Diffusion Forcing
下一 token 预测与全序列扩散的统一

Diffusion Forcing: Next-token Prediction Meets Full-Sequence Diffusion

Boyuan Chen · Diego Marti Monso · Yilun Du
Max Simchowitz · Russ Tedrake · Vincent Sitzmann

NeurIPS 2024 · arXiv:2407.01392v4 · 2024-12-10
原版35页 · 中文译编 2026-11-25

许可：CC BY 4.0（creativecommons.org/licenses/by/4.0/）。原作署名如上。改动为中文翻译、重新排版和两张 arXiv HTML 失效表格重建。全部正文、附录A–F、原图、公式与英文书目保留；原作者未参与或认可译文。源PDF与逐段中英对照、源映射一并提供。

阅读路线

先读第3节的双轴噪声与算法，再读第4节各任务的实际证据；A给出ELBO及表达力前提，B解释噪声日程，D列出可复现实现，E–F保留完整结果和数据口径。

原文：arxiv.org/abs/2407.01392v4

本文提出 Diffusion Forcing：一种训练扩散模型的新范式，令每个 token 具有相互独立的噪声等级，并学习对这一组 token 去噪。我们把它用于序列生成：训练因果下一 token 预测模型，在不必将过去 token 完全扩散的情况下生成一个或多个未来 token。该方法兼有下一 token 模型的可变长度生成能力，以及全序列扩散通过引导获得理想轨迹的能力。它还支持若干新功能：（1）把视频等连续 token 序列滚动生成到超过训练时域的长度，而基线会在此处发散；（2）利用可变时域和因果架构设计新的采样、引导方案，从而显著改善决策和规划。除经验结果外，我们证明该方法优化了真实联合分布中所有 token 子序列似然的一个变分下界。项目网站：https://boyuan.space/diffusion-forcing。

1 引言

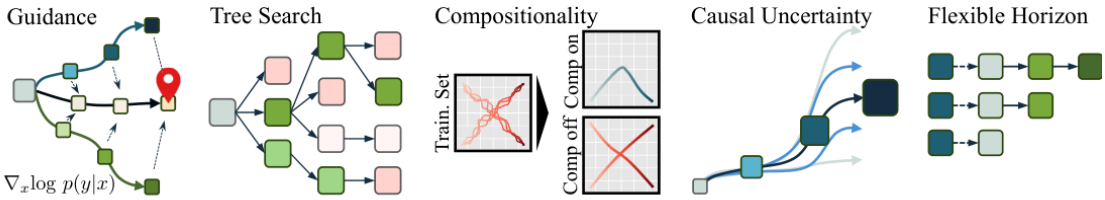
概率序列建模在自然语言处理 [6,48]、视频预测 [32,70,26] 和决策 [3,22] 等机器学习应用中至关重要。尤其是下一 token 预测模型具有多项优点：能够生成可变长度序列 [33,21,38]，既能只生成一个 token，也能经自回归采样生成“无限”多个 token；能以不同长度的历史为条件 [21,38]；支持高效树搜索 [71,23,25]；还可用于在线反馈控制 [22,3]。

现有下一 token 模型通过教师强制训练 [65]，即根据真实的过去 token 历史预测紧接着的 token。这带来两项限制：（1）没有机制引导整条序列的采样去最小化某个目标；（2）在连续数据上容易失稳。例如，自回归生成视频（不同于文本 [6] 或向量量化潜变量 [34]）并超过训练时域时，微小的帧间误差会累积，使模型发散。

全序列扩散似乎提供了解法。视频生成和长时域规划常直接对固定数量 token 的拼接向量做扩散，以建模其联合分布 [32,1]，所有 token 具有相同噪声等级。扩散引导 [31,16] 可将采样推向理想序列，对规划至关重要 [37,35]；此类模型也擅长生成视频等连续信号 [32]。然而，全序列扩散普遍采用非因果、无掩码架构。它不但把采样限制为完整序列而非可变长度生成，而且如图1所示，限制了引导和子序列生成。我们还证明，简单地训练一个下一 token 模型来做全序列扩散，并不能兼得二者优点，生成质量反而较差；直觉原因是它未体现：早期 token 的微小不确定性，必然使后续 token 具有更高不确定性。

