HOT FUZZ: TEMPERATURE-TUNABLE COMPOSITION OF DIFFUSION MODELS WITH FUZZY LOGIC

Anonymous authors
Paper under double-blind review

ABSTRACT

Composing pretrained diffusion models provides a cost-effective mechanism to encode constraints and unlock complex generative capabilities. Prior work relies on crafting compositional operators that seek to extend set-theoretic notions such as union and intersection to diffusion models, e.g., using a product or mixture of the underlying energy functions. We expose the inadequacy and inconsistency of combining these operators in terms of limited mode coverage, biased sampling, instability under negation queries, and failure to satisfy basic compositional laws such as idempotency and distributivity. We introduce a principled calculus grounded in fuzzy logic that resolves these issues. Specifically, we define a general class of conjunction, disjunction, and negation operators that generalize the classical mixtures, illustrating how they circumvent various pathologies and enable precise combinatorial reasoning with score models. Beyond existing methods, the proposed *Dombi* operators afford complex generative outcomes such as Exclusive-Union (XOR) of individual scores. We establish rigorous theoretical guarantees on the stability and temperature scaling of Dombi compositions, and derive Feynman-Kac correctors to mitigate the sampling bias in score composition. Empirical results on image generation with stable diffusion and multi-objective molecular generation substantiate the conceptual, theoretical, and methodological benefits. Overall, this work lays the foundation for systematic design, analysis, and deployment of diffusion ensembles.

1 INTRODUCTION

Pretrained general-purpose generative machine learning models (Devlin et al., 2019; Brown et al., 2020) have become practically synonymous with the term artificial intelligence itself. Their vast capabilities (Bommasani, 2021; Wei et al., 2022), however, come at the cost of an excessive need for growing datasets (Kaplan et al., 2020; Villalobos et al., 2022), and yet additional techniques are needed to reach adequate performance in downstream tasks. Finetuning (Devlin et al., 2019), human-feedback-based reinforcement learning (Christiano et al., 2017; Ouyang et al., 2022; Zhang et al., 2023), retrieval augmented generation (Lewis et al., 2020), or even specialized prompting techniques (Brown et al., 2020) are then used to retrofit models to specialized tasks and domains.

As an alternative to monolithic general models, compositional generation (Jordan & Jacobs, 1994; Hinton, 1999; 2002; Yuksel et al., 2012; Vedantam et al., 2018; Du et al., 2020) seeks to combine the domain knowledge from different models to solve a task at hand. As many models follow probabilistic formulations, using probabilistic language for composition is a natural approach. Products of Experts (PoEs) (Hinton, 1999; 2002; Liu et al., 2022; Du et al., 2023; Skreta et al., 2025a) have been devised and widely used as a mechanism to enforce conjunctive constraints, with the idea that their product is only large when all components are large. The assumption underlying this approach to model joint distributions, statistical independence of the factors, however, does not in general hold.

Often tackled as a separate problem is the concept *avoidance* in generation. Similar to other tasks, *unlearning* (Ginart et al., 2019; Nguyen et al., 2022; Wang et al., 2024) as a specific form of finetuning or post-training *avoidance* and steering methods (Dhariwal & Nichol, 2021; Ho & Salimans, 2021; Dong et al., 2023; Garipov et al., 2023; Kirchhof et al., 2025) have been proposed, which often utilize PoE with inverse probability densities for avoidance or rely on training additional models.

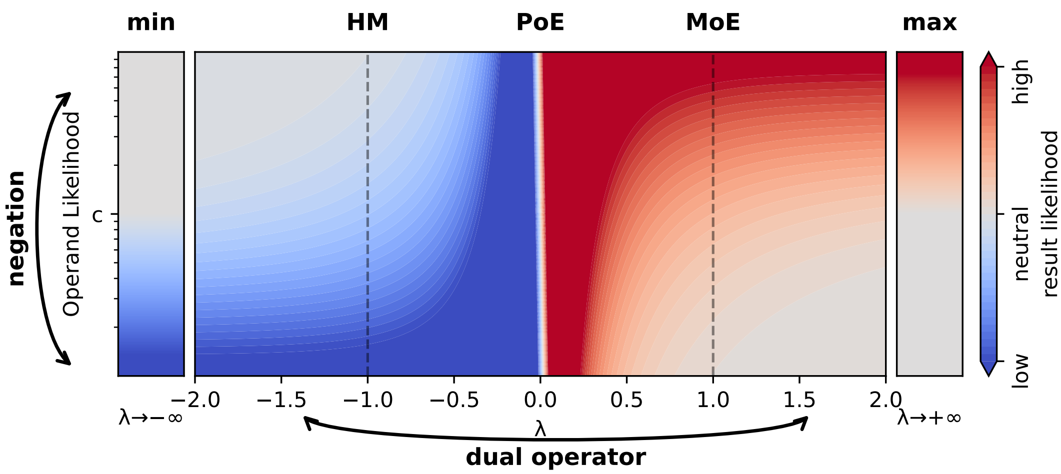


Figure 1: Visualisation of Dombi Composition $p(\mathbf{x}) \circ_\lambda q(\mathbf{x})$ with $q(\mathbf{x})$ fixed. Flipping the sign of λ gives the DeMorgan dual operator. For the negation $\neg_c p(\mathbf{x}) \wedge q(\mathbf{x})$, the y-axis of the figure flips. Different choices of λ correspond to known operators.

Section 3 Failure Modes in Score-Composition
- Unstable Negation. — Section 3.1
- Inconsistent Temperature Scaling. — Section 3.2
- Violation of DeMorgan, Idempotency, Distributivity. — Section 3.3

Section 4 DeMorgan Dual Dombi Composition Operators
- Properties of Dombi Operators. — Section 4.1
- Properties of Stable Negations. — Section 4.2

Section 5 Guarantees on Precision and Temperature Stability
- Bounds on Idempotency and Distributivity Bias. — Corollary 5.1
- Bounds on Composition Stability. — Proposition 5.2

Section 6 Exact Sampling with Feynman-Kac Correction

Section 7 Experiments on Image and Molecule Synthesis

Figure 2: Overview of the main contributions in this work.

In this paper, we investigate compositions of diffusion models from the viewpoint of fuzzy set theory and fuzzy logic. We propose a procedure to derive sets of well-behaved composition operators, and among them, propose *Dombi operators* in Section 4 as a one-parameter family, extending and uniting commonly used operations such as mixture of experts (Jordan & Jacobs, 1994) (MoE), harmonic mean (Garipov et al., 2023) (HM), and as a special case, the geometric mean—a tempered Product of Experts (Hinton, 1999) (PoE), as visualized in Figure 1. In contrast to many existing effective methods, our approach is purely online and utilizes pre-trained diffusion models. An overview of our main contributions is provided in Figure 2.

2 Background and Related Work

2.1 Score-Based Models

We want to approximate a probability distribution p defined over $\mathbb{R}^d$ to sample from it. In the context of score-based modelling, we first recast p as a Boltzmann distribution, and let the model learn the *score function* $s_\theta(\mathbf{x}) \approx \nabla \log p(\mathbf{x})$, avoiding the unknown partition function. To facilitate sampling via MCMC, the data distribution p is gradually destroyed according to the forward noising SDE (Øksendal, 2003)

$$d\mathbf{x}_\tau = f_\tau(\mathbf{x}) d\tau + \sigma_\tau d\overline{\mathbf{w}}_\tau, \qquad \mathbf{x}_0 \sim p(\mathbf{x}_0).$$

Here $f_\tau : \mathbb{R}^d \to \mathbb{R}^d$ is some, usually linear, drift function and $\sigma_\tau : \mathbb{R} \to \mathbb{R}$ is a time-dependent diffusion coefficient and $\overline{\mathbf{w}}_\tau$ is the Wiener process. These functions are chosen such that $\mathbf{x}_{\tau=1} \sim$

$\mathcal{N}(0, \mathbf{I}_d)$, the standard Gaussian. For sampling, we simulate the backward process with $t = 1 - \tau$ as

$$d\mathbf{x}_t = \left[-f_t(\mathbf{x}_t) + \sigma_t^2 \nabla_{\mathbf{x}} \log p_t(\mathbf{x}_t)\right] dt + \sigma_t d\mathbf{w}_t. \tag{1}$$

which satisfies the Fokker-Planck equation

$$\frac{\partial p_t(\mathbf{x})}{\partial t} = -\langle \nabla, p_t(\mathbf{x})(-f_t + \sigma_t^2 \nabla_{\mathbf{x}} \log p_t(\mathbf{x}))\rangle + \frac{\sigma_t^2}{2} \Delta p_t(\mathbf{x}), \tag{2}$$

where Δp_t^i denotes the Laplacian of p_t and $\langle \nabla, \cdot \rangle$ is the divergence operator. For the rest of this paper, we assume we are given a set of pre-trained score models $\{s_t^i\}_{i=1}^k$, which model the respective probability distributions $\{p_t^i\}_{i=1}^k$. For statements about the $t = 1$, we omit the index.

To translate the theory developed in this paper to practice, we rely on efficient density estimation to assign responsibility to score functions. We can efficiently estimate densities during inference with Itô's Lemma (Karczewski et al., 2025a; Skreta et al., 2025b) as

$$d \log p_t(\mathbf{x}_t) \approx \big\langle d\mathbf{x}_t,\ s_t(\mathbf{x}_t)\big\rangle + \Big(\langle \nabla,\ f_t(\mathbf{x}_t)\rangle + \langle f_t(\mathbf{x}_t),\ s_t(\mathbf{x}_t)\rangle - \frac{\sigma_t^2}{2}\|s_t(\mathbf{x}_t)\|^2\Big) dt. \tag{3}$$

2.2 Composition of Score Fields

There is a quickly growing body of work on compositions, mixtures, and products of energy-based models (EBMs), as well as flow and diffusion models. We explicitly focus on *training-free* mixtures of score functions in diffusion. Prior work (Du et al., 2020; Ho & Salimans, 2021; Skreta et al., 2025a;b; Gaudi et al., 2025) mainly bases composition on probabilistic operations on the underlying distributions. As the interpretation of these operations is often logical or set-theoretic, we will use the symbols $\{\vee, \wedge, \neg\}$ to denote them, for both probability densities and their scores. In score-based modelling, conjunctions are then usually represented by (sometimes geometric) products

$$p^1(\mathbf{x}) \wedge_\times p^2(\mathbf{x}) \coloneqq p^1(\mathbf{x}) p^2(\mathbf{x}) \Longrightarrow s^1(\mathbf{x}) \wedge_\times s^2(\mathbf{x}) = s^1(\mathbf{x}) + s^2(\mathbf{x}) \tag{4}$$

and disjunctions by mixtures, where we use the weighting $\alpha^i = \frac{p^i(\mathbf{x})}{p^1(\mathbf{x}) + p^2(\mathbf{x})}$, with

$$p^1(\mathbf{x}) \vee_+ p^2(\mathbf{x}) \coloneqq \frac{1}{2} p^1(\mathbf{x}) + \frac{1}{2} p^2(\mathbf{x}) \Longrightarrow s^1(\mathbf{x}) \vee_+ s^2(\mathbf{x}) \ = \alpha^1 s^1(\mathbf{x}) + \alpha^2 s^2(\mathbf{x}). \tag{5}$$

Two noteworthy exceptions from product-based conjunctions are Garipov et al. (2023), who model conjunctions with the *harmonic mean* $p^1(\mathbf{x})p^2(\mathbf{x}) / \big(p^1(\mathbf{x}) + p^2(\mathbf{x})\big)$ and Skreta et al. (2025b), who reweigh individual scores to steer towards equal density directly.

Importantly, under the usual dynamics of diffusion processes, for $t \neq 1$, nonlinear compositions do not commute with the noising operator, i.e., $p_t^1 \vee_+ p_t^2 = (p^1 \vee_+ p^2)_t$ but $p_t^1 \wedge_\times p_t^2 \neq (p^1 \wedge_\times p^2)_t$. This means that naive composition of perturbed score models leads to a bias that can be corrected with methods like sequential Monte Carlo (SMC) (Skreta et al., 2025a; Thornton et al., 2025). The typical formulation of Equations (1) and (2) is then extended to *weighted* SDEs, where samples have time-dependent log-weights w_t which are defined via the weight field $g_t(\mathbf{x})$ as

$$dw_t = \overline{g}(\mathbf{x}_t) dt \Longrightarrow \frac{\partial p_t(\mathbf{x})}{\partial t} = \overline{g}_t(\mathbf{x}) p_t(\mathbf{x}), \quad \text{with} \quad \overline{g}(\mathbf{x}) \coloneqq g_t(\mathbf{x}) - \int g_t(\mathbf{x}) p_t(\mathbf{x}) dt.$$

These weighted SDEs with $g_t(\mathbf{x})$ then must satisfy the Feynman-Kac PDE

$$\frac{\partial p_t(\mathbf{x})}{\partial t} = -\langle \nabla, p_t(\mathbf{x})(-f_t + \sigma_t^2 \nabla_{\mathbf{x}} \log p_t(\mathbf{x}))\rangle + \frac{\sigma^2}{2} \Delta p_t(\mathbf{x}) + \overline{g}_t(\mathbf{x}) p_t(\mathbf{x}). \tag{6}$$

For nonlinear score operations like annealing, CFG, or PoE, Skreta et al. (2025a) then explicitly derive the biases incurred by approximating the true composed distribution with the composition of noisy scores, collect the "left-over" terms in g, and use additional correction methods. We adapt their formalism to improve the simulation of our operators in Section 4.

To *avoid* certain distributions, EBM's and score models are usually only negated *relative* to others (Vedantam et al., 2018; Du et al., 2020; 2023; Garipov et al., 2023; Dong et al., 2023; Skreta et al., 2025a; Gaudi et al., 2025), as also done in classifier-free guidance (Ho & Salimans, 2021) (CFG). In these settings, independent concept negation (ICN) for a concept y is often defined, for $0 < \gamma < 1$

as $p(\mathbf{x}|\neg y) \propto p(\mathbf{x})/p(\mathbf{x}|y)^{\gamma}$ in the EBM context (Hinton, 2002; Du & Kaelbling, 2024). In more recent work (Liu et al., 2022; Du et al., 2020; Ho & Salimans, 2021), often the formulation $p(\mathbf{x}|\neg) \propto p(\mathbf{x})^{1+\gamma} p(\mathbf{x}|y)^{-\gamma}$ used instead, derived via Bayes rule.

From a perspective of logic, these variants make use of the reciprocal as pseudo-inverse $\neg p(\mathbf{x}|y) = 1/p(\mathbf{x}|y)$, but to our knowledge, explicit negations in score-models are not often explored or theoretically justified, and alternatives (Chang et al., 2024) also lack clear theoretical interpretation.

2.3 Fuzzy Logic

Our proposed method directly draws from the theory of fuzzy logic. Fuzzy logic relaxes classical logic from a binary domain to real-valued *memberships* in $[0, 1]$. We follow the definitions and notation from Klement et al. (2013) for the following concepts. We define a *t-norm*, a generalization of conjunction or intersection operations, as a function $T : [0, 1]^2 \to [0, 1]$ which is commutative, associative, monotonously increasing, and fulfills the boundary condition $\forall x \in [0, 1] : T(x, 1) = x$. Under the standard negation $N(x) = 1 - x$, we can define the *dual t-conorm* $S : [0, 1]^2 \to [0, 1]$, the corresponding disjunction, via DeMorgan's law as $S(x, y) = N(T(N(x), N(y)))$.

T-norms that are *strict*, i.e., continuous and strictly increasing, can be *generated* (Dombi, 1982; Klement et al., 2013) by a continuous, strictly decreasing function $f : [0, 1] \to [0, \infty]$ with $f(1) = 0$, as so-called *additive generator*, i.e., $T(x, y) := f^{-1}(f(x) + f(y))$. For this work, the parametrised Dombi t-norm is the most important representative, generated by $f_\lambda(x) = (\frac{1}{x} - 1)^\lambda$. A favorable property of the Dombi t-norm is that $\lim_{\lambda\to\infty} T_\lambda = T_M = \min$. The min t-norm T_M together with $S_M = \max$ is the *only* continuous DeMorgan dual that is idempotent with $T_M(x, x) = x$ and distributive with $T_M(x, S_M(y, z)) = S_M(T_M(x, y), T_M(x, z))$ (Klement et al., 2013). To make the domain of probability densities compatible with the theory of fuzzy logic, we utilize some bijective, order-preserving function $\phi : \mathbb{R}_{\geq 0} \cup \{\infty\} \to [0, 1]$ which converts densities into fuzzy membership.

3 Failure Modes in Score Composition

We provide further motivation for our approach with a brief illustration of the mismatch between expectation and true behaviour for score composition using PoE and MoE methods. Existing operators do not carry the well-understood and favorable properties of fuzzy set operators. This makes them ill-equipped to deal with more complex compositions of models or to encode model constraints.

