6) \wedge (\texttt{beige} \vee \texttt{blue})$.



Figure 11: **Samples of the CelebA experiments.** Here we show samples from the logical condition `glasses` $\vee$ `hat`. On the left the source images and on the right the final images are shown.

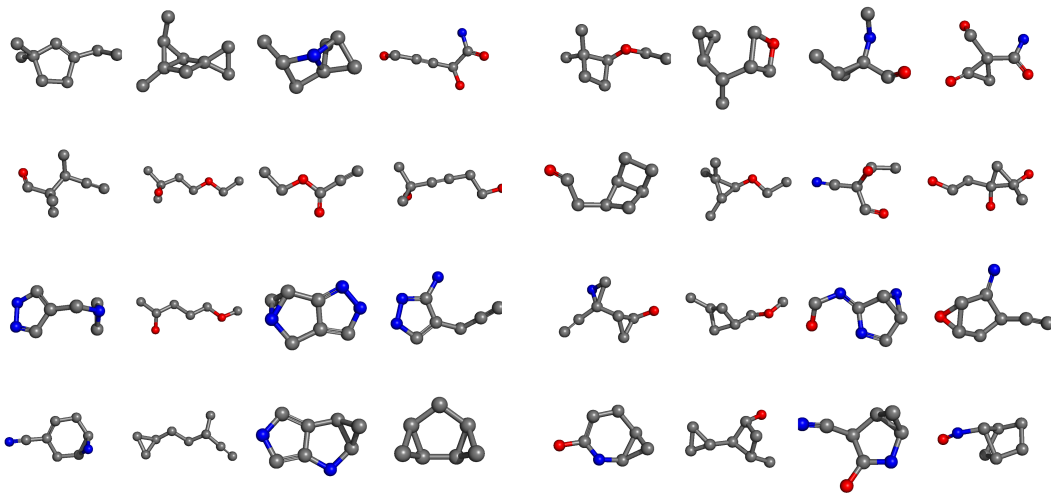

Figure 12: **Samples of the QM9 experiments.** Both valid and conform samples of the logical statement `rings` $\wedge$ `oxygen` are shown. On the left the starting molecules and on the right the generated molecules are depicted.