我们提出 Diffusion Forcing（DF）训练与采样范式：每个 token 关联随机且独立的噪声等级，共享的下一或下几个 token 预测模型可按任意、逐 token 独立的日程去噪。出发点是：加噪是一种部分掩码；零噪声表示未遮蔽，完全噪声表示完全遮蔽。DF 因而迫使模型学会对任意一组具有不同噪声的 token “解除掩码”（图2）。同时，把预测参数化为下一 token 模型的组合，使系统能灵活生成不同长度序列，并组合泛化到新轨迹（图1）。

我们用因果架构实现序列生成中的 DF，称为因果 Diffusion Forcing（CDF），使未来 token 依赖过去。训练时同时对整个序列去噪，但每个 token 的噪声等级独立。采样时，从高斯噪声帧逐渐得到干净序列；在每一步，不同帧可以处于不同噪声等级。CDF 像下一 token 模型一样支持可变长度，且从紧接着的 token 到未来数千个 token 均能稳定生成，即使 token 是连续值；又像全序列扩散一样支持高奖励引导。因果性、灵活时域和可变噪声日程协同产生了蒙特卡洛引导（MCG）：相对非因果全序列扩散，显著改善高奖励样本的采样。图1概括这些能力。



Teacher Forcing	✗	✓	✓	✓	✓
Full-Seq. Diffusion	✓	✗	✗	✗	✗
Diffusion Forcing	✓	✓	✓	✓	✓

图1：Diffusion Forcing 的能力。语言建模 [6]、规划 [37] 和视频生成 [32,70] 等应用，现今根据不同能力需求选用自回归下一 token 预测或全序列扩散。本文的新型序列生成模型 Diffusion Forcing 同时具有两类模型的关键优点。

贡献概括如下：（1）提出新的概率序列模型 Diffusion Forcing，兼具下一 token 模型的灵活性和全序列扩散的长时域引导能力。（2）利用其独有能力，提出同时把 DF 作为策略 [10] 和规划器 [37] 的决策框架。（3）形式化证明：在适当条件下，优化所提训练目标可最大化训练时所见全部子序列联合分布似然的一个下界。（4）在视频生成、基于模型的规划、视觉模仿学习和时间序列预测中进行实验，展示长滚动自回归视频稳定化、按用户设定记忆时域组合训练子序列、蒙特卡洛引导等能力。

2 相关工作与预备知识

本节讨论核心应用——序列生成建模——的相关工作与预备知识；更多文献见附录C。

本方法统一了序列建模的两种视角：沿时间轴的贝叶斯滤波，以以下标 t 表示；沿“不确定性”（噪声等级）轴的扩散，以上标 k 表示。下文将观测记为 $x \in \mathcal{X}$，潜状态记为 $z \in \mathcal{Z}$。

> t　k　$\mathbf{x} \in \mathcal{X}$　$\mathbf{z} \in \mathcal{Z}$

贝叶斯滤波。

给定由潜状态 z_t 和观测 x_t 定义的隐马尔可夫模型，贝叶斯滤波利用新到来的观测递归估计潜状态。先验模型 $p(z_{t+1}|z_t)$ 仅根据当前状态推断下一状态的信念；观测模型 $p(x_t|z_t)$ 根据当前潜状态推断下一观测的信念。当新观测到来，后验模型 $p(z_{t+1}|z_t,x_{t+1})$ 更新对下一潜状态的估计。用神经网络端到端训练时 [22,23]，潜状态未必估计某个物理量，而是以足够有表达力的潜表示概括过去，供预测未来观测使用。