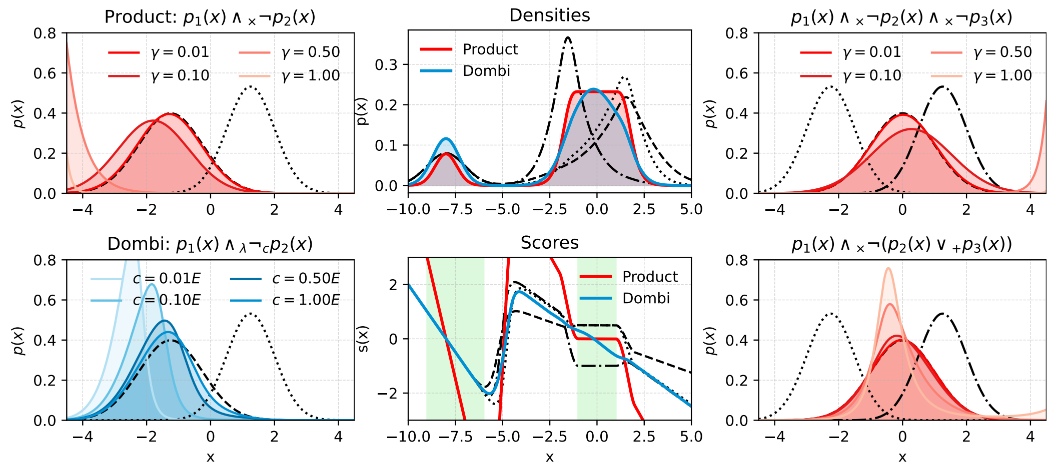
3.1 Unstable Negation

We first discuss the EBM-style negation $p^1(\mathbf{x})/p^2(\mathbf{x})^\gamma$. While widespread, this negation seems to have seen only limited theoretical investigation. While the score operation is straightforward, negative prompts tend to shift the target distribution (Garipov et al., 2023; Chang et al., 2024; Ban et al., 2024) and require careful calibration of the γ parameter. For the simplest case $p^1(\mathbf{x})/p^2(\mathbf{x})$, normalizability can generally not be guaranteed, unless $p^1(\mathbf{x})$ decays much faster in the tails than $p^2(\mathbf{x})$.

The common CFG-style negation in diffusion, $p^1(\mathbf{x})^{1+\gamma}/p^2(\mathbf{x})^\gamma$, has more favorable properties in terms of stability. However, theoretical arguments for its use are still limited in the relevant theory. In Section 4.2, we explore this formalism for negations more in depth, without the context of conditional generation. While better behaved, CFG-style negation still exhibits unfavorable properties, like *overaccentuation* of $p^1(\mathbf{x})$ where $p^2(\mathbf{x})$ vanishes (Chidambaram et al., 2024), leading to a similar bias as the one depicted in Figure 3c.

3.2 Inconsistent Temperature Scaling

PoE uses the score calculus $s^1 \wedge_\times s^2 := s^1 + s^2$. This leads to a scaling of scores depending on their alignment: $\|s^1 \wedge_\times s^2\| = \sqrt{\|s^1\|^2 + \|s^2\|^2 + 2\|s^1\|\|s^2\|\cos\theta}$, where θ is the angle between s^1, s^2. In diffusion, temperature scaling is one of the main methods to control the behavior of the model (Guo et al., 2017; Karczewski et al., 2025b;a). As the alignment of scores can generally be assumed to be arbitrary, PoE arbitrarily changes temperature-scaling behavior. In regions with high score alignment (small θ), temperature is decreased, and the composition is biased towards higher density regions than what is dictated by any component. Conversely, in regions with



(a) Negation Stability (b) Temperature Rescaling (c) No DeMorgan Duality

Figure 3: Failure Modes of PoE composition in combinatorial settings. a illustrates that ICN can lead to unstable behaviour, compared to referenced Dombi negation. b shows an intersection, where the product can lead to locally underscaled *and* overscaled temperatures simultaneously (green), in contrast to Dombi composition. c shows that logically equivalent formulas result in different PoE/MoE compositions.

low alignment between scores ($\theta > \pi$), the temperature is increased, *discouraging* higher density regions. Figure 3b illustrates this behavior in contrast to the Dombi operators, which guarantee $\|s^1 \wedge_\times s^2\| \leq \max\{\|s^1\|, \|s^2\|\}$. Moving from the usual PoE to a geometric mean with $s^1(\mathbf{x})/2 + s^2(\mathbf{x})/2$, this problem does not disappear, rather shift: While the geometric mean does not overscale scores, the effective temperature of the composition is higher than intended, for the same reason as in classic PoE.

3.3 Composition Properties

Model composition is often interpreted as a *logical* operation over the underlying models. This interpretation leads to pitfalls, as MoE and PoE do not exhibit the favourable properties expected of logical or set operations. An important example of this is avoiding *multiple* distributions p_2, p_3 *individually*. Intuitively, one might use a conjunction over multiple negated distributions. The resulting operation, however, does not match the expected result, as negations and conjunctions commute:

$$p^1 \wedge_\times \tilde{\neg}\, p^2 \wedge_\times \tilde{\neg}\, p^3 = p^1 \wedge_\times \tilde{\neg}\, (p^2 \wedge_\times p^3) = \frac{p^1}{p^2 p^3} \neq p^1 \wedge_\times \tilde{\neg}\, (p^2 \vee_+ p^3) = \frac{p^1}{p^2 + p^3}.$$

This pitfall is a manifestation of failure to adhere to DeMorgans law and shown in Figure 3c. In a more general sense, PoE is also neither idempotent, as $p \wedge_\times p = p^2 \neq p$ and distributes only in one direction, i.e., $(p^1 \wedge_\times p^2) \vee_+ p^3 \neq (p^1 \vee_+ p^3) \wedge_\times (p^2 \vee_+ p^3)$. This severely restricts the options for rewriting compositions for different purposes, such as collecting terms.

4 Dombi Operators

In this section, we extend the definition of T-norm-conorm pairs to obtain DeMorgan dual density and score operators. Appendix A describes the exact requirements to generate a set of DeMorgan dual operators. As a special class we propose and investigate the DeMorgan operators generated by $f_\lambda(x) = \left(\frac{1}{x} - 1\right)^{-\lambda}$ and map between densities and membership with $\phi_c(x) = \frac{x}{x+c}$ for $\lambda, c \in \mathbb{R}_{\geq 0}$. This choice of f not only recovers the Dombi t-norm, but ϕ_c expresses negation with *reference* to some constant c. This constant can be interpreted as a normalising factor and serves as a neutral element in negations. As our composition properties act at each $\mathbf{x}$ independently, we can choose a *different constant* for each value: $c(\mathbf{x})$. In the context of distributions, this normalization by a reference distribution $c(\mathbf{x})$ is analogous to the probability ratios used in CFG, or the PoE conjunction, e.g., presented by Liu et al. (2022). With abuse of notation, we will write $\phi_c(p(\mathbf{x})) := \phi_c(p;\mathbf{x}) = \frac{p(\mathbf{x})}{p(\mathbf{x})+c(\mathbf{x})}$.

Definition 4.1 (Dombi Operators). *Choose $\lambda \in \mathbb{R}_{>0}$ and a continuously differentiable function $c : \mathbb{R}^d \to \mathbb{R}_{\geq 0}$ with $s_c = \nabla_{\mathbf{x}} \log c$. For $f_\lambda(x) = \left(\frac{1}{x} - 1\right)^\lambda$ and $\phi_c(p(\mathbf{x})) = \frac{p(\mathbf{x})}{p(\mathbf{x})+c(\mathbf{x})}$, let $\alpha^i_\lambda = \frac{\exp(\lambda \log p^i(\mathbf{x}))}{\sum_{j\in\{1,2\}} \exp(\lambda \log p^j(\mathbf{x}))}$. The* Dombi operators *are the DeMorgan dual operators induced by f_λ, ϕ_c:*

$$\neg_c p(\mathbf{x}) := \frac{c(\mathbf{x})^2}{p(\mathbf{x})} \implies \neg_c s(\mathbf{x}) = 2s_c(\mathbf{x}) - s(\mathbf{x}) \tag{7}$$

$$p^1(\mathbf{x}) \wedge_\lambda p^2(\mathbf{x}) := \frac{p^1(\mathbf{x})p^2(\mathbf{x})}{\left(p^1(\mathbf{x})^\lambda + p^2(\mathbf{x})^\lambda\right)^{1/\lambda}} \implies s^1(\mathbf{x}) \wedge_\lambda s^2(\mathbf{x}) = \alpha^1_{-\lambda} s^1(\mathbf{x}) + \alpha^2_{-\lambda} s^2(\mathbf{x}) \tag{8}$$

$$p^1(\mathbf{x}) \vee_\lambda p^2(\mathbf{x}) := \left(p^1(\mathbf{x})^\lambda + p^2(\mathbf{x})^\lambda\right)^{1/\lambda} \implies s^1(\mathbf{x}) \vee_\lambda s^2(\mathbf{x}) = \alpha^1_{\lambda} s^1(\mathbf{x}) + \alpha^2_{\lambda} s^2(\mathbf{x}) \tag{9}$$

A detailed derivation of this result can be found in Appendix A.

This definition bears multiple remarkable properties. While being constructed to adhere to DeMorgan duality, we can see many similarities to the existing body of work.

4.1 Properties of Dombi Operators

First, dombi compositions over distributions are power norms, and with different choices for the exponent λ, we recover well-known operators, such as $\min$ for $\lambda \to -\infty$, the harmonic mean for $\lambda = -1$, the conventional mixture for $\lambda = 1$, and $\max$ for $\lambda \to \infty$. For $\lambda \to 0$, Dombi composition is undefined on densities and log-densities, yet the score calculus for $\lambda \to 0$ is equivalent to the geometric mean. These relations are visualized in Figure 1. This resemblance is consistent with *power means* (Amari, 2007), which differ from the Dombi operators by a constant factor of $1/2^\lambda$, resulting in equivalent score operators, and tying Dombi composition closely to α-divergence. While derived score operators are equivalent, power means are not associative and cannot form a logic that allows for nesting of operations.

4.2 Properties of referenced Negation

Under our definition, referenced negation results in an expression equivalent to CFG-style negation for $\gamma = 1$. We argue that this is favorable from both the perspectives of fuzzy logic and probability theory. The reference (unconditional) distribution $c(\mathbf{x})$ forms a *neutral element* for negation, i.e., $\neg_c c(\mathbf{x}) = c(\mathbf{x})$, which is semantically intuitive for conditional generation. From a perspective of probability theory, we know that a negated distribution results in a normalizable distribution under bounded χ^2 divergence. We have, per definition (Nishiyama & Sason, 2020)

$$\chi^2(p||q) := \int \frac{(p(\mathbf{x}) - q(\mathbf{x}))^2}{q(\mathbf{x})} dx = \int \frac{p(\mathbf{x})^2}{q(\mathbf{x})} - 1 < \infty. \tag{10}$$

Negation with other γ violates properties of the logic: $\neg_{c,\gamma} p(\mathbf{x}) := c(\mathbf{x})^{1+\gamma}/p(\mathbf{x})^\gamma$ is not involutive for positive $\gamma \neq 1$. In practice, this might not be problematic if compositions are in negation normal form (NNF).

Combined, our composition and negation show strong grounding in existing theory and are, by definition, equipped for model composition far beyond the simple use cases of MoE and PoE. In the next section, we describe how their behaviour in score composition changes for different values of λ.

5 Influence of λ on Distributivity and Mixture Stability

Besides the connection to prior work, the parameter λ from the Dombi operators naturally appears as inverse temperature in the score composition. For $\lambda \to \infty$, the Dombi operators recover the exact $\{\min, \max\}$ lattice and with it distributive and idempotent behavior. For finite λ, the simple bounds in Proposition A.3 can be used to quantify biases in density compositions. We use this to present a simple bound for the maximal density bias we introduce when applying distributive laws.

Corollary 5.1 (Idempotency and Distributivity Bias). *Let $\wedge_\lambda, \vee_\lambda$ be the Dombi density operators. From Proposition A.3 it follows that*

$$\forall x \in \mathbb{R}_{\geq 0}: \qquad x \vee_\lambda x = 2^{1/\lambda} x, \qquad x \wedge_\lambda x = 2^{-1/\lambda} x \tag{11}$$

$$\begin{aligned} \forall x, y, z \in \mathbb{R}_{\geq 0}: \qquad & x \vee_\lambda (y \wedge_\lambda z) \in ((x \vee_\lambda y) \wedge_\lambda (x \vee_\lambda z)) 2^{\pm 2/\lambda} \\ \forall x, y, z \in \mathbb{R}_{\geq 0}: \qquad & x \wedge_\lambda (y \vee_\lambda z) \in ((x \wedge_\lambda y) \vee_\lambda (x \wedge_\lambda z)) 2^{\pm 2/\lambda} \end{aligned} \tag{12}$$

These easily obtainable bounds trivially generalize to arbitrary compositions, allowing us to make immediate statements about the stability of our composition. As our score coefficients vary during the inference process, we would naturally be interested in the rate of change of these coefficients, as drastic change rates might cause the composite model to "oscillate" between two scores, especially in conjunctions. As before, the statement can be extended to more complex formulas trivially.

Proposition 5.2 (Mixture Stability). *Let $\alpha_t = \mathit{softmax}_1(\lambda \log p^1, \lambda \log p^2)$, for a dombi composition $p^1 \circ_\lambda p^2$. Then it holds for the scores s_t^1, s_t^2*

$$|\mathbb{E}[d\alpha_t \mid \mathbf{x}_t]| \leq \frac{\sigma_t^2}{8} \|\lambda s^1 - \lambda s^2\| (\|s^1\| + \|s^2\| + \frac{1}{2} \|\lambda s^1 - \lambda s^2\|) dt \tag{13}$$

Together, Corollary 5.1 and proposition 5.2 quantify the tradeoff between compositional precision and mixture stability. High λ results in small biases over the ground truth of the composition, but for large differences between the component scores $\|s_t^1 - s_t^2\|$, the mixing coefficients α^i might drastically oscillate. When λ is chosen smaller, the volatility of the mixture is naturally bounded.

6 Precise Sampling with Feynman-Kac Correction

While Definition 4.1 explicitly states how the densities and consequently the scores of our target distribution look, simulation with, e.g., $d\mathbf{x}_t = \left[-f_t(\mathbf{x}_t) + \sigma_t^2 (s_1(\mathbf{x}) \wedge_\lambda s_1(\mathbf{x}))\right] dt + \sigma_t d\overline{\mathbf{w}}$ will not *not* sample from the desired marginals during the reverse process and consequently not from the correct target distribution $p_1(\mathbf{x}) \wedge_\lambda p_2(\mathbf{x})$. Skreta et al. (2025a) introduce *Feynman-Kac Correctors* (FKCs) for diffusion, which correct for the biases of score composition. We recast the composition with Dombi operators as weighted SDEs, then collect all terms that are missing from our score proposal into the weight field g. At inference time, SMC methods like systematic sampling can be used to correct for these biases.

In this section, we extend the FKC terms to our Dombi operators, and refer to Appendix B.1 for proofs. As the Dombi-composition just reduces to "power norms" of our densities, as well as a special case of geometric averages in the case of referenced negation, we present these two correction terms here. More complex compositions then propagate the weight-fields $g_t(\mathbf{x})$ of components.

Proposition 6.1 (Referenced Negation as CFG+FKC, Skreta et al., 2025a). *Consider two diffusion models $q_t^1(\mathbf{x}), q_t^2(\mathbf{x})$ defined via the Fokker-Planck equation in Equation (2). The weighted SDE corresponding to the referenced negation of $p_t(\mathbf{x}) \propto \neg_{q_t^2(\mathbf{x})} q_t^1(\mathbf{x})$ is, with $dw_t(\mathbf{x}) = g_t(\mathbf{x}) dt$*

$$\begin{aligned} d\mathbf{x}_t &= \left[-f_t(\mathbf{x}_t) + \sigma_t^2 (2\nabla \log q_t^2(\mathbf{x}_t) - \nabla \log q_t^1(\mathbf{x}_t))\right] dt + \sigma_t d\overline{\mathbf{w}}_t \\ g_t(\mathbf{x}) &= \sigma_t^2 \|\nabla \log q_t^1(\mathbf{x}_t) - \nabla \log q_t^2(\mathbf{x}_t)\|^2 + 2 g_t^2(\mathbf{x}) - g_t^1(\mathbf{x}), \end{aligned} \tag{14}$$

As stated in Equation (10), $p_t(\mathbf{x})$ is then a normalizable probability distribution, if and only if $\chi^2(q_t^1 || q_t^2) < \infty$. We might also want to anneal q^2 to tune the "narrowness" of the concept we avoid. We propose a combined annealing of the form $q^2(\mathbf{x})^{1+\gamma} / q^1(\mathbf{x})^\gamma$ to allow tuning the two distributions in relation to each other, while still maintaining slightly improved normalizability compared to the standard CFG, and maintaining an unbiased energy estimate for further composition.

Next, we state how FKC terms propagate through connectives. As both our connectives are essentially power-norms with positive or negative exponent, both cases can be handled at once.

Theorem 6.2. *Consider two weighted diffusion models $q_t^1(\mathbf{x}), q_t^2(\mathbf{x})$ defined via the Feynman-Kac equation with weights $g_t^1(\mathbf{x}), g_t^2(\mathbf{x})$, and a parameter $\lambda \in \mathbb{R} \backslash \{0\}$. The weighted SDE corresponding* *to* $p_t(\mathbf{x}) \propto \left(q_t^1(\mathbf{x})^\lambda + q_t^2(\mathbf{x})^\lambda\right)^{1/\lambda}$, *with* $\alpha_t^i = \frac{q_t^i(\mathbf{x})^\lambda}{q_t^1(\mathbf{x})^\lambda + q_t^2(\mathbf{x})^\lambda} \in (0,1)$, *and* $dw_t = g_t(\mathbf{x})dt$ *is*

$$
\begin{aligned}
d\mathbf{x}_t &= \left[-f_t(\mathbf{x}_t) + \sigma_t^2(\alpha_t^1 \nabla \log q_t^1(\mathbf{x}_t) + \alpha_t^2 \nabla \log q_t^2(\mathbf{x}_t))\right] dt + \sigma_t d\overline{\mathbf{w}}_t \\
g_t(\mathbf{x}) &= (1-\lambda)\frac{\sigma^2}{2}\left[\left\|\sum_{i\in\{1,2\}} \alpha_t^i \nabla \log q_t^i(\mathbf{x}_t)\right\|^2 - \sum_{i\in\{1,2\}} \alpha_t^i \|\nabla \log q_t^i(\mathbf{x}_t)\|^2\right] + \sum_{i\in\{1,2\}} \alpha_t^i g_t^i(\mathbf{x}_t).
\end{aligned}
\tag{15}
$$

Proposition 6.1 and theorem 6.2 are presented in a modular form. This allows us to use arbitrary combinations of operators and propagate the log-weights of components.

6.1 Inference Procedure

Together, Definition 4.1, proposition 6.1, and theorem 6.2 define our theoretical basis for arbitrarily nested model composition. During the sampling process, we keep track of the evolution of loglikelihoods with the Itô density estimator from Equation (3). This efficient density estimation method enables us to perform complex model compositions with minimal overhead. During composition, we can then compose our scores, log-likelihoods, and FKC terms with the procedure described in Algorithm 1. To improve sampling, we can use SMC techniques during the simulation trajectories (Naesseth et al., 2019). In our experiments, we use systematic sampling proportional to the exponentially weighted momentary weight-field $\exp\{g_t(\mathbf{x})dt\}$ (Douc & Cappé, 2005).

Algorithm 1: DombiComposition over arbitrary formulas

Input : scores $\{s^i\}_{i=1}^k$, log-likelihoods $\{\log q^i\}_{i=1}^k$, weights $\{g^i\}_{i=1}^k$, formula $F ::= i|\neg_j i|F_1 \circ F_2$

Output: Composite score s, Composite density $\log q$, Composite weight g

if $F = i$ **then return** $s^i, \log q^i, g^i$

else if $F = \neg_j i$ **then return** $2s^j - s^i$, $2\log q^j - \log q^i$, $\sigma_t^2\|s^j - s^i\|^2 + 2g^j - g^i$ `// Prop. 6.1`

else if $F = F_1 \wedge_\lambda F_2$ **then** $\lambda \leftarrow -\lambda$ `// Conjunction is a negative power norm`

`/* Case` $F = F_1 \wedge_\lambda F_2 \mid F_1 \vee_\lambda F_2$`: evaluate subformulas first */`

$\bar{s}^1, \overline{\log q}^1, \bar{g}^1 \leftarrow$ DombiComposition$(\{s^i\}_{i=1}^k, \{\log q^i\}_{i=1}^k, \{g^i\}_{i=1}^k, F_1)$

$\bar{s}^2, \overline{\log q}^2, \bar{g}^2 \leftarrow$ DombiComposition$(\{s^i\}_{i=1}^k, \{\log q^i\}_{i=1}^k, \{g^i\}_{i=1}^k, F_2)$

$\alpha^1 \leftarrow \mathrm{softmax}_1(\lambda\overline{\log q}^1, \lambda\overline{\log q}^2);\ \alpha^2 \leftarrow 1 - \alpha^1$

$\bar{g} \leftarrow (1-\lambda)\frac{\sigma^2}{2}\left[\|\alpha^1\bar{s}^1 + \alpha^2\bar{s}^2\|^2 - (\alpha^1\|\bar{s}^1\|^2 + \alpha^2\|\bar{s}^2\|^2)\right]$ `// Theorem 6.2`

return $\alpha^1\bar{s}^1 + \alpha^2\bar{s}^2$, $\frac{1}{\lambda}\mathit{LogSumExp}(\lambda\overline{\log q}^1, \lambda\overline{\log q}^2)$, $\bar{g} + \alpha^1\bar{g}^1 + \alpha^2\bar{g}^2$

---

7 Experiments

7.1 Combinatorial Bias in Composition Samples

We first test the ability of our method to sample from complex compositions of diffusion models. We compose three pretrained models that generate colored MNIST digits (LeCun, 1998). Our three models are defined as follows: Model p_1 generates the digits $\{0, 1, 2, 3\}$ in cyan, p_2 generates digits smaller 2: $\{0, 1, 0, 1\}$ in cyan or beige and p_3 generates the even digits $\{0, 2, 0, 2\}$ in cyan or beige. We would now like to perform set operations on these 7 unique digits, similar to Garipov et al. (2023), but with general operations. Figure 4 shows a set of chosen set operations on our models. Beyond the intersection $p_\cap = p_1 \wedge p_2 \wedge p_3$ and the union $p_\cup = p_1 \vee p_2 \vee p_3$ we show results for the exclusive-or operation $p_{\text{xor}} = (p_1 \vee p_2) \wedge (\neg p_1 \vee \neg p_2)$, that samples digits from *either* p_1 or p_2 but not from their intersection. We then show $p_{\text{xor}} \wedge p_3 = \{2, 0\}$ as well as $p_{\text{xor}} \wedge \neg p_3 = \{3, 1\}$.

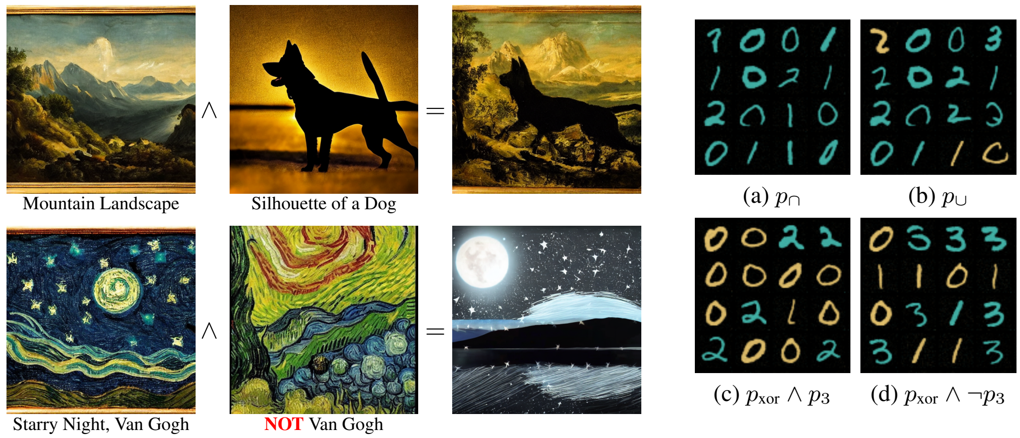


Figure 4: Generated Image Compositions with MNIST ($\lambda \in \{5 \cdot 10^{-3}, 5 \cdot 10^{-2}\}$) and Stable Diffusion ($\lambda = 10$).

As we have no baseline model, we express negation by the mixture of all three models. With few exceptions, we can see that our approach lets us sample from complex compositions like p_{xor}, solely by score-composition of the pretrained diffusion models.

7.2 Multi-Prompt Image Generation and Avoidance

To show the performance of Dombi composition in production scale diffusion models, we compare its ability to generate images that interpolate between or avoid concepts using Stable Diffusion (SD) v1-4. For all our compositions, we choose two prompts c_1, c_2, e.g., `"a mountain landscape"` and `"a silhouette of a dog"`. We then evaluate twenty pairs of images composed conjunctively, as $p(\mathbf{x}|c_1) \wedge p(\mathbf{x}|c_2)$, and compare against `and` Skreta et al. (2025b) and scaled PoE, i.e. unweighted averaging of scores (Liu et al., 2022). We further investigate $p(\mathbf{x}|c_1) \wedge \neg_{p(\mathbf{x})} p(\mathbf{x}|c_2)$ on ten pairs of prompts to illustrate the ability of our model to avoid concepts. As baselines for contrastive prompting, we use ICN (Ho & Salimans, 2021) and the conjunction of (Skreta et al., 2025b), combined with our referenced negation. We use the composed scores in the usual CFG pipeline of SD and measure for all prompts the min. CLIP score (Radford et al., 2021), which measures cosine similarity between image embedding and prompt embedding, and the minimum ImageReward value (Xu et al., 2023), which estimates how closely generated images align with human preferences. For contrastive prompts, we report the difference of each mectric between $c1$ and c_2.

Dombi Composition shows improvement beyond state-of-the-art methods in both CLIP and ImageReward scores, as shown in Tables 5a and 5b. with an example of generated images in Figure 4. For the full list of used prompts, we refer to Appendix C.2. A stark contrast between our method and SuperDiff can be seen in Figure 3b, depicting the mixture stability during the first 100 iterations of the generation process. The batch variances of the mixture coefficient α are shown to correspond nicely to λ, with an increase over time caused by different equilibrium points per batch. Superdiffs `and` shows strong fluctuations in mixing coefficients, especially during the initial iterations. This effect is more pronounced when we retrofit `and` to contrastive settings with our negation definition.

7.3 Multi-Target Protein Synthesis with FKC Correction

As a final experiment, we test Dombi composition combined with FKC in the setting of structure-based drug design (SBDD). The goal here is to generate molecules (ligands) using the structure of a protein as a guide and evaluate their binding energy (Anderson, 2003). In our experiments, we investigate the impact of FKC from Theorem 6.2 on the quality of Dombi composed results. We generated 32 ligands of sizes $\{15, 19, 23, 27, 35\}$ each, for 14 protein pairs, and evaluated their docking scores using Autodock Vina (Eberhardt et al., 2021) and reproduced the experimental setup of (Skreta et al., 2025a). In this experiment, we use annealing on the base distributions: We evaluate $p(\mathbf{x}|\mathtt{P}_1)^\gamma \wedge p(\mathbf{x}|\mathtt{P}_2)^\gamma$ as well as $p(\mathbf{x}|\mathtt{P}_1)^\gamma p(\mathbf{x}|\mathtt{P}_2)^\gamma$, and propagate the FKC term of the annealed base distributions to our dombi operator as in Algorithm 1. Per batch, we report the average joint docking perfor-

Figure 5: Joint generation performance with Stable Diffusion, and paired improvement over baselines with 20 seeds. a shows results for 20 joint prompts $p(\mathbf{x}|c_1) \wedge p(\mathbf{x}|c_2)$. b shows results for 10 contrastive prompts $p(\mathbf{x}|c_1) \wedge \neg_{p(\mathbf{x})} p(\mathbf{x}|c_2)^\gamma$. c shows the variance of α during conjunctive (top) and contrastive (bottom) composition. `and` is from SuperDiff (Skreta et al., 2025b).

(a) Conjunctive prompts

Method	λ	CLIP (↑) mean±std	Im. reward (↑) mean±std
and		24.87$_{\pm 2.92}$	-1.33$_{\pm 0.83}$
PoE		24.41$_{\pm 2.71}$	-1.55$_{\pm 0.75}$
Prompt		23.79$_{\pm 2.72}$	-1.64$_{\pm 0.75}$
Dombi	0.1	25.25$_{\pm 2.79}$	**-1.18**$_{\pm 0.84}$
Dombi	1.0	25.32$_{\pm 2.55}$	-1.16$_{\pm 0.85}$
Dombi	10.0	**25.50**$_{\pm 2.54}$	**-1.18**$_{\pm 0.85}$
Improvements vs. baseline (and)			
and		0.00$_{\pm 0.00}$	0.00$_{\pm 0.00}$
PoE		-0.46$_{\pm 3.11}$	-0.22$_{\pm 1.00}$
Prompt		-5.11$_{\pm 3.25}$	0.13$_{\pm 0.97}$
Dombi	0.1	0.38$_{\pm 2.57}$	0.15$_{\pm 0.83}$
Dombi	1.0	0.46$_{\pm 2.62}$	**0.18**$_{\pm 0.80}$
Dombi	10.0	**0.64**$_{\pm 2.56}$	0.15$_{\pm 0.82}$

(b) Contrastive prompts ($\lambda = 1$)

Method	γ	CLIP (↑) mean±std	Im. reward (↑) mean±std
ICN		7.29$_{\pm 2.76}$	1.14$_{\pm 0.72}$
and ¬		5.44$_{\pm 3.49}$	0.51$_{\pm 0.65}$
Dombi	1	6.91$_{\pm 2.63}$	0.90$_{\pm 0.75}$
Dombi	3	**7.40**$_{\pm 2.62}$	1.11$_{\pm 0.72}$
Dombi	10	7.02$_{\pm 2.54}$	**1.21**$_{\pm 0.66}$
Improvements vs. baseline (ICN)			
ICN		0.00$_{\pm 0.00}$	0.00$_{\pm 0.00}$
and ¬		-1.85$_{\pm 3.54}$	-0.63$_{\pm 0.77}$
Dombi	1	-0.38$_{\pm 2.15}$	-0.24$_{\pm 0.66}$
Dombi	3	**0.10**$_{\pm 1.88}$	-0.03$_{\pm 0.63}$
Dombi	10	-0.27$_{\pm 1.82}$	**0.07**$_{\pm 0.55}$

(c) Mixture Stability

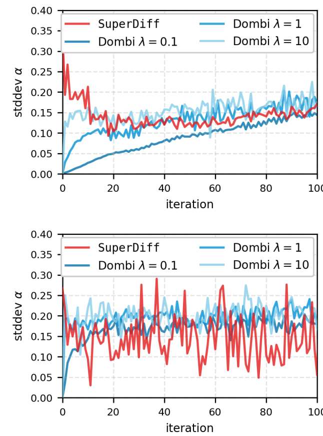


Table 1: Docking Scores of generated ligands for 14 protein target pairs (P_1, P_2), in batches of 32 ligands for 5 molecule lengths each. We compare conjunction with Dombi ($\lambda = 1$) with and without FKC with annealed base distribution and also report TargetDiff from (Guan et al., 2023) as baseline.

Method	Temp. γ	FKC?	($P_1 * P_2$) (↑)	max(P_1, P_2) (↓)	Better than ref. (↑)	Div. (↑)	Val. & Uniq. (↑)	QED (↑)	SA (↓)
TargetDiff	–	–	62.19$_{\pm 27.08}$	−7.24$_{\pm 2.35}$	0.32$_{\pm 0.37}$	**0.89**$_{\pm 0.01}$	0.95$_{\pm 0.07}$	0.57$_{\pm 0.14}$	**0.59**$_{\pm 0.09}$
Dombi	1	✗	68.60$_{\pm 28.09}$	−7.42$_{\pm 2.57}$	0.28$_{\pm 0.34}$	0.88$_{\pm 0.02}$	**0.96**$_{\pm 0.09}$	0.58$_{\pm 0.13}$	**0.59**$_{\pm 0.10}$
Dombi	1	✓	72.83$_{\pm 22.42}$	−7.71$_{\pm 1.65}$	0.27$_{\pm 0.35}$	0.86$_{\pm 0.03}$	0.95$_{\pm 0.08}$	0.57$_{\pm 0.13}$	**0.59**$_{\pm 0.11}$
Dombi	2	✗	71.36$_{\pm 29.44}$	−7.59$_{\pm 2.48}$	0.30$_{\pm 0.34}$	0.88$_{\pm 0.01}$	0.93$_{\pm 0.16}$	**0.59**$_{\pm 0.12}$	0.62$_{\pm 0.09}$
Dombi	2	✓	**81.63**$_{\pm 25.91}$	**−8.25**$_{\pm 1.56}$	**0.38**$_{\pm 0.40}$	0.85$_{\pm 0.11}$	0.93$_{\pm 0.17}$	**0.59**$_{\pm 0.12}$	0.62$_{\pm 0.10}$

mance to each target protein as their product ($P_1 * P_2$), the objective of PoE, as well as $\max(P_2, P_2)$, which is closer to the objective of the Dombi composition. Further, we measure the fraction of molecules that have a higher docking score than the known reference molecules, the diversity of molecules, as well as the fraction of valid *and* unique molecules, and their drug-likeness (QED) (Bickerton et al., 2012) and how easy they are to synthesize (SA) (Ertl & Schuffenhauer, 2009).

FKC Correction improves the docking performance in annealed and unannealed settings, as shown in Table 1. The difference is more pronounced for $\gamma = 2$, where we also collect FKC terms for the annealed base distributions. In Appendix C.3 we show results with an additional small sweep over λ values, where the performance for $\lambda = 0.3$ and $\lambda = 3$ shows to be similar.

8 Conclusion and Future Work

In this work, we introduced Dombi composition operators as a purely online, well-defined, general class of score-composition operators. Based on power norms, our method recovers and unifies prior work, like MoE, the harmonic mean, or contrast operators (Garipov et al., 2023), yet offers theoretical benefits that are crucial to ensure stability when score compositions become more complex. An important future direction are dynamic schedules for λ, as Proposition 5.2 suggests that adaptive choices depending on the scores might be better suited to ensure stability. This work opens up some exciting possibilities, e.g., potential applications in neurosymbolic methods, where modular diffusion models could be coupled to solve combinatorial tasks. Furthermore, the option to rewrite formulas might in principle be utilized to switch to different sampling techniques for, e.g., factoring out subformulas.