> $\mathbf{z}_t$　$\mathbf{x}_t$　$p(\mathbf{z}_{t+1}|\mathbf{z}_t)$　$p(\mathbf{x}_t|\mathbf{z}_t)$　$p(\mathbf{z}_{t+1}|\mathbf{z}_t,\mathbf{x}_{t+1})$　$\mathbf{z}_{t+1}$　$(\mathbf{x}_{t'})_{t'>t}$

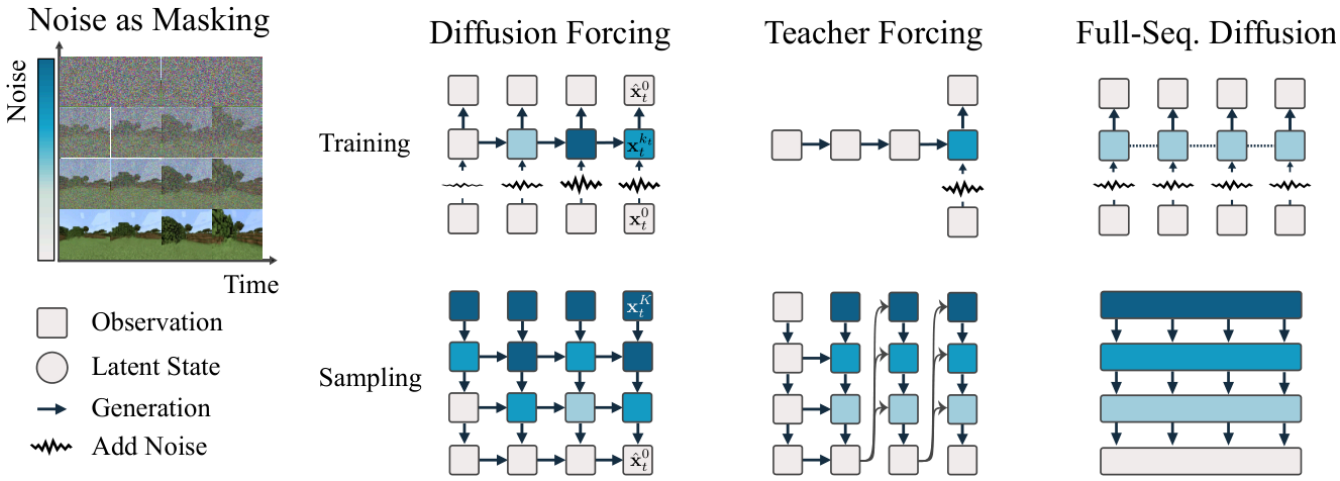


图2：方法概览。Diffusion Forcing 训练 RNN 或带掩码 Transformer 等因果序列神经网络，对可变长度序列去噪，其中每帧可具有不同噪声等级。相反，语言建模中常见的下一 token 模型通过教师强制 [65]，以真实历史预测单个后继 token；视频生成常用的全序列扩散则训练非因果架构，以同一噪声等级同时对所有帧去噪。因此，DF 交织了序列时间轴和扩散噪声轴，统一二者优点，并产生全新能力（3.2、3.4节）。

扩散模型。

扩散模型 [57,29] 已显示出很强的表达力和可靠性。设 q(x) 为目标数据分布，$x^0 \equiv x \sim q$。正向扩散在一系列步骤中逐渐向数据添加高斯噪声；它是马尔可夫链，每一步均增量加噪，定义如下。

> $q(\mathbf{x})$ $\mathbf{x}^0 \equiv \mathbf{x} \sim q$ k

$$q(\mathbf{x}^k|\mathbf{x}^{k-1}) = \mathcal{N}(\mathbf{x}^k, \sqrt{1-\beta_k}\mathbf{x}^{k-1}, \beta_k\mathbf{I}) \quad (2.1)$$

其中，$\mathcal{N}$ 表示正态分布；β_k 是每步所加噪声的方差，由日程 $\{\beta_k \in (0,1)\}$ 控制。持续加噪直至 x^K 近似为纯噪声。逆过程同样是马尔可夫链，以参数模型 pθ 尝试从噪声重建原始数据。