9 Reproducibility Statement

Detailed proofs are provided in the Appendix for all our theoretical results. We also provide a link to an anonymous github repository containing all the code used to reproduce the results in this manuscript[1]. The Repository contains the details required to reproduce the empirical results including our hyperparameter settings. We will make our code public under MIT License upon acceptance.

References

Shun-ichi Amari. Integration of stochastic models by minimizing α-divergence. *Neural computation*, 19(10):2780–2796, 2007.

Amy C Anderson. The process of structure-based drug design. *Chemistry & biology*, 10(9):787–797, 2003.

Yuanhao Ban, Ruochen Wang, Tianyi Zhou, Minhao Cheng, Boqing Gong, and Cho-Jui Hsieh. Understanding the impact of negative prompts: When and how do they take effect? In *european conference on computer vision*, pp. 190–206. Springer, 2024.

G Richard Bickerton, Gaia V Paolini, Jérémy Besnard, Sorel Muresan, and Andrew L Hopkins. Quantifying the chemical beauty of drugs. *Nature chemistry*, 4(2):90–98, 2012.

Rishi Bommasani. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Jinho Chang, Hyungjin Chung, and Jong Chul Ye. Contrastive cfg: Improving cfg in diffusion models by contrasting positive and negative concepts. *arXiv preprint arXiv:2411.17077*, 2024.

Muthu Chidambaram, Khashayar Gatmiry, Sitan Chen, Holden Lee, and Jianfeng Lu. What does guidance do? a fine-grained analysis in a simple setting. *Advances in Neural Information Processing Systems*, 37:84968–85005, 2024.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pp. 4171–4186, 2019.

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

J. Dombi. A general class of fuzzy operators, the demorgan class of fuzzy operators and fuzziness measures induced by fuzzy operators. *Fuzzy Sets and Systems*, 8(2):149–163, August 1982. ISSN 0165-0114. doi: 10.1016/0165-0114(82)90005-7.

Peiran Dong, Song Guo, Junxiao Wang, Bingjie Wang, Jiewei Zhang, and Ziming Liu. Towards test-time refusals via concept negation. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

Randal Douc and Olivier Cappé. Comparison of resampling schemes for particle filtering. In *ISPA 2005. Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis, 2005.*, pp. 64–69. Ieee, 2005.

[1] Anonymous Repository

Yilun Du and Leslie Pack Kaelbling. Compositional generative modeling: A single model is not all you need. *CoRR*, abs/2402.01103, 2024. URL https://doi.org/10.48550/arXiv.2402.01103.

Yilun Du, Shuang Li, and Igor Mordatch. Compositional visual generation with energy based models. *Advances in Neural Information Processing Systems*, 33:6637–6647, 2020.

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.

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

Peter Ertl and Ansgar Schuffenhauer. Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions. *Journal of cheminformatics*, 1(1):8, 2009.

Timur Garipov, Sebastiaan De Peuter, Ge Yang, Vikas Garg, Samuel Kaski, and Tommi Jaakkola. Compositional Sculpting of Iterative Generative Processes. *Advances in neural information processing systems*, 36:12665–12702, 2023.

Sachit Gaudi, Gautam Sreekumar, and Vishnu Boddeti. Coind: Enabling logical compositions in diffusion models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=cCRlEvjrx4.

Antonio Ginart, Melody Guan, Gregory Valiant, and James Y Zou. Making ai forget you: Data deletion in machine learning. *Advances in neural information processing systems*, 32, 2019.

Jiaqi Guan, Wesley Wei Qian, Xingang Peng, Yufeng Su, Jian Peng, and Jianzhu Ma. 3d equivariant diffusion for target-aware molecule generation and affinity prediction. *arXiv preprint arXiv:2303.03543*, 2023.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International conference on machine learning*, pp. 1321–1330. PMLR, 2017.

Geoffrey E Hinton. Products of experts. In *Proceedings of the Ninth International Conference on Artificial Neural Networks*, volume 1, pp. 1–6. IET, 1999.

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. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. URL https://openreview.net/forum?id=qw8AKxfYbI.

Michael I Jordan and Robert A Jacobs. Hierarchical mixtures of experts and the EM algorithm. *Neural Computation*, 6(2):181–214, 1994.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

Rafal Karczewski, Markus Heinonen, and Vikas Garg. Diffusion models as cartoonists: The curious case of high density regions. In *The Thirteenth International Conference on Learning Representations*, 2025a. URL https://openreview.net/forum?id=RiS2cxpENN.

Rafal Karczewski, Markus Heinonen, and Vikas K Garg. Devil is in the details: Density guidance for detail-aware generation with flow models. In *Forty-second International Conference on Machine Learning (ICML)*, 2025b. URL https://openreview.net/forum?id=C8pGYyfhoF.

Michael Kirchhof, James Thornton, Louis Béthune, Pierre Ablin, Eugene Ndiaye, and Marco Cuturi. Shielded Diffusion: Generating Novel and Diverse Images using Sparse Repellency, May 2025.

Erich Peter Klement, Radko Mesiar, and Endre Pap. *Triangular norms*, volume 8. Springer Science & Business Media, 2013.

Yann LeCun. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*, 1998.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33: 9459–9474, 2020.

Nan Liu, Shuang Li, Yilun Du, Antonio Torralba, and Joshua B. Tenenbaum. Compositional Visual Generation with Composable Diffusion Models. In Shai Avidan, Gabriel Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner (eds.), *Computer Vision – ECCV 2022*, pp. 423–439, Cham, 2022. Springer Nature Switzerland. ISBN 978-3-031-19790-1. doi: 10.1007/978-3-031-19790-1_26.

Christian A Naesseth, Fredrik Lindsten, Thomas B Schön, et al. Elements of sequential monte carlo. *Foundations and Trends® in Machine Learning*, 12(3):307–392, 2019.

Thanh Tam Nguyen, Thanh Trung Huynh, Zhao Ren, Phi Le Nguyen, Alan Wee-Chung Liew, Hongzhi Yin, and Quoc Viet Hung Nguyen. A survey of machine unlearning. *ACM Transactions on Intelligent Systems and Technology*, 2022.

Tomohiro Nishiyama and Igal Sason. On relations between the relative entropy and chi-square-divergence, generalizations and applications. *Entropy*, 22(5):563, May 2020. ISSN 1099-4300. doi: 10.3390/e22050563. URL http://dx.doi.org/10.3390/e22050563.

Bernt Øksendal. Stochastic differential equations. In *Stochastic differential equations: an introduction with applications*, pp. 38–50. Springer, 2003.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35: 27730–27744, 2022.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Marta Skreta, Tara Akhound-Sadegh, Viktor Ohanesian, Roberto Bondesan, Alán Aspuru-Guzik, Arnaud Doucet, Rob Brekelmans, Alexander Tong, and Kirill Neklyudov. Feynman-Kac Correctors in Diffusion: Annealing, Guidance, and Product of Experts, June 2025a.

Marta Skreta, Lazar Atanackovic, Avishek Joey Bose, Alexander Tong, and Kirill Neklyudov. The Superposition of Diffusion Models Using the Itô Density Estimator, February 2025b.

James Thornton, Louis Bethune, Ruixiang Zhang, Arwen Bradley, Preetum Nakkiran, and Shuangfei Zhai. Composition and Control with Distilled Energy Diffusion Models and Sequential Monte Carlo, February 2025.

Ramakrishna Vedantam, Ian Fischer, Jonathan Huang, and Kevin Murphy. Generative models of visually grounded imagination. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=HkCsm6lRb.

Pablo Villalobos, Jaime Sevilla, Lennart Heim, Tamay Besiroglu, Marius Hobbhahn, and Anson Ho. Will we run out of data? an analysis of the limits of scaling datasets in machine learning. *arXiv preprint arXiv:2211.04325*, 1:1, 2022.

Wenhao Wang, Yifan Sun, Zongxin Yang, Zhengdong Hu, Zhentao Tan, and Yi Yang. Replication in visual diffusion models: A survey and outlook. *arXiv preprint arXiv:2408.00001*, 2024.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*, 2022.

Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation. *Advances in Neural Information Processing Systems*, 36:15903–15935, 2023.

Seniha Esen Yuksel, Joseph N. Wilson, and Paul D. Gader. Twenty Years of Mixture of Experts. *IEEE Transactions on Neural Networks and Learning Systems*, 23(8):1177–1193, August 2012. ISSN 2162-2388. doi: 10.1109/TNNLS.2012.2200299.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 3836–3847, 2023.

A FUZZY LOGIC OPERATORS

In this section, we define the class of DeMorgan dual density and score operators, and investigate one example, the Dombi operators, in detail. We show that they generalize probabilistic mixtures and the harmonic mean, and discuss methods to stabilize explicitly used negations with these operators. We first extend the definition of fuzzy logic operators to the domain of probability densities.

Definition A.1 (DeMorgan Density Operators)**.** *Let $\phi : [0, \infty] \to [0, 1]$ be an order-isomorphism and $f : [0, 1] \to [0, \infty]$ be a continuous, strictly decreasing function with $f(0) = \infty$. For $g = f \circ \phi$, we define*

$$\neg p(\mathbf{x}) \coloneqq \phi^{-1}(1 - \phi(p(\mathbf{x}))) \tag{16}$$

$$p_1(\mathbf{x}) \wedge p_2(\mathbf{x}) \coloneqq g^{-1}(g(p_1(\mathbf{x})) + g(p_2(\mathbf{x}))) \tag{17}$$

$$p_1(\mathbf{x}) \vee p_2(\mathbf{x}) \coloneqq \neg(\neg p_1(\mathbf{x}) \wedge \neg p_2(\mathbf{x})) \tag{18}$$

For differentiable f and ϕ, the application to scores follows directly:

Proposition A.2 (DeMorgan score calculus)**.** *Let ϕ and f be fully differentiable functions that generate the DeMorgan density operators $\{\wedge, \vee, \neg\}$. Then with $g = f \circ \phi$, $h : x \mapsto f(1 - \phi(x))$, $w(x) := x\, g'(x)$ and $\overline{w}(x) := x\, h'(x)$ the corresponding operations on the energies and scores are defined as*

$$\neg s(\mathbf{x}) = -\frac{\phi'(p(\mathbf{x}))p(\mathbf{x})}{\phi'(\neg p(\mathbf{x}))\neg p(\mathbf{x})}\, s(\mathbf{x}) \tag{19}$$

$$s_1(\mathbf{x}) \wedge s_2(\mathbf{x}) = \frac{w(p_1(\mathbf{x}))s_1(\mathbf{x}) + w(p_2(\mathbf{x}))s_2(\mathbf{x})}{w(p_1(\mathbf{x}) \wedge p_2(\mathbf{x}))} \tag{20}$$

$$s_1(\mathbf{x}) \vee s_2(\mathbf{x}) = \frac{\overline{w}(p_1(\mathbf{x}))s_1(\mathbf{x}) + \overline{w}(p_2(\mathbf{x}))s_2(\mathbf{x})}{\overline{w}(p_1(\mathbf{x}) \vee p_2(\mathbf{x}))}. \tag{21}$$

Proof. See Appendix B. □

This result shows that score operations are, in essence, just responsibility-weighted combinations of the component scores. It is then easy to see that bounds on $\frac{w(p_1(\mathbf{x}))+w(p_2(\mathbf{x}))}{w(p_1(\mathbf{x}) \circ p_2(\mathbf{x}))}$ for $\circ \in \{\wedge, \vee\}$ can serve as stability guarantees on ours operators.

A.1 DERIVATION OF DOMBI OPERATORS

We now define the dombi operators with $\phi_c(x) = \frac{x}{x+c} = \frac{1}{\frac{c}{x}+1}$ and $f_\lambda(x) = \left(\frac{1}{x} - 1\right)^\lambda$, and derive their corresponding score calculus here. First, we can see here that $\phi_c^{-1}(x) = \frac{cx}{1-x} = \frac{c}{\frac{1}{x}-1}$, $g(x) = f_\lambda(\phi_c(x)) = \left(\frac{c}{x}\right)^\lambda$, $h(x) = f_\lambda(1 - \phi_c(x)) = f_\lambda\left(\frac{c}{x+c}\right) = f_\lambda\left(\frac{1}{\frac{x}{c}+1}\right) = \left(\frac{x}{c}\right)^\lambda$. Further $g^{-1}(x) =$

$cx^{-1/\lambda}$. With this we can derive Definition 4.1 as:

$$\neg p(\mathbf{x}) = \phi_c^{-1}(1-\phi_c(p(\mathbf{x}))) = \phi_c^{-1}\left(\frac{c(\mathbf{x})}{c(\mathbf{x})+p(\mathbf{x})}\right) = \frac{c(\mathbf{x})^2}{p(\mathbf{x})} \tag{22}$$

$$p_1(\mathbf{x}) \wedge_\lambda p_2(\mathbf{x}) \coloneqq g^{-1}(g(p_1(\mathbf{x})) + g(p_2(\mathbf{x}))) \tag{23}$$

$$= c(\mathbf{x})\left(\left(\frac{c(\mathbf{x})}{p_1(\mathbf{x})}\right)^\lambda + \left(\frac{c(\mathbf{x})}{p_2(\mathbf{x})}\right)^\lambda\right)^{-1/\lambda} \tag{24}$$

$$= \left(\left(\frac{1}{p_1(\mathbf{x})}\right)^\lambda + \left(\frac{1}{p_2(\mathbf{x})}\right)^\lambda\right)^{-1/\lambda} \tag{25}$$

$$= \left(p_1(\mathbf{x})^{-\lambda} + p_2(\mathbf{x})^{-\lambda}\right)^{-1/\lambda} \tag{26}$$

$$p_1(\mathbf{x}) \vee_\lambda p_2(\mathbf{x}) \coloneqq \neg_c(\neg_c p_1(\mathbf{x}) \wedge_\lambda \neg_c p_2(\mathbf{x})) \tag{27}$$

$$= \frac{c(\mathbf{x})^2}{\frac{c(\mathbf{x})^2}{p_1(\mathbf{x})} \wedge_\lambda \frac{c(\mathbf{x})^2}{p_2(\mathbf{x})}} \tag{28}$$

$$= \frac{1}{\frac{1}{p_1(\mathbf{x})} \wedge_\lambda \frac{1}{p_2(\mathbf{x})}} \tag{29}$$

$$= \frac{1}{(p_1(\mathbf{x})^\lambda + p_2(\mathbf{x})^\lambda)^{-1/\lambda}} \tag{30}$$

$$= \left(p_1(\mathbf{x})^\lambda + p_2(\mathbf{x})^\lambda\right)^{1/\lambda} \tag{31}$$

In log-likelihoods and scores, the negation is straightforward. For a power-mixture $\left(p_1(\mathbf{x})^\lambda + p_2(\mathbf{x})^\lambda\right)^{1/\lambda}$, the log-likelihood and score operations are familiar. We investigate disjunction and conjunction at the same time and state for all $\lambda \neq 0$:

$$q(\mathbf{x}) = \left(p_1(\mathbf{x})^\lambda + p_2(\mathbf{x})^\lambda\right)^{1/\lambda} \implies \tag{32}$$

$$\log q(\mathbf{x}) = \frac{1}{\lambda}\log\left(p_1(\mathbf{x})^\lambda + p_2(\mathbf{x})^\lambda\right) \tag{33}$$

$$\frac{1}{\lambda}\log\left(\exp(\lambda\log p_1(\mathbf{x})) + \exp(\lambda\log p_2(\mathbf{x}))\right) \tag{34}$$

$$\frac{1}{\lambda}\text{LogSumExp}(\lambda\log p_1(\mathbf{x}), \lambda\log p_2(\mathbf{x})) \implies \tag{35}$$

$$\nabla_{\mathbf{x}}\log q(\mathbf{x}) = \sum_{i\in\{1,2\}}\left(\text{softmax}_i(\lambda\log p_1(\mathbf{x}), \lambda\log p_2(\mathbf{x}))\nabla_{\mathbf{x}}\log p_i(\mathbf{x})\right) \tag{36}$$

$$= \sum_{i\in\{1,2\}}\left(\frac{p_i(\mathbf{x})^\lambda}{p_1(\mathbf{x})^\lambda + p_2(\mathbf{x})^\lambda}\nabla_{\mathbf{x}}\log p_i(\mathbf{x})\right) \tag{37}$$

In terms of score calculus, or Dombi Operators, end up being softmax-weighted, convex combinations of the component scores.

A.2 Dombi Error Bounds

For a given value of λ, the maximal difference between the Dombi operators and the $\min/\max$ functions can be easily bounded as an additive term in log-likelihood:

Proposition A.3. *Let $\wedge_\lambda, \vee_\lambda$ be the Dombi density operators. Then it holds that*

$$\forall x, y \in \mathbb{R}_{\geq 0}: \quad \min\{x,y\}2^{-1/\lambda} \leq x \wedge_\lambda y \leq \min\{x,y\} \tag{38}$$

$$\forall x, y \in \mathbb{R}_{\geq 0}: \quad \max\{x,y\} \leq x \vee_\lambda y \leq \max\{x,y\}2^{1/\lambda} \tag{39}$$

Proof. See Appendix B □

B PROOFS

Proposition A.2 (DeMorgan score calculus)**.** *Let ϕ and f be fully differentiable functions that generate the DeMorgan density operators $\{\wedge, \vee, \neg\}$. Then with $g = f \circ \phi$, $h : x \mapsto f(1-\phi(x))$, $w(x) := x\, g'(x)$ and $\overline{w}(x) := x\, h'(x)$ the corresponding operations on the energies and scores are defined as*

$$\neg s(\mathbf{x}) = -\frac{\phi'(p(\mathbf{x}))p(\mathbf{x})}{\phi'(\neg p(\mathbf{x}))\neg p(\mathbf{x})}\, s(\mathbf{x}) \tag{19}$$

$$s_1(\mathbf{x}) \wedge s_2(\mathbf{x}) = \frac{w(p_1(\mathbf{x}))s_1(\mathbf{x}) + w(p_2(\mathbf{x}))s_2(\mathbf{x})}{w(p_1(\mathbf{x}) \wedge p_2(\mathbf{x}))} \tag{20}$$

$$s_1(\mathbf{x}) \vee s_2(\mathbf{x}) = \frac{\overline{w}(p_1(\mathbf{x}))s_1(\mathbf{x}) + \overline{w}(p_2(\mathbf{x}))s_2(\mathbf{x})}{\overline{w}(p_1(\mathbf{x}) \vee p_2(\mathbf{x}))}. \tag{21}$$

Proof. $\neg$

$$\begin{aligned}
\neg s_1(\mathbf{x}) &= \nabla_{\mathbf{x}} \log \neg p(\mathbf{x}) && (40)\\
&= \frac{\nabla_{\mathbf{x}} \neg p(\mathbf{x})}{\neg p(\mathbf{x})} && (41)\\
&= \frac{\nabla_{\mathbf{x}} \phi^{-1}(1-\phi(p(\mathbf{x})))}{\phi^{-1}(1-\phi(p(\mathbf{x})))} && (42)\\
&= \frac{\nabla_{\mathbf{x}}(1-\phi(p(\mathbf{x})))}{\phi'(\phi^{-1}(1-\phi(p(\mathbf{x}))))\, \phi^{-1}(1-\phi(p(\mathbf{x})))} && (43)\\
&= \frac{-\phi'(p(\mathbf{x}))p(\mathbf{x})}{\phi'(\phi^{-1}(1-\phi(p(\mathbf{x}))))\, \phi^{-1}(1-\phi(p(\mathbf{x})))}\, s(\mathbf{x}) && (44)\\
&= \frac{-\phi'(p(\mathbf{x}))p(\mathbf{x})}{\phi'(\neg p(\mathbf{x}))\, \neg p(\mathbf{x})}\, s(\mathbf{x}) && (45)
\end{aligned}$$

$\wedge$

$$\begin{aligned}
s_1(\mathbf{x}) \wedge s_2(\mathbf{x}) &= \nabla_{\mathbf{x}} \log(p_1(\mathbf{x}) \wedge p_2(\mathbf{x})) && (46)\\
&= \frac{\nabla_{\mathbf{x}}(p_1(\mathbf{x}) \wedge p_2(\mathbf{x}))}{p_1(\mathbf{x}) \wedge p_2(\mathbf{x})} && (47)\\
&= \frac{\nabla_{\mathbf{x}} g^{-1}\big(g(p_1(\mathbf{x})) + g(p_2(\mathbf{x}))\big)}{p_1(\mathbf{x}) \wedge p_2(\mathbf{x})} && (48)\\
&= \frac{g'(p_1(\mathbf{x}))p_1(\mathbf{x})s_1(\mathbf{x}) + g'(p_2(\mathbf{x}))p_2(\mathbf{x})s_2(\mathbf{x})}{g'(p_1(\mathbf{x}) \wedge p_2(\mathbf{x}))\,\big(p_1(\mathbf{x}) \wedge p_2(\mathbf{x})\big)} && (49)
\end{aligned}$$

$\vee$ Symmetric derivation with h instead of g.

We note that, if we can relate the ratios of the weights, we can give upper *and* lower bounds on the norm of the scores of compositions. ☐

Proposition A.3. *Let $\wedge_\lambda, \vee_\lambda$ be the Dombi density operators. Then it holds that*

$$\forall x, y \in \mathbb{R}_{\geq 0}: \quad \min\{x, y\}2^{-1/\lambda} \leq x \wedge_\lambda y \leq \min\{x, y\} \tag{38}$$

$$\forall x, y \in \mathbb{R}_{\geq 0}: \quad \max\{x, y\} \leq x \vee_\lambda y \leq \max\{x, y\}2^{1/\lambda} \tag{39}$$

Proof. We show the case for $p \vee_\lambda q = \big(p^\lambda + q^\lambda\big)^{1/\lambda}$ first. The definition of $\vee_\lambda$ is equivalent to that of a P-norm over two components. We have the standard inequality (w.l.o.g. for $p \geq q$)

$$p \vee_\lambda q = \big(p^\lambda + q^\lambda\big)^{1/\lambda} \leq \big(2p^\lambda\big)^{1/\lambda} = 2^{1/\lambda} \max\{p, q\} \tag{50}$$

The lower bound similarly follows from

$$p \vee_\lambda q = \big(p^\lambda + q^\lambda\big)^{1/\lambda} \geq \big(p^\lambda\big)^{1/\lambda} = \max\{p, q\} \tag{51}$$

For $\wedge_\lambda$, we can use DeMorgan to obtain the symmetric bounds. We can note that the upper bound is tight for $p = q$ and the lower bound is tight for $q = 0$. □

Proposition 5.2 (Mixture Stability)**.** *Let $\alpha_t = softmax_1(\lambda \log p^1, \lambda \log p^2)$, for a dombi composition $p^1 \circ_\lambda p^2$. Then it holds for the scores s_t^1, s_t^2*

$$|\mathbb{E}[d\alpha_t \mid \mathbf{x}_t]| \leq \frac{\sigma_t^2}{8} \|\lambda s^1 - \lambda s^2\| (\|s^1\| + \|s^2\| + \frac{1}{2}\|\lambda s^1 - \lambda s^2\|) dt \tag{13}$$

Proof. First, we can show easily that $|\frac{\lambda}{4} d(\log p^1 - \log p^2)| + \frac{\lambda\sqrt{3}}{36} d[\log p^1 - \log p^2]$.

$$\alpha = \text{softmax}_1(\lambda \log p^1, \lambda \log p^2) \tag{52}$$
$$= \text{sigmoid}(\lambda \log p^1 - \lambda \log p^2) \tag{53}$$

Now, by Itô's Lemma we have, for $\phi = \text{sigmoid}(\lambda \log p^1 - \lambda \log p^2)$

$$d\alpha = \phi(1-\phi)\lambda d(\log p^1 - \log p^2) + \frac{1}{2}\phi'' \lambda^2 d[\log p^1 - \log p^2] \tag{54}$$

We know that, as ϕ is sigmoid, we can bound its derivative with $\frac{1}{4}$, and second derviative with $\frac{\sqrt{3}}{18}$.

$$|d\alpha| \leq |\frac{\lambda}{4} d(\log p^1 - \log p^2)| + \frac{\lambda\sqrt{3}}{36} d[\log p^1 - \log p^2] \tag{55}$$

Now, we derive a bound for $|\mathbb{E}[d \log p_t^1 - d \log p_t^2 \mid x_\tau]|$ using Equation (3), defining $\ell = \log p_t^1 - \log p_t^2$, $s = \alpha s^1 + (1-\alpha) s^2$ and $u_t(\mathbf{x}) = -f_t(\mathbf{x}) + \frac{\sigma^2}{2} s_t(\mathbf{x})$.

We then have

$$d\ell_t = \langle s_t^1 - s_t^2, u_t \rangle dt + \langle s_t^1 - s_t^2, f_t \rangle dt - \frac{\sigma_t^2}{2}\left(\|s_t^1\|^2 - \|s_t^2\|^2\right) dt + \sigma_t \langle s^1 - s^2, d\overline{\mathbf{w}} \rangle \tag{56}$$
$$= \frac{\sigma_t^2}{2} \langle s_t^1 - s_t^2, s_t - (s_t^1 + s_t^2) \rangle dt + \sigma_t \langle s^1 - s^2, d\overline{\mathbf{w}} \rangle \tag{57}$$

If we condition on $\mathbf{x}_t$, the stochastic part vanishes in expectation, we are left with

$$d\ell = \frac{\sigma_t^2}{2} \langle s_t^1 - s_t^2, s_t - (s_t^1 + s_t^2) \rangle dt \tag{58}$$
$$\leq \frac{\sigma_t^2}{2} \|s^1 - s^2\| \|s_t - (s_t^1 + s_t^2)\| dt \tag{59}$$
$$\leq \frac{\sigma_t^2}{2} \|s^1 - s^2\| \|-((1-\alpha)s_t^1 + \alpha s_t^2)\| dt \tag{60}$$
$$\leq \frac{\sigma_t^2}{2} \|s^1 - s^2\| \frac{1}{2} (\|s^1 + s^2\| + \|s^1 - s^2\|) dt \tag{61}$$
$$\leq \frac{\sigma_t^2}{2} \|s^1 - s^2\| (\|s^1\| + \|s^2\|) dt \tag{62}$$
$$\tag{63}$$

Furthermore, we have

$$d[\ell]_t = \sigma_t^2 \|s_t^1 - s_t^2\|^2 dt \tag{64}$$
$$\mathbb{E}[d[\ell]_t | \mathbf{x}_t] = \sigma^2 \|s_t^1 - s_t^2\|^2 dt \tag{65}$$

Finally, we have

$$|d\alpha| \leq |\frac{\lambda}{4}(d\log p^1 - d\log p^2)| + \frac{\lambda^2\sqrt{3}}{36} d[\log p^1 - \log p^2] \tag{66}$$

$$|\mathbb{E}[d\alpha \mid \mathbf{x}_t] \leq \left|\frac{\lambda}{4}\mathbb{E}[d\ell \mid \mathbf{x}_t] + \frac{\lambda^2\sqrt{3}}{36}\mathbb{E}[d[\ell]_t|\mathbf{x}_t]\right| \tag{67}$$

$$\leq |\frac{\lambda\sigma_t^2}{8}\|s^1 - s^2\|(\|s^1\| + \|s^2\|)dt + \frac{\lambda^2\sigma^2\sqrt{3}}{36}\|s^1 - s^2\|^2 dt| \tag{68}$$

$$\leq \frac{\sigma_t^2}{8}\|\lambda s^1 - \lambda s^2\|(\|s^1\| + \|s^2\| + \frac{1}{2}\|\lambda s^1 - \lambda s^2\|)dt \tag{69}$$

□

B.1 Feynman-Kac Correction

The reweighting equation

$$dw_t = \bar{g}(\mathbf{x})dt \Longrightarrow \frac{\partial p_t(\mathbf{x})}{\partial t} = \bar{g}_t(\mathbf{x})p_t(\mathbf{x}) \tag{70}$$

describes how the log-weight-field influences the marginals of the weighted SDE. The translation of continuity (drift) terms and diffusion terms into log-weights is then given by the following schemes:

$$\begin{aligned}\frac{\partial p_t(\mathbf{x})}{\partial t} &= -\langle\nabla, p_t(\mathbf{x})v_t(\mathbf{x})\rangle = \left(\frac{-1}{p_t(\mathbf{x})}\langle\nabla, p_t(\mathbf{x})v_t(\mathbf{x})\rangle\right)p_t(\mathbf{x}) \Longrightarrow \\ dw_t &= (-\langle\nabla, v_t(\mathbf{x})\rangle - \langle\nabla\log p_t(\mathbf{x}), v_t(\mathbf{x})\rangle)\end{aligned} \tag{71}$$

for drift terms and

$$\begin{aligned}\frac{\partial p_t(\mathbf{x})}{\partial t} &= \frac{\sigma^2}{2}\Delta p_t(\mathbf{x}) = \frac{\sigma^2}{2}p_t(\mathbf{x})\left(\Delta\log p_t(\mathbf{x}) + \|\nabla\log p_t(\mathbf{x})\|^2\right) \Longrightarrow \\ dw_t &= \frac{\sigma^2}{2}\left(\Delta\log p_t(\mathbf{x}) + \|\nabla\log p_t(\mathbf{x})\|^2\right)\end{aligned} \tag{72}$$

for diffusion terms.

Dombi Composition is equivalent to applying a power-norm to probability distributions. We recast this as annealing, a case shown by Skreta et al. (2025a), then taking an (unweighted) mixture and then inverse annealing of the mixture of annealed distributions.

We state the following results before proceeding with the main proofs.

Lemma B.1 (Mixture of SDEs + FKC). *Consider two weighted diffusion models $q_t^1(\mathbf{x}), q_t^2(\mathbf{x})$ defined via the Feynman-Kac equation with corresponding weights $g_t^1(\mathbf{x}), g_t^2(\mathbf{x})$. The weighted SDE corresponding to the sum of the marginals $p_t(\mathbf{x}) \propto q_t^1(\mathbf{x}) + q_t^2(\mathbf{x})$, with $\alpha_t^i = \frac{q_t^i(\mathbf{x})}{q_t^1(\mathbf{x})+q_t^2(\mathbf{x})} \in (0,1)$*

$$\begin{aligned}d\mathbf{x}_t &= \left[-f_t(\mathbf{x}_t) + \sigma_t^2(\alpha_t^1\nabla\log q_t^1(\mathbf{x}_t) + \alpha_t^2\nabla\log q_t^2(\mathbf{x}_t))\right]dt + \sigma_t d\overline{\mathbf{w}}_t \\ dw_t &= \left[\alpha_t^1 g_t^1(\mathbf{x}) + \alpha_t^2 g_t^2(\mathbf{x})\right]dt\end{aligned} \tag{73}$$

Proof. The proof in this case is straightforward.

We have, for $\bar{g}_t(\mathbf{x}) = \alpha_t^1\bar{g}_t^1(\mathbf{x}) + \alpha_t^2\bar{g}_t^2(\mathbf{x})$

$$\frac{\partial p_t}{\partial t} = \frac{\partial q_t^1}{\partial t} + \frac{\partial q_t^2}{\partial t} - \int \frac{\partial q_t^1}{\partial t} + \frac{\partial q_t^2}{\partial t} d\mathbf{x} \tag{74}$$

$$= \langle \nabla, q_t^1(\mathbf{x})(-f_t - \sigma_t^2 \nabla \log q_t^1(\mathbf{x})) \rangle + \frac{\sigma_t^2}{2} \Delta q_t^1(\mathbf{x}) + q_t^1(\mathbf{x}) \left[\bar{g}_t^1(\mathbf{x}) \right] + \langle \nabla, q_t^2(\mathbf{x})(-f_t - \sigma_t^2 \nabla \log q_t^2(\mathbf{x})) \rangle + \frac{\sigma_t^2}{2} \Delta q_t^2(\mathbf{x}) + q_t^2(\mathbf{x}) \left[\bar{g}_t^2(\mathbf{x}) \right] - \int \frac{\partial q_t^1}{\partial t} + \frac{\partial q_t^2}{\partial t} d\mathbf{x} \tag{75}$$

$$= \langle \nabla, q_t^1(\mathbf{x})(-f_t - \sigma_t^2 \frac{1}{q_t^1(\mathbf{x})} \nabla q_t^1(\mathbf{x})) \rangle + \frac{\sigma_t^2}{2} \Delta q_t^1(\mathbf{x}) + q_t^1(\mathbf{x}) \left[\bar{g}_t^1(\mathbf{x}) \right] + \langle \nabla, q_t^2(\mathbf{x})(-f_t - \sigma_t^2 \frac{1}{q_t^2(\mathbf{x})} \nabla q_t^2(\mathbf{x})) \rangle + \frac{\sigma_t^2}{2} \Delta q_t^2(\mathbf{x}) + q_t^2(\mathbf{x}) \left[\bar{g}_t^2(\mathbf{x}) \right] - \int \frac{\partial q_t^1}{\partial t} + \frac{\partial q_t^2}{\partial t} d\mathbf{x} \tag{76}$$

$$= \langle \nabla, q_t^1(\mathbf{x})(-f_t - \sigma_t^2 \frac{1}{q_t^1(\mathbf{x})} \nabla q_t^1(\mathbf{x})) + q_t^2(\mathbf{x})(-f_t - \sigma_t^2 \frac{1}{q_t^2(\mathbf{x})} \nabla q_t^2(\mathbf{x})) \rangle + \frac{\sigma_t^2}{2} \Delta p_t(\mathbf{x}) + p_t(\mathbf{x}) \bar{g}_t(\mathbf{x}) - \int \frac{\partial p_t}{\partial t} d\mathbf{x} \tag{77}$$

$$= \langle \nabla, (q_t^1(\mathbf{x}) + q_t^2(\mathbf{x}))(-f_t) + q_t^1(\mathbf{x})(-\sigma_t^2 \frac{1}{q_t^1(\mathbf{x})} \nabla q_t^1(\mathbf{x})) + q_t^2(\mathbf{x})(-\sigma_t^2 \frac{1}{q_t^2(\mathbf{x})} \nabla q_t^2(\mathbf{x})) \rangle + \frac{\sigma_t^2}{2} \Delta p_t(\mathbf{x}) + p_t(\mathbf{x}) \bar{g}_t(\mathbf{x}) - \int \frac{\partial p_t}{\partial t} d\mathbf{x} \tag{78}$$