> $\mathcal{N}$ β_k $\{\beta_k \in (0,1)\}_{k=1}^K$ $\mathbf{x}^K$ p_θ

$$p_\theta(\mathbf{x}^{k-1}|\mathbf{x}^k) = \mathcal{N}(\mathbf{x}^{k-1}, \boldsymbol{\mu}(\mathbf{x}^k, k), \gamma_k\mathbf{I}), \quad (2.2)$$

其中均值 μ 由神经网络建模；已有工作 [30] 表明，协方差可设为单位矩阵乘以只依赖 k 的固定常数 γ_k。按标准推导，用噪声预测 ε 对均值重新参数化，可得到以下最小二乘目标 [29]。

> $\boldsymbol{\mu}$ γ_k k $\boldsymbol{\mu}$ $\epsilon = (\sqrt{1-\bar{\alpha}_t})^{-1}\mathbf{x}_t^{k_t} - \sqrt{\bar{\alpha}_t}\boldsymbol{\mu}$

$$\mathcal{L}(\theta) = \mathbb{E}_{k,\mathbf{x}^0,\epsilon}\left[\|\epsilon^k - \epsilon_\theta(\mathbf{x}^k, k)\|^2\right], \quad (2.3)$$

其中，加噪样本由干净数据与标准高斯噪声按 $\sqrt{\bar{\alpha}}$ 和 $\sqrt{(1-\bar{\alpha})}$ 线性组合。随后可通过所列朗之万动态更新，从模型采样 [29]。原文该处时间/噪声下标沿用底本，具体正反向关系见保留的公式。

> $\mathbf{x}^k = \sqrt{\bar{\alpha}_t}\mathbf{x}^0 + \sqrt{1-\bar{\alpha}_t}\epsilon^k$ $\epsilon^k \sim \mathcal{N}(0,\mathbf{I})$ $\mathbf{x}^{k-1} \leftarrow \frac{1}{\sqrt{\alpha_k}}(\mathbf{x}_t^k - \frac{1-\alpha_k}{\sqrt{1-\bar{\alpha}_k}}\epsilon_\theta(\mathbf{x}_t^k, k) + \sigma_k\mathbf{w})$

扩散模型的引导。

引导 [31,16] 在采样时使扩散生成偏向理想预测。我们关注分类器引导 [16]：给定期望标签 y（如类别或成功指示）的分类器 c(y|xᵏ)，将朗之万采样 [30] 的噪声预测减去 √(1−ᾱₖ)∇log c。这样无需训练条件模型，就能从 x 与 y 的联合分布采样。决策等应用还研究了其他能量，例如比较模型输出和期望真值的最小二乘目标 [16,37]。

> $c(y|\mathbf{x}^k)$ y $\epsilon_\theta(\mathbf{x}^k, k)$ $\epsilon_\theta(\mathbf{x}^k, k) - \sqrt{1-\bar{\alpha}_k}\,\nabla_{x^k} \log c(y|\mathbf{x}^k)$ $\mathbf{x}$ y

下一 token 预测模型。

下一 token 模型根据历史帧 x₁:ₜ 预测 xₜ₊₁。训练时输入历史，连续数据最小化平方误差，离散数据最小化交叉熵 [65]。采样时从 p(xₜ₊₁|x₁:ₜ) 生成下一帧，再将预测当作真实下一帧，递归预测其后各帧，直至生成完整序列。不同于全序列扩散，下一 token 模型不支持多步引导，因为采样未来之前，过去帧必须完全确定。

> $\mathbf{x}_{t+1}$ $\mathbf{x}_{1:t}$ $\mathbf{x}_{1:t}$ $\|\hat{\mathbf{x}} - \mathbf{x}\|^2$ $\hat{\mathbf{x}}_{t+1}$ $p(\mathbf{x}_{t+1} \mid \mathbf{x}_{1:t})$ $\hat{\mathbf{x}}_{t+1}$ $\mathbf{x}_{t+1}$ $\mathbf{x}_{t+2}$