$$= \langle \nabla, (q_t^1(\mathbf{x}) + q_t^2(\mathbf{x}))(-f_t) + (-\sigma_t^2 \nabla (q_t^1(\mathbf{x}) + q_t^2(\mathbf{x}))) \rangle + \frac{\sigma_t^2}{2} \Delta p_t(\mathbf{x}) + p_t(\mathbf{x}) \bar{g}_t(\mathbf{x}) - \int \frac{\partial p_t}{\partial t} d\mathbf{x} \tag{79}$$

$$= \langle \nabla, (q_t^1(\mathbf{x}) + q_t^2(\mathbf{x}))(-f_t) + p_t(\mathbf{x}) \left(-\sigma_t^2 \left(\frac{\nabla q_t^1(\mathbf{x})}{p_t(\mathbf{x})} + \frac{\nabla q_t^2(\mathbf{x})}{p_t(\mathbf{x})} \right) \right) \rangle + \frac{\sigma_t^2}{2} \Delta p_t(\mathbf{x}) + p_t(\mathbf{x}) \bar{g}_t(\mathbf{x}) - \int \frac{\partial p_t}{\partial t} d\mathbf{x} \tag{80}$$

$$= \langle \nabla, p_t(\mathbf{x})(-f_t) + p_t(\mathbf{x}) \left(-\sigma_t^2 \left(\frac{\nabla q_t^1(\mathbf{x})}{p_t(\mathbf{x})} + \frac{\nabla q_t^2(\mathbf{x})}{p_t(\mathbf{x})} \right) \right) \rangle + \frac{\sigma_t^2}{2} \Delta p_t(\mathbf{x}) + p_t(\mathbf{x}) \bar{g}_t(\mathbf{x}) - \int \frac{\partial p_t}{\partial t} d\mathbf{x} \tag{81}$$

$$= \langle \nabla, p_t(\mathbf{x}) \left(-f_t - \sigma_t^2 \left(\frac{\nabla q_t^1(\mathbf{x})}{p_t(\mathbf{x})} + \frac{\nabla q_t^2(\mathbf{x})}{p_t(\mathbf{x})} \right) \right) \rangle + \frac{\sigma_t^2}{2} \Delta p_t(\mathbf{x}) + p_t(\mathbf{x}) \bar{g}_t(\mathbf{x}) - \int \frac{\partial p_t}{\partial t} d\mathbf{x} \tag{82}$$

$$= \langle \nabla, p_t(\mathbf{x}) \left(-f_t - \sigma_t^2 \left(\frac{q_t^1(\mathbf{x})}{p_t(\mathbf{x})} \nabla \log q_t^1(\mathbf{x}) + \frac{q_t^2(\mathbf{x})}{p_t(\mathbf{x})} \nabla \log q_t^2(\mathbf{x}) \right) \right) \rangle + \frac{\sigma_t^2}{2} \Delta p_t(\mathbf{x}) + p_t(\mathbf{x}) \bar{g}_t(\mathbf{x}) - \int \frac{\partial p_t}{\partial t} d\mathbf{x} \tag{83}$$

$$= \langle \nabla, p_t(\mathbf{x}) \left(-f_t - \sigma_t^2 \left(\alpha_t^1 \nabla \log q_t^1(\mathbf{x}) + \alpha_t^2 \nabla \log q_t^2(\mathbf{x}) \right) \right) \rangle + \frac{\sigma_t^2}{2} \Delta p_t(\mathbf{x}) + p_t(\mathbf{x}) \bar{g}_t(\mathbf{x}) - \int \frac{\partial p_t}{\partial t} d\mathbf{x} \tag{84}$$

$$= \langle \nabla, p_t(\mathbf{x}) \left(-f_t - \sigma_t^2 \left(\alpha_t^1 \nabla \log q_t^1(\mathbf{x}) + \alpha_t^2 \nabla \log q_t^2(\mathbf{x}) \right) \right) \rangle + \frac{\sigma_t^2}{2} \Delta p_t(\mathbf{x}) + p_t(\mathbf{x}) \bar{g}_t(\mathbf{x}) - 0 \tag{85}$$

We can simulate this as

$$\begin{aligned} d\mathbf{x}_t &= \left[-f_t(\mathbf{x}_t) + \sigma_t^2(\alpha_t^1 \nabla \log q_t^1(\mathbf{x}_t) + \alpha_t^2 \nabla \log q_t^2(\mathbf{x}_t))\right] dt + \sigma_t d\overline{\mathbf{w}}_t \\ dw_t &= \left[\alpha_t^1 g_t^1(\mathbf{x}) + \alpha_t^2 g_t^2(\mathbf{x})\right] dt \end{aligned} \tag{86}$$

□

Lemma B.2 (Target Score Annealed SDE + FKC, Skreta et al., 2025a)**.** *Consider a diffusion model $q_t(\mathbf{x})$ defined via the Feynman-Kac equation with the weight-field $g_t(\mathbf{x})$ and some parameter $\lambda \in \mathbb{R} \setminus \{0\}$. The weighted SDE corresponding to the annealed marginals $p_t(\mathbf{x}) \propto q_t(\mathbf{x})^\lambda$ can be performed by simulating the following weighted SDE*

$$\begin{aligned} d\mathbf{x}_t &= \left[-f_t(\mathbf{x}_t) + \sigma_t^2 \lambda \nabla \log q_t(\mathbf{x}_t)\right] dt + \sigma_t d\overline{\mathbf{w}}_t \\ dw_t &= \left[(\lambda - 1)\langle \nabla, f_t(\mathbf{x})\rangle + \lambda(\lambda - 1)\frac{\sigma_t^2}{2}\|\nabla \log q_t(\mathbf{x})\|^2 + \lambda g(\mathbf{x})\right] dt \end{aligned} \tag{87}$$

Proof. We follow the proofs of Skreta et al. (2025a).

We aim to find the partial derivative of the density $p_t(\mathbf{x}) = \frac{q_t(\mathbf{x})^\lambda}{\int q_t(\mathbf{x})^\lambda dx}$ over time $\frac{\partial p_t(\mathbf{x})}{\partial t}$, where

$$\frac{\partial q_t(\mathbf{x})}{\partial t} = -\langle \nabla, q_t(\mathbf{x})(-f_t + \sigma_t^2 \nabla \log q_t(\mathbf{x}))\rangle + \frac{\sigma_t^2}{2}\Delta q_t(\mathbf{x}) + q_t(\mathbf{x})\left[\bar{g}_t(\mathbf{x})\right].$$

Then we have

$$\frac{\partial \log q_t(\mathbf{x})}{\partial t} = \frac{1}{q_t(\mathbf{x})}\frac{\partial q_t(\mathbf{x})}{\partial t} \tag{88}$$

$$= -\frac{1}{q_t(\mathbf{x})}\langle \nabla,\ q_t(\mathbf{x})(-f_t + \sigma_t^2 \nabla \log q_t(\mathbf{x}))\rangle + \frac{\sigma_t^2}{2}\frac{\Delta q_t(\mathbf{x})}{q_t(\mathbf{x})} + \bar{g}(\mathbf{x}) \tag{89}$$

$$= -\frac{1}{q_t(\mathbf{x})}\langle \nabla,\ q_t(\mathbf{x})(-f_t + \sigma_t^2 \nabla \log q_t(\mathbf{x}))\rangle + \frac{\sigma_t^2}{2}(\Delta \log q_t + \|\nabla \log q_t\|^2) + \bar{g}(\mathbf{x}) \tag{90}$$

$$\begin{aligned} &= -\langle \nabla, -f_t + \sigma_t^2 \nabla \log q_t\rangle \ - \ \langle -f_t + \sigma_t^2 \nabla \log q_t, \nabla \log q_t\rangle \\ &\quad + \frac{\sigma_t^2}{2}(\Delta \log q_t + \|\nabla \log q_t\|^2) + \bar{g}(\mathbf{x}) \end{aligned} \tag{91}$$

$$\begin{aligned} &= \langle \nabla, f_t\rangle + \langle f_t, \nabla \log q_t\rangle - \sigma_t^2\, \Delta \log q_t - \sigma_t^2\, \|\nabla \log q_t\|^2 \\ &\quad + \frac{\sigma_t^2}{2}(\Delta \log q_t + \|\nabla \log q_t\|^2) + \bar{g}(\mathbf{x}) \end{aligned} \tag{92}$$

$$= \langle \nabla, f_t\rangle + \langle f_t, \nabla \log q_t\rangle - \frac{\sigma_t^2}{2}\left(\Delta \log q_t + \|\nabla \log q_t\|^2\right) + \bar{g}(\mathbf{x}). \tag{93}$$

and can now compute

$$\begin{aligned}
\frac{\partial \log p_t(\mathbf{x})}{\partial t} &= \lambda \frac{\partial \log q_t(\mathbf{x})}{\partial t} - \int \lambda\, p_t(\mathbf{x}) \frac{\partial \log q_t(\mathbf{x})}{\partial t}\, d\mathbf{x} && (94)\\
&= \lambda\Big[\langle \nabla, f_t\rangle + \langle f_t, \nabla \log q_t\rangle - \frac{\sigma_t^2}{2}\big(\Delta \log q_t + \|\nabla \log q_t\|^2\big) + \bar{g}\Big] - \\
&\quad \int \lambda\, p_t(\mathbf{x}) \frac{\partial \log q_t(\mathbf{x})}{\partial t}\, d\mathbf{x} && (95)\\
&= \lambda\langle \nabla, f_t\rangle + \lambda\langle f_t, \nabla \log q_t\rangle - \frac{\lambda\sigma_t^2}{2}\big(\Delta \log q_t + \|\nabla \log q_t\|^2\big) + \lambda\bar{g} - \\
&\quad \int \lambda\, p_t(\mathbf{x}) \frac{\partial \log q_t(\mathbf{x})}{\partial t}\, d\mathbf{x} && (96)\\
&= \langle \nabla, \lambda f_t\rangle + \langle f_t, \nabla \log p_t\rangle - \frac{\lambda\sigma_t^2}{2}\big(\Delta \log q_t + \|\nabla \log q_t\|^2\big) + \lambda\bar{g} - \\
&\quad \int \lambda\, p_t(\mathbf{x}) \frac{\partial \log q_t(\mathbf{x})}{\partial t}\, d\mathbf{x} && (97)\\
&= \langle \nabla, f_t\rangle + \langle f_t, \nabla \log p_t\rangle - (1-\lambda)\langle \nabla, f_t\rangle - \frac{\lambda\sigma_t^2}{2}\big(\Delta \log q_t + \|\nabla \log q_t\|^2\big) + \lambda\bar{g} - \\
&\quad \int p_t\Big[\langle \nabla, f_t\rangle + \langle f_t, \nabla \log p_t\rangle - (1-\lambda)\langle \nabla, f_t\rangle - \frac{\lambda\sigma_t^2}{2}\big(\Delta \log q_t + \|\nabla \log q_t\|^2\big) + \lambda\bar{g}\Big]\, d\mathbf{x} && (98)\\
&= \langle \nabla, f_t\rangle + \langle f_t, \nabla \log p_t\rangle - (1-\lambda)\langle \nabla, f_t\rangle - \frac{\lambda\sigma_t^2}{2}\big(\Delta \log q_t + \|\nabla \log q_t\|^2\big) + \lambda g \\
&\quad - \int p_t\Big[-(1-\lambda)\langle \nabla, f_t\rangle - \frac{\lambda\sigma_t^2}{2}\big(\Delta \log q_t + \|\nabla \log q_t\|^2\big) + \lambda g\Big]\, d\mathbf{x} && (99)\\
&= \langle \nabla, f_t\rangle + \langle f_t, \nabla \log p_t\rangle - (1-\lambda)\langle \nabla, f_t\rangle - \frac{\sigma_t^2}{2}\Delta \log p_t - \frac{\sigma_t^2}{2\lambda}\|\nabla \log p_t\|^2 + \lambda g \\
&\quad - \int p_t\Big[-(1-\lambda)\langle \nabla, f_t\rangle - \frac{\sigma_t^2}{2}\Delta \log p_t - \frac{\sigma_t^2}{2\lambda}\|\nabla \log p_t\|^2 + \lambda g\Big]\, d\mathbf{x} && (100)\\
&= \langle \nabla, f_t\rangle + \langle f_t, \nabla \log p_t\rangle - (1-\lambda)\langle \nabla, f_t\rangle - \frac{\sigma_t^2}{2}\Delta \log p_t - \frac{\sigma_t^2}{2}\|\nabla \log p_t\|^2 + \\
&\quad \Big(1 - \frac{1}{\lambda}\Big)\frac{\sigma_t^2}{2}\|\nabla \log p_t\|^2 + \lambda g \; - \int p_t\Big[-(1-\lambda)\langle \nabla, f_t\rangle - \frac{\sigma_t^2}{2}\Delta \log p_t - \frac{\sigma_t^2}{2}\|\nabla \log p_t\|^2 + \\
&\quad \Big(1 - \frac{1}{\lambda}\Big)\frac{\sigma_t^2}{2}\|\nabla \log p_t\|^2 + \lambda g\Big] d\mathbf{x} && (101)\\
&= \langle \nabla, f_t\rangle + \langle f_t, \nabla \log p_t\rangle - (1-\lambda)\langle \nabla, f_t\rangle - \frac{\sigma_t^2}{2}\Delta \log p_t - \frac{\sigma_t^2}{2}\|\nabla \log p_t\|^2 + \\
&\quad \Big(1 - \frac{1}{\lambda}\Big)\frac{\sigma_t^2}{2}\|\nabla \log p_t\|^2 + \lambda g \; - \int p_t\Big[-(1-\lambda)\langle \nabla, f_t\rangle + \\
&\quad \Big(1 - \frac{1}{\lambda}\Big)\frac{\sigma_t^2}{2}\|\nabla \log p_t\|^2 + \lambda g\Big] d\mathbf{x}. && (102)
\end{aligned}$$

With this, defining $g' = -(1-\lambda)\langle \nabla, f_t\rangle + (1 - \frac{1}{\lambda})\frac{\sigma_t^2}{2}\|\nabla \log p_t\|^2 + \lambda g$ we finally have

$$\frac{\partial \log p_t}{\partial t} = \langle \nabla, f_t\rangle + \langle f_t, \nabla \log p_t\rangle - \frac{\sigma_t^2}{2}\Delta \log p_t - \frac{\sigma_t^2}{2}\|\nabla \log p_t\|^2 + g' - \int p_t(\mathbf{x}) g' d\mathbf{x} \quad (103)$$

$$\frac{\partial p_t}{\partial t} = p_t \frac{\partial \log p_t}{\partial t} \tag{104}$$

$$= p_t \left[\langle \nabla, f_t \rangle + \langle f_t, \nabla \log p_t \rangle - \frac{\sigma_t^2}{2} \Delta \log p_t - \frac{\sigma_t^2}{2} \|\nabla \log p_t\|^2 + g'(\mathbf{x}) - \mathbb{E}_{p_t} g'(\mathbf{x}) \right] \tag{105}$$

$$= -\langle \nabla, -f_t p_t \rangle + p_t \left[-\frac{\sigma_t^2}{2} \Delta \log p_t - \frac{\sigma_t^2}{2} \|\nabla \log p_t\|^2 + g'(\mathbf{x}) - \mathbb{E}_{p_t} g'(\mathbf{x}) \right] \tag{106}$$

$$= -\langle \nabla, -f_t p_t \rangle + p_t \left[-\frac{\sigma_t^2}{2} \frac{\Delta p_t}{p_t} + \frac{\sigma_t^2}{2} \|\nabla \log p_t\|^2 - \frac{\sigma_t^2}{2} \|\nabla \log p_t\|^2 + g'(\mathbf{x}) - \mathbb{E}_{p_t} g'(\mathbf{x}) \right] \tag{107}$$

$$= -\langle \nabla, p_t(-f_t + \sigma_t^2 \nabla \log p_t) \rangle + \frac{\sigma_t^2}{2} \Delta p_t + p_t \left[g'(\mathbf{x}) - \mathbb{E}_{p_t} g'(\mathbf{x}) \right] \tag{108}$$

And finally, we can reexpress this as

$$\frac{\partial p_t}{\partial t} = -\langle \nabla, p_t(-f_t + \sigma^2 \lambda \nabla \log q_t) \rangle + \frac{\sigma_t^2}{2} \Delta p_t + p_t \left[g'(\mathbf{x}) - \mathbb{E}_{p_t} g'(\mathbf{x}) \right] \tag{109}$$

And for $\lambda > 0$ we can simulate this as

$$\begin{aligned} d\mathbf{x}_t &= \left[-f_t(\mathbf{x}_t) + \sigma_t^2 \lambda \nabla \log q_t(\mathbf{x}_t) \right] dt + \sigma_t d\overline{\mathbf{w}}_t \\ dw_t &= g_t'(\mathbf{x}) dt = \left[-(1-\lambda) \langle \nabla, f_t(\mathbf{x}) \rangle + \lambda(\lambda - 1) \frac{\sigma_t^2}{2} \|\nabla \log q_t\|^2 + \lambda g \right] dt \end{aligned} \tag{110}$$

□

Proposition 6.1 (Referenced Negation as CFG+FKC, Skreta et al., 2025a)**.** *Consider two diffusion models $q_t^1(\mathbf{x}), q_t^2(\mathbf{x})$ defined via the Fokker-Planck equation in Equation (2). The weighted SDE corresponding to the referenced negation of $p_t(\mathbf{x}) \propto \neg_{q_t^2(\mathbf{x})} q_t^1(\mathbf{x})$ is, with $dw_t(\mathbf{x}) = g_t(\mathbf{x})dt$*

$$\begin{aligned} d\mathbf{x}_t &= \left[-f_t(\mathbf{x}_t) + \sigma_t^2 (2\nabla \log q_t^2(\mathbf{x}_t) - \nabla \log q_t^1(\mathbf{x}_t)) \right] dt + \sigma_t d\overline{\mathbf{w}}_t \\ g_t(\mathbf{x}) &= \sigma_t^2 \|\nabla \log q_t^1(\mathbf{x}_t) - \nabla \log q_t^2(\mathbf{x}_t)\|^2 + 2 g_t^2(\mathbf{x}) - g_t^1(\mathbf{x}), \end{aligned} \tag{14}$$

Proof. We start with the annealed distribution $q_t^2(\mathbf{x})^2$ and the annealed pseudo-distribution $q_t^1(\mathbf{x})^{-1}$. We now try to find

$$\frac{\partial \log p_t}{\partial t} = 2\frac{\partial \log q_t^2}{\partial t} - \frac{\partial \log q_t^1}{\partial t} - \int p_t \left[2\frac{\partial \log q_t^2}{\partial t} - \frac{\partial \log q_t^1}{\partial t}\right] \tag{111}$$

$$= 2\frac{\partial \log q_t^2}{\partial t} - \frac{\partial \log q_t^1}{\partial t} - \int p_t \left[2\frac{\partial \log q_t^2}{\partial t} - \frac{\partial \log q_t^1}{\partial t}\right] \tag{112}$$

$$\begin{aligned} =&2\left[\langle\nabla, f_t\rangle + \langle f_t, \nabla \log q_t^2\rangle - \frac{\sigma_t^2}{2}\left(\Delta \log q_t^2 + \|\nabla \log q_t^2\|^2\right) + \bar{g}^2(\mathbf{x})\right] - \\ &\left[\langle\nabla, f_t\rangle + \langle f_t, \nabla \log q_t^1\rangle - \frac{\sigma_t^2}{2}\left(\Delta \log q_t^1 + \|\nabla \log q_t^1\|^2\right) + \bar{g}^1(\mathbf{x})\right] - \\ &\int p_t(\mathbf{x}) \left[2\frac{\partial \log q_t^2(\mathbf{x})}{\partial t} - \frac{\partial \log q_t^1(\mathbf{x})}{\partial t}\right] d\mathbf{x} \end{aligned} \tag{113}$$

$$\begin{aligned} =&\langle\nabla, f_t\rangle + \langle f_t, 2\nabla \log q_t^2\rangle - \langle f_t, \nabla \log q_t^1\rangle + 2\left[-\frac{\sigma_t^2}{2}\left(\Delta \log q_t^2 + \|\nabla \log q_t^2\|^2\right) + \bar{g}^2(\mathbf{x})\right] - \\ &\left[-\frac{\sigma_t^2}{2}\left(\Delta \log q_t^1 + \|\nabla \log q_t^1\|^2\right) + \bar{g}^1(\mathbf{x})\right] - \int p_t(\mathbf{x}) \left[2\frac{\partial \log q_t^2(\mathbf{x})}{\partial t} - \frac{\partial \log q_t^1(\mathbf{x})}{\partial t}\right] d\mathbf{x} \end{aligned} \tag{114}$$

$$\begin{aligned} =&\langle\nabla, f_t\rangle + \langle f_t, \nabla \log p_t\rangle - \frac{\sigma_t^2}{2}\left(2(\Delta \log q_t^2 + \|\nabla \log q_t^2\|^2) - (\Delta \log q_t^1 + \|\nabla \log q_t^1\|^2)\right) + \\ &2\bar{g}^2(\mathbf{x}) - \bar{g}^1(\mathbf{x}) - \int p_t(\mathbf{x}) \left[2\frac{\partial \log q_t^2(\mathbf{x})}{\partial t} - \frac{\partial \log q_t^1(\mathbf{x})}{\partial t}\right] d\mathbf{x} \end{aligned} \tag{115}$$

$$\begin{aligned} =&\langle\nabla, f_t\rangle + \langle f_t, \nabla \log p_t\rangle - \frac{\sigma_t^2}{2}\left(\Delta \log p_t + \|\nabla \log p_t\|^2 - 2\|\nabla \log q_t^2 - \nabla \log q_t^1\|^2\right) + \\ &2\bar{g}^2(\mathbf{x}) - \bar{g}^1(\mathbf{x}) - \int p_t(\mathbf{x}) \left[2\frac{\partial \log q_t^2(\mathbf{x})}{\partial t} - \frac{\partial \log q_t^1(\mathbf{x})}{\partial t}\right] d\mathbf{x} \end{aligned} \tag{116}$$

$$\begin{aligned} =&\langle\nabla, f_t\rangle + \langle f_t, \nabla \log p_t\rangle - \frac{\sigma_t^2}{2}\left(\Delta \log p_t + \|\nabla \log p_t\|^2\right) + \sigma_t^2\|\nabla \log q_t^2 - \nabla \log q_t^1\|^2 + \\ &2g^2(\mathbf{x}) - g^1(\mathbf{x}) - \mathbb{E}_{p_t}\left[\sigma_t^2\|\nabla \log q_t^2 - \nabla \log q_t^1\|^2 + 2g^2(\mathbf{x}) - g^1(\mathbf{x})\right] \end{aligned} \tag{117}$$

And with $g(\mathbf{x}) = \sigma_t^2\|\nabla \log q_t^2 - \nabla \log q_t^1\|^2 + 2g^2(\mathbf{x}) - g^1(\mathbf{x})$

$$\frac{\partial p_t}{\partial t} = p_t \frac{\partial \log p_t}{\partial t} \tag{118}$$

$$= p_t \left[\langle\nabla, f_t\rangle + \langle f_t, \nabla \log p_t\rangle - \frac{\sigma_t^2}{2}\left(\Delta \log p_t + \|\nabla \log p_t\|^2\right)\right] + p_t\left[g(\mathbf{x}) - \mathbb{E}_{p_t} g(\mathbf{x})\right] \tag{119}$$

$$= -\langle\nabla, p_t(\mathbf{x})(-f_t + \nabla \log p_t)\rangle + \frac{\sigma_t^2}{2}\Delta p_t + p_t\left[g(\mathbf{x}) - \mathbb{E}_{p_t} g(\mathbf{x})\right], \tag{120}$$

which we can simulate with

$$\begin{aligned} d\mathbf{x}_t &= \left[-f_t(\mathbf{x}_t) + \sigma_t^2(2\nabla \log q_t^2(\mathbf{x}_t) - \nabla \log q_t^1(\mathbf{x}_t))\right] dt + \sigma_t d\overline{\mathbf{w}}_t \\ g_t(\mathbf{x}) &= \sigma_t^2\|\nabla \log q_t^1(\mathbf{x}_t) - \nabla \log q_t^2(\mathbf{x}_t)\|^2 + 2g_t^2(\mathbf{x}) - g_t^1(\mathbf{x}). \end{aligned} \tag{121}$$

□

Theorem 6.2. *Consider two weighted diffusion models $q_t^1(\mathbf{x}), q_t^2(\mathbf{x})$ defined via the Feynman-Kac equation with weights $g_t^1(\mathbf{x}), g_t^2(\mathbf{x})$, and a parameter $\lambda \in \mathbb{R}\backslash\{0\}$. The weighted SDE corresponding to $p_t(\mathbf{x}) \propto \left(q_t^1(\mathbf{x})^\lambda + q_t^2(\mathbf{x})^\lambda\right)^{1/\lambda}$, with $\alpha_t^i = \frac{q_t^i(\mathbf{x})^\lambda}{q_t^1(\mathbf{x})^\lambda + q_t^2(\mathbf{x})^\lambda} \in (0,1)$, and $dw_t = g_t(\mathbf{x})dt$ is*

$$d\mathbf{x}_t = \left[-f_t(\mathbf{x}_t) + \sigma_t^2(\alpha_t^1\nabla \log q_t^1(\mathbf{x}_t) + \alpha_t^2\nabla \log q_t^2(\mathbf{x}_t))\right] dt + \sigma_t d\overline{\mathbf{w}}_t$$

$$g_t(\mathbf{x}) = (1-\lambda)\frac{\sigma^2}{2}\left[\left\|\sum_{i\in\{1,2\}} \alpha_t^i \nabla \log q_t^i(\mathbf{x}_t)\right\|^2 - \sum_{i\in\{1,2\}} \alpha_t^i \|\nabla \log q_t^i(\mathbf{x}_t)\|^2\right] + \sum_{i\in\{1,2\}} \alpha_t^i g_t^i(\mathbf{x}_t). \tag{15}$$

Proof of Theorem 6.2. We now use our two lemmas to show the main result. We begin with

$$\begin{aligned} d\mathbf{x}_t &= \left[-f_t(\mathbf{x}_t) + \sigma_t^2 \lambda \nabla \log q_t^i(\mathbf{x}_t)\right] dt + \sigma_t d\overline{\mathbf{w}}_t \\ dw_t &= (\lambda - 1)\left(\langle \nabla, f_t(\mathbf{x}_t)\rangle + \frac{\sigma^2}{2}\lambda \|\nabla \log q_t^i(\mathbf{x}_t)\|^2\right) dt + \lambda g_t^i(\mathbf{x}) \end{aligned} \tag{122}$$

for both annealed distributions, according to Lemma B.2. Then, by Lemma B.1, we have a mixture of these distributions with

$$\begin{aligned} d\mathbf{x}_t &= \left[-f_t(\mathbf{x}_t) + \sigma_t^2 \lambda (\alpha_t^1 \nabla \log q_t^1(\mathbf{x}_t) + \alpha_t^2 \nabla \log q_t^2(\mathbf{x}_t))\right] dt + \sigma_t d\overline{\mathbf{w}}_t \\ dw_t &= \alpha_t^1 \left[(\lambda - 1)\left(\langle \nabla, f_t(\mathbf{x}_t)\rangle + \frac{\sigma^2}{2}\lambda \|\nabla \log q_t^1(\mathbf{x}_t)\|^2\right) dt + \lambda g_t^1(\mathbf{x})\right] + \\ &\quad \alpha_t^2 \left[(\lambda - 1)\left(\langle \nabla, f_t(\mathbf{x}_t)\rangle + \frac{\sigma^2}{2}\lambda \|\nabla \log q_t^2(\mathbf{x}_t)\|^2\right) dt + \lambda g_t^2(\mathbf{x})\right] \end{aligned} \tag{123}$$

which simplifies to

$$dw_t = (\lambda - 1)\langle \nabla, f_t(\mathbf{x}_t)\rangle dt + \lambda \left[\sum_{i \in \{1,2\}} \alpha_t^i \left((\lambda - 1)\frac{\sigma^2}{2}\|\nabla \log q_t^i(\mathbf{x}_t)\|^2 dt + g_t^i(\mathbf{x}_t)\right)\right]. \tag{124}$$

Finally, we apply Lemma B.2 to the resulting mixture with $1/\lambda$. This then results in

$$d\mathbf{x}_t = \left[-f_t(\mathbf{x}_t) + \sigma_t^2 (\alpha_t^1 \nabla \log q_t^1(\mathbf{x}_t) + \alpha_t^2 \nabla \log q_t^2(\mathbf{x}_t))\right] dt + \sigma_t d\overline{\mathbf{w}}_t, \tag{125}$$

which is the target score as desired. For our weight-field we then have

$$dw_t = \begin{aligned} &(\frac{1}{\lambda} - 1)\left(\langle \nabla, f_t(\mathbf{x}_t)\rangle + \frac{\sigma^2}{2}\frac{1}{\lambda}\|\alpha_t^1 \lambda \nabla \log q_t^1(\mathbf{x}_t) + \alpha_t^2 \lambda \nabla \log q_t^2(\mathbf{x}_t)\|^2\right) dt + \\ &\frac{1}{\lambda}\left[(\lambda - 1)\langle \nabla, f_t(\mathbf{x}_t)\rangle + \lambda \left[\sum_{i \in \{1,2\}} \alpha_t^i \left((\lambda - 1)\frac{\sigma^2}{2}\|\nabla \log q_t^i(\mathbf{x}_t)\|^2 + g_t^i(\mathbf{x}_t)\right)\right]\right] dt \end{aligned} \tag{126}$$

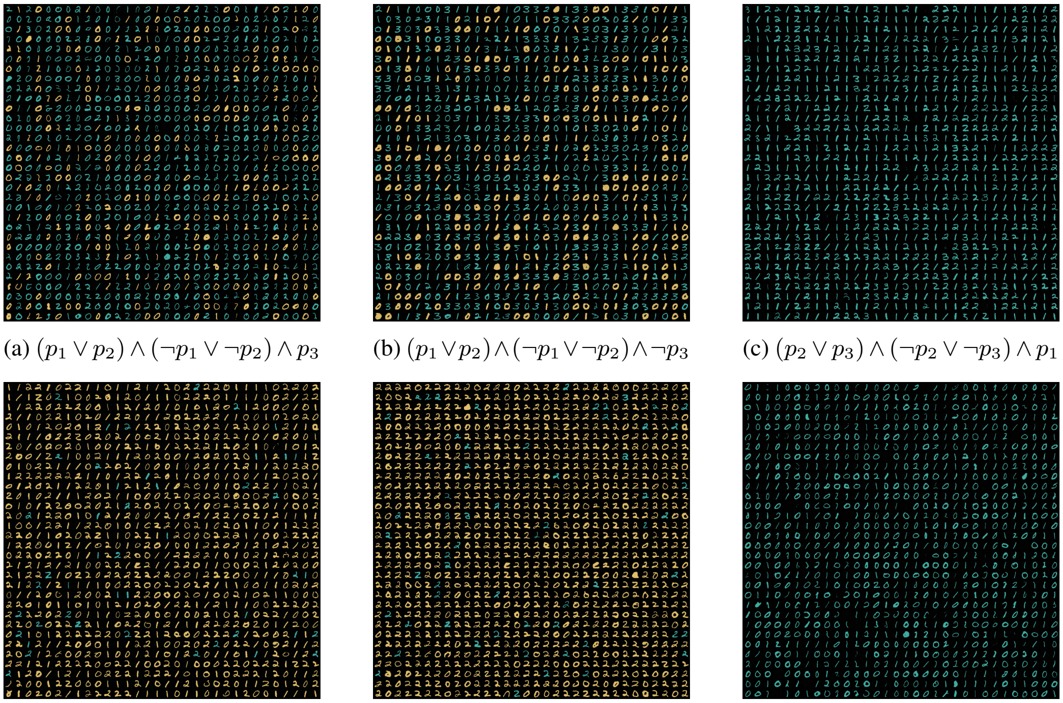
$$= \begin{aligned} &\frac{1 - \lambda}{\lambda}\langle \nabla, f_t(\mathbf{x}_t)\rangle dt + \frac{1 - \lambda}{\lambda}\frac{\sigma^2}{2}\frac{1}{\lambda}\|\alpha_t^1 \lambda \nabla \log q_t^1(\mathbf{x}_t) + \alpha_t^2 \lambda \nabla \log q_t^2(\mathbf{x}_t)\|^2 dt + \\ &\frac{\lambda - 1}{\lambda}\langle \nabla, f_t(\mathbf{x}_t)\rangle dt + \sum_{i \in \{1,2\}} \alpha_t^i \left((\lambda - 1)\frac{\sigma^2}{2}\|\nabla \log q_t^i(\mathbf{x}_t)\|^2 + g_t^i(\mathbf{x}_t)\right) dt \end{aligned} \tag{127}$$

$$= \begin{aligned} &(1 - \lambda)\frac{\sigma^2}{2}\left\|\sum_{i \in \{1,2\}} \alpha_t^i \nabla \log q_t^i(\mathbf{x}_t)\right\|^2 dt + \\ &\sum_{i \in \{1,2\}} \alpha_t^i \left((\lambda - 1)\frac{\sigma^2}{2}\|\nabla \log q_t^i(\mathbf{x}_t)\|^2 + g_t^i(\mathbf{x}_t)\right) dt \end{aligned} \tag{128}$$

$$= (1 - \lambda)\frac{\sigma^2}{2}\left[\left\|\sum_{i \in \{1,2\}} \alpha_t^i \nabla \log q_t^i(\mathbf{x}_t)\right\|^2 - \sum_{i \in \{1,2\}} \alpha_t^i \|\nabla \log q_t^i(\mathbf{x}_t)\|^2\right] dt + \sum_{i \in \{1,2\}} \alpha_t^i g_t^i(\mathbf{x}_t) dt \tag{129}$$

We can see that, as expected, for $\lambda = 1$ we are left with the unweighted mixture of distributions. For more complex compositions, the weight fields just propagate as well, we can see that the statement trivially generalizses to more than two diffusion models, so we maintain associativity.