扩散序列模型。

扩散已广泛用于序列建模。[44] 用全序列扩散和引导实现按指定词性等条件的文本生成；[32] 用全序列扩散生成短视频，再通过以先前帧为条件的滑动窗口生成长视频；[37] 用环境交互轨迹训练全序列扩散，并在采样时经分类器引导得到通向目标的高奖励轨迹，作为离线强化学习规划器。[50] 让自回归模型以过去 token 为条件，对下一个 token 去噪，以教师强制训练并自回归生成时间序列。最接近我们的是 AR-Diffusion [66]：以因果架构训练全序列文本扩散，噪声等级沿时间轴线性相关。详细比较见附录C。

3 方法

3.1 把加噪视为部分掩码

掩码指遮蔽数据的一部分，例如图像块 [27] 或序列时间步 [15,49]，再训练模型恢复未遮蔽内容。不失一般性，可把任意 token 集合（无论是否成序列）视为按 t 索引的有序集合。教师强制的下一 token 训练就相当于在时刻 t 遮蔽 xₜ，再从过去 x₁:ₜ₋₁ 预测。对序列而言，这些操作称为沿时间轴掩码。全序列正向扩散——逐步给干净序列加噪——同样可视为部分掩码，称为沿噪声轴掩码。因为 K 步加噪后，序列近似为不含原始数据信息的纯白噪声。

> t $\mathbf{x}_t$ t $\mathbf{x}_{1:t-1}$ $\mathbf{x}^0{}_{1:T} \equiv \mathbf{x}_{1:T}$ K $\mathbf{x}^K{}_{1:T}$

我们统一这两个掩码轴（图2）。x₁:ₜ 的下标表示时间轴；xₜᵏᵗ 表示通过正向扩散式(2.1)获得的第 t 个 token，噪声等级为 kₜ。xₜ⁰ 是无噪声 token，xₜᴷ 是标准白噪声。因此，序列 (xₜᵏᵗ) 中各 token 的不同 kₜ，可看作由加噪施加的部分掩码程度。

> $\mathbf{x}_{1:T}$ $\mathbf{x}_t^{k_t}$ $\mathbf{x}_t$ k_t $\mathbf{x}_t^0 = \mathbf{x}$ $\mathbf{x}_t^K$ $\mathcal{N}(0, \mathbf{I})$ $(\mathbf{x}_t^{k_t})_{1 \le t \le T}$ k_t

3.2 Diffusion Forcing：不同 token 使用不同噪声等级

DF 是训练并采样任意长度噪声 token 序列的框架；关键在于每个 token 的 k_t 可随时间步独立变化。本文关注时间序列，因而用因果架构实例化：当前噪声 token 只依赖过去噪声 token，称为 CDF。为简洁起见，采用普通 RNN [11] 的最小实现；也可用 Transformer，见附录B.1。

> $(\mathbf{x}_t^{k_t})_{1 \le t \le T}$ k_t $\mathbf{x}_t^{k_t}$

算法1：Diffusion Forcing 训练

循环
  采样观测轨迹 $(\mathbf{x}_1, ..., \mathbf{x}_T)$.
  对 $t = 1, ..., T$ 执行
    采样独立噪声等级 $k_t \in \{0, 1, ..., K\}$
    $\mathbf{x}_t^{k_t} =$ 正向扩散 $(\mathbf{x}_t, k_t)$
    定义 $\epsilon_t = \frac{\mathbf{x}_t^{k_t} - \sqrt{\alpha_{k_t}}\mathbf{x}_t}{\sqrt{1-\alpha_{k_t}}}$
    更新 $\mathbf{z}_t \sim p_\theta(\mathbf{z}_t | \mathbf{z}_{t-1}, \mathbf{x}_t^{k_t}, k_t)$.
    设置 $\hat{\epsilon}_t = \epsilon_\theta(\mathbf{z}_{t-1}, \mathbf{x}_t^{k_t}, k_t)$
  结束 对
  $L =$ 均方误差 $([\hat{\epsilon}_1, ..., \hat{\epsilon}_n], [\epsilon_1, ..., \epsilon_n])$
  以此损失反向传播 L 并更新 θ
结束 循环