□



(a) $(p_1 \vee p_2) \wedge (\neg p_1 \vee \neg p_2) \wedge p_3$ (b) $(p_1 \vee p_2) \wedge (\neg p_1 \vee \neg p_2) \wedge \neg p_3$ (c) $(p_2 \vee p_3) \wedge (\neg p_2 \vee \neg p_3) \wedge p_1$

(d) $(p_2 \vee p_3) \wedge (\neg p_2 \vee \neg p_3) \wedge \neg p_1$ (e) $p_3 \wedge \neg p_1 \wedge \neg p_2$ (f) $p_1 \wedge p_2 \wedge p_3$

Figure 6: Generated MNIST score compositions.

C EXPERIMENTS

All our experiments on stable diffusion and SBDD were performed on unmodified, pretrained models. We performed inference on Nvidia v100 and a100 GPUs.

C.1 MNIST EXPERIMENTS

We reproduce the setup of (Garipov et al., 2023), and generate images from the score composition of the three toy mnist models. The code to training the models can be obtained from the code repository and training was performed on a Nvidia GTX 3080 desktop within 10 minutes.

We show image collages for non-trivial example formulas in Figure 6. For each formula we generated a batch of 1024 images.

C.2 STABLE DIFFUSION IMAGE GENERATION

We reproduce the stable diffusion experimental setup of (Skreta et al., 2025b) with Stable Diffusion v1-4 available pretrained publically at huggingface: `https://huggingface.co/CompVis/stable-diffusion-v1-4`. We then report, PoE, superdiffs `and` as well as joint prompts.

We use 20 pairs of conjunctive prompt-pairs and generate 20 images each. We provide a batch of the generated images in the supplementary material. and list the prompts here, also reused from (Skreta et al., 2025b):

- `"a mountain landscape"` $\wedge$ `"silhouette of a dog"`
- `"a flamingo"` $\wedge$ `"a candy cane"`
- `"a dragonfly"` $\wedge$ `"a helicopter"`
- `"dandelion"` $\wedge$ `"fireworks"`
- `"a sunflower"` $\wedge$ `"a lemon"`

- `"a rocket"` ∧ `"a cactus"`
- `"moon"` ∧ `"cookie"`
- `"a snail"` ∧ `"a cinnamon roll"`
- `"an eagle"` ∧ `"an airplane"`
- `"zebra"` ∧ `"barcode"`
- `"chess pawn"` ∧ `"bottle cap"`
- `"a pineapple"` ∧ `"a beehive"`
- `"a spider web"` ∧ `"a bicycle wheel"`
- `"a waffle cone"` ∧ `"a volcano"`
- `"a cat"` ∧ `"a dog"`
- `"a chair"` ∧ `"an avocado"`
- `"a donut"` ∧ `"a map"`
- `"otter"` ∧ `"duck"`
- `"pebbles on a beach"` ∧ `"a turtle"`
- `"teddy bear"` ∧ `"panda"`

For the contrastive Prompts, we partially use our own prompts and partially use the prompts from (Dong et al., 2023). We provide a batch of the generated images in the supplementary material. and list the prompts here:

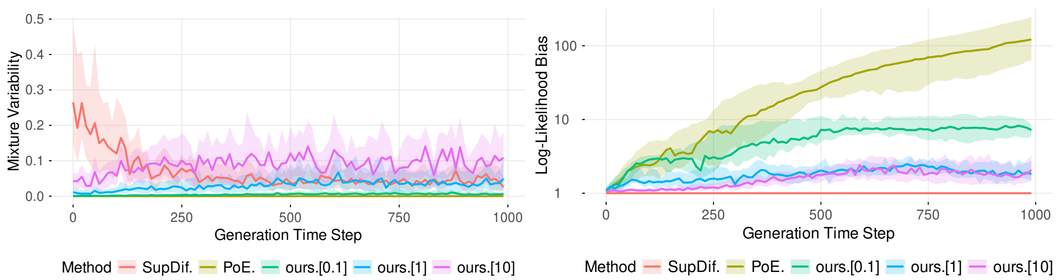
- `"A night sky with stars and a crescent moon, reminiscent of Van Gogh's 'Starry Night'."` ∧¬ `"Van Gogh"`
- `"A night sky with stars and a crescent moon, reminiscent of Van Gogh's 'Starry Night'."` ∧¬ `"Picasso's Cubist style"`
- `"A portrait of a man with a distorted and fragmented face painted in Picasso's Cubist style."` ∧¬ `"Picasso's Cubist style"`
- `"A cat and a ball on the shelf"` ∧¬ `"cat, ball"`
- `"There are a bicycle and a car in front of the house"` ∧¬ `"a bicycle and a car"`
- `"orange fruit"` ∧¬ `"orange color palette"`
- `"a banana"` ∧¬ `"yellow color palette"`
- `"an ocean"` ∧¬ `"blue color palette"`
- `"strawberry"` ∧¬ `"red color palette"`
- `"round shape"` ∧¬ `"circle"`

C.2.1 Additional results

We provide additional plots illustrating the behaviour of composition under varying values of λ in Figure 7.

C.3 Additional Results on SBDD Molecule Generation

We report a sweep across three values of λ for the molecule generation task in Table 2. As the variance in this experiment is high, none of the differences can be considered significant.



(a) Variability of mixture coefficients for conjunction (b) Absolute difference in likelihood during generation

Figure 7: Mixture Stability vs Likelihood Bias in SD experiment. Figure 7a shows the absolute changerate of α^i, Figure 7b shows the median absolute log-density ratio. PoE (or geometric mean) has constant mixture coefficients, but log-likelihoods diverge during the diffusion process. Superdiff forces equal likelihoods as the cost of a highly variable mixture, especially early during the diffusion process. Dombi composition (ours.[λ]) provides a tradeoff, depending on λ.

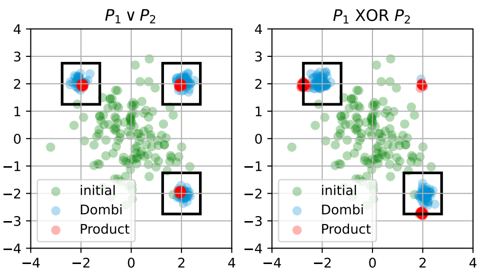
Table 2: Docking Scores of generated ligands for 14 protein target pairs ($\mathtt{P}_1$, $\mathtt{P}_2$), in batches of 32 ligands for 5 molecule lengths each. Extended runs across temperatures $\gamma \in \{1, 2\}$. We compare conjunction with Dombi with various λ with and without FKC with annealed base distribution and also report TargetDiff from (Guan et al., 2023) as baseline.

Method	Temp. γ	λ	FKC?	($\mathtt{P}_1$ * $\mathtt{P}_2$) (↑)	max($\mathtt{P}_1$, $\mathtt{P}_2$) (↓)	Better than ref. (↑)	Div. (↑)	Val. & Uniq. (↑)	QED (↑)	SA (↓)
TargetDiff	–	–	–	62.19 $\pm$ 27.08	−7.24 $\pm$ 2.35	0.32 $\pm$ 0.37	**0.89** $\pm$ 0.01	0.95 $\pm$ 0.07	0.57 $\pm$ 0.14	**0.59** $\pm$ 0.09
Dombi	1	0.3	✗	68.12 $\pm$ 27.38	−7.37 $\pm$ 2.51	0.26 $\pm$ 0.32	0.88 $\pm$ 0.02	**0.96** $\pm$ 0.10	0.58 $\pm$ 0.12	**0.59** $\pm$ 0.10
Dombi	1	1	✗	68.60 $\pm$ 28.09	−7.42 $\pm$ 2.57	0.28 $\pm$ 0.34	0.88 $\pm$ 0.02	**0.96** $\pm$ 0.09	0.58 $\pm$ 0.13	**0.59** $\pm$ 0.10
Dombi	1	3	✗	67.92 $\pm$ 28.17	−7.33 $\pm$ 2.61	0.28 $\pm$ 0.34	0.88 $\pm$ 0.01	**0.96** $\pm$ 0.09	0.57 $\pm$ 0.13	**0.59** $\pm$ 0.10
Dombi	1	0.3	✓	72.09 $\pm$ 31.16	−7.51 $\pm$ 2.64	0.31 $\pm$ 0.37	0.87 $\pm$ 0.02	0.95 $\pm$ 0.12	0.56 $\pm$ 0.13	**0.59** $\pm$ 0.11
Dombi	1	1	✓	72.83 $\pm$ 22.42	−7.71 $\pm$ 1.65	0.27 $\pm$ 0.35	0.86 $\pm$ 0.03	0.95 $\pm$ 0.08	0.57 $\pm$ 0.13	**0.59** $\pm$ 0.11
Dombi	1	3	✓	70.01 $\pm$ 27.94	−7.50 $\pm$ 2.50	0.28 $\pm$ 0.33	0.86 $\pm$ 0.02	**0.96** $\pm$ 0.10	0.58 $\pm$ 0.13	0.61 $\pm$ 0.09
Dombi	2	0.3	✗	72.54 $\pm$ 29.03	−7.67 $\pm$ 2.41	0.32 $\pm$ 0.35	0.88 $\pm$ 0.02	0.93 $\pm$ 0.16	0.59 $\pm$ 0.13	0.61 $\pm$ 0.10
Dombi	2	1	✗	71.36 $\pm$ 29.44	−7.59 $\pm$ 2.48	0.30 $\pm$ 0.34	0.88 $\pm$ 0.01	0.93 $\pm$ 0.16	0.59 $\pm$ 0.12	0.62 $\pm$ 0.09
Dombi	2	3	✗	72.92 $\pm$ 29.50	−7.74 $\pm$ 2.46	0.31 $\pm$ 0.36	0.88 $\pm$ 0.02	0.94 $\pm$ 0.16	**0.60** $\pm$ 0.12	0.62 $\pm$ 0.09
Dombi	2	0.3	✓	78.75 $\pm$ 33.36	−7.98 $\pm$ 2.51	0.37 $\pm$ 0.40	0.87 $\pm$ 0.03	0.94 $\pm$ 0.15	0.59 $\pm$ 0.12	0.61 $\pm$ 0.10
Dombi	2	1	✓	81.63 $\pm$ 25.91	−8.25 $\pm$ 1.56	0.38 $\pm$ 0.40	0.85 $\pm$ 0.11	0.93 $\pm$ 0.17	0.59 $\pm$ 0.12	0.62 $\pm$ 0.10
Dombi	2	3	✓	**83.06** $\pm$ 27.02	**−8.40** $\pm$ 1.61	**0.40** $\pm$ 0.41	0.85 $\pm$ 0.03	0.94 $\pm$ 0.12	0.57 $\pm$ 0.13	0.62 $\pm$ 0.09

D SAT-EXPERIMENT

D.1 SETUP

We illustrate the capability of Dombi compositions to adhere to combinatorial constraints by sampling uniformly from satisfying variable assignments of propositional formulas. For a formula with k propositional variables P_i, for $i \in [1, k]$, we set up our diffusion ensemble as follows: In $\mathbb{R}^k$, we place 2^k Guassian modes, one for each possible variable assignment. Then, in our ensemble, each of k score models simulates one propositional variable. For $i \in [1, k]$, we have access to s_i, which defines a denoising process to a uniform mixture of the 2^{k-1} Gaussian modes, where the P_i is true. Additionally, a reference model defines a denoising process uniformly to *all* 2^k Gaussian modes. For $k = 2$, this setup is visualized in Figure 8a.



(a) SAT experiment in $\mathbb{R}^2$.

		$k=2$			$k=5$			$k=10$		
Formula	Method	η_{corr}	η_{all}	PPL↑	η_{corr}	η_{all}	PPL↑	η_{corr}	η_{all}	PPL↑
Maj_k	Dombi	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**0.98**
	Prod	**1.00**	**1.00**	0.82	**1.00**	**1.00**	0.09	0.00	0.00	0.00
XOR_k	Dombi	0.97	**1.00**	**1.00**	**0.95**	**1.00**	**1.00**	**0.91**	**1.00**	**0.98**
	Prod	**1.00**	0.00	**1.00**	0.00	0.00	0.00	0.00	0.00	0.00
OneHot_k	Dombi	0.97	**1.00**	**1.00**	**0.59**	**1.00**	**1.00**	**0.09**	**1.00**	**0.99**
	Prod	**1.00**	0.00	**1.00**	0.00	0.00	0.00	0.00	0.00	0.00

(b) Overview of SAT experiment for three formulas.

Figure 8: Figure 8a shows the SAT experiment in $\mathbb{R}^2$, with squares corresponding to satisfying assignments. The corresponding numerical overview for $k \in \{2, 5, 10\}$ in Figure 8b. Best are bold.

Our objective is then to use score-composition to uniformly sample from all satisfying variable assignments. We repeat this setup for the Dombi operators, as well as PoE/MoE composition for three formulas for $k \in [1, 10]$, and report mode coverage, uniformity, and stability of the composition.

D.1.1 SAT FORMULAS

We use three different propositional formulas: majority, xor, and one-hot. The formulations of these formulas are designed to test different aspects of the score composition.

Majority We define the formula over k variables as

$$\text{Maj}_k(P_1, \ldots, P_k) \equiv \bigwedge_{\substack{S \subseteq \{P_1, \ldots, P_k\} \\ |S| = \lceil k/2 \rceil}} \bigvee_{P \in S}.$$

This formula is negation-free, but might lead to mode dropping for variable assignments with fewer positive variables.

One-Hot We define a formula where exactly one variable has to be true as

$$\text{OneHot}_k(P_1, \ldots, P_k) \equiv \Big(\bigvee_{i=1}^{k} P_i \Big) \wedge \Big(\bigwedge_{1 \le i < j \le k} (\neg P_i \vee \neg P_j) \Big).$$

It is only quadratic in the length of the variables, but it contains many clauses without positive literals, requiring precise handling of explicit negation.

Exclusive Or We define xor as a parity function over k variables as

$$\text{XOR}_k(P_1, \ldots, P_k) \equiv \bigwedge_{\substack{v \in \{0,1\}^k \\ \sum_i v_i \equiv 0 \pmod 2}} \bigvee_{i=1}^{k} \big(v_i ? \neg P_i : P_i\big).$$

This formula can only be expressed in exponential length with 2^{k-1} clauses, which explicitly exclude one assignment with even parity.

D.2 Score Model Setup

We translate each of the 2^k propositional variable assignments to a Gaussian mode in $\mathbb{R}^k$ as

$$p(\mathbf{x}) = \frac{1}{2^k} \sum_{v \in \{0,1\}^k} \mathcal{N}_k(\mathbf{x}|4v - 2, \sigma^2).$$

We then define "directional" diffusion models

$$\forall i \in [1, k] : p_i(\mathbf{x}) = \frac{1}{2^{k-1}} \sum_{\substack{v \in \{0,1\}^k \\ v_i = 1}} \mathcal{N}_k(\mathbf{x}|4v - 2, \sigma^2).$$

In this setup, each distribution plays the role of one propositional variable. The distributions p_i can then be composed to mirror a propositional formula, with the goal that particles converge only to modes that correspond to satisfying variable assignments. We use p as an additional stabilizing model to guide particles to any location that corresponds to an assignment.

As these models are mixtures of Gaussians, we derive optimal scores and energy functions from the standard Gaussian to our distributions in closed form.

We then model each type of formula for $k \in [1, 10]$ as direct composition and simulate 2^{14} particles over 100 denoising steps.

For each mode, when then check a L_∞ bounding box around its mean of sidelength 3σ and consider all particles within that radius to be valid assignments.

In Figure 8b we show the most important metrics: η_{corr}, the fraction of particles within bounding boxes of satisfying modes, η_{all}, the fraction of particles converging to any mode. Additionally, we measure the normalized perplexity in the particle distributions across as PPL. In this experiment, PPL measures mode uniformity, where a higher number indicates more uniform samples from satisfying modes of the formula. In a formula with K satisfying variable assignments, for a batch of n particles, with $n\eta_{\text{corr}}$ particles within satisfying modes, we denote the fraction of particles within the bounding box of the *assignment index* $i \in [1, K]$ as η_i with $\sum_i \eta_i = \eta_{\text{corr}}$. We then calculate PPL for mode confusion as

$$\text{PPL} = 2^{\left(-\sum_{i=1}^{K} \frac{\eta_i}{\eta_{\text{corr}}} \log_2 \frac{\eta_i}{\eta_{\text{corr}}}\right)} / K.$$

D.3 Results

Figure 8a shows samples of formulas in $\mathbb{R}^2$. An overview of the experimental results is provided in Figure 8b. We can see multiple shortcomings of products in our experimental results. On the negation-free Maj_k, PoE drastically reduces the per-mode variance, as seen in Figure 8a, drops most of the modes for $k = 5$, and completely breaks down for $k = 10$. In contrast to this, the dombi Operators do not drop modes and maintain a close-to-uniform distribution over modes in high dimensions. For XOR_k and OneHot_k PoE breaks down for $k = 2$ already, due to the negated literals. In Figure 8a, the modes of the PoE sample appear drastically biased by the negated clause. Somewhat surprisingly, the Dombi composition can sample comparatively well from the exponentially sized XOR_{10}, and struggles much more for OneHot, which is comprised of many purely negative clauses.