算法2：带引导的 DF 采样

输入: Model θ, 日程矩阵 $\mathcal{K}$, 初始潜变量 $\mathbf{z}_0$, 引导代价 $c(\cdot)$.
Initialize $\mathbf{x}_1, ..., \mathbf{x}_T \sim \mathcal{N}(0, \sigma_K^2 I)$.
对 row $m = M-1, ..., 0$ 执行
  对 $t = 1, ..., T$ 执行
    $\mathbf{z}_t^{\text{new}} \sim p_\theta(\mathbf{z}_t | \mathbf{z}_{t-1}, \mathbf{x}_t, \mathcal{K}_{m+1,t})$.
    $k \leftarrow \mathcal{K}_{m,t}$, $\mathbf{w} \sim \mathcal{N}(0, \mathbf{I})$.
    $\mathbf{x}_t^{\text{new}} \leftarrow \frac{1}{\sqrt{\alpha_k}}(\mathbf{x}_t - \frac{1-\alpha_k}{\sqrt{1-\alpha_k}}\epsilon_\theta(\mathbf{z}_t^{\text{new}}, \mathbf{x}_t, k)) + \sigma_k \mathbf{w}$
    更新 $\mathbf{z}_t \leftarrow \mathbf{z}_t^{\text{new}}$.
  结束 对
  $\mathbf{x}_{1:H} \leftarrow$ AddGuidance $(\mathbf{x}_{1:H}^{\text{new}}, \nabla_{\mathbf{x}} \log c(\mathbf{x}_{1:H}^{\text{new}}))$
结束 对
返回 $\mathbf{x}_{1:T}$.

权重为 θ 的 RNN 以潜变量 z_t 概括过去 token 的影响，并由递归层按 $p\theta(z_t|z_{t-1},x_t^{kt},k_t)$ 演化。新噪声观测到来时，隐藏状态以马尔可夫方式更新。脚注1：实现中该映射是确定性的；z_t 表示信念分布，而非从该分布抽取的样本，因而可以通过式(3.1)中的潜动态反向传播。$k_t=0$ 时，它对应贝叶斯滤波的后验更新；$k_t=K$ 时输入是无信息纯噪声，相当于先验 $p\theta(z_t|z_{t-1})$。给定潜状态，观测模型 $p\theta(x_t^o|z_t)$ 预测干净 x_t。

> θ $\mathbf{z}_t$ $\mathbf{z}_t \sim p_\theta(\mathbf{z}_t|\mathbf{z}_{t-1}, \mathbf{x}_t^{k_t}, k_t)$ $\mathbf{x}_t^{k_t}$ $\mathbf{z}_t \sim p_\theta(\mathbf{z}_t|\mathbf{z}_{t-1}, \mathbf{x}_t^{k_t}, k_t)$ $\mathbf{z}_t = p_\theta(\mathbf{z}_t|\mathbf{z}_{t-1}, \mathbf{x}_t^{k_t}, k_t)$ $\mathbf{z}_t$ $k_t = 0$ $k_t = K$ $\mathbf{x}_t^K$ $p_\theta(\mathbf{z}_t|\mathbf{z}_{t-1})$ $\mathbf{z}_t$ $p_\theta(\mathbf{x}_t^0|\mathbf{z}_t)$ $\mathbf{x}_t$

训练。

动态模型和观测模型共同构成一个 RNN 单元。其输入输出与标准条件扩散模型一致：以 z_{t-1} 和噪声 token 为输入，预测干净 token，并经仿射重参数化间接预测噪声 [30]。因此可直接用常规扩散目标训练（因果）DF。我们将该单元参数化为噪声预测 $\varepsilon\theta(z_{t-1},x_t^{kt},k_t)$，并最小化下列损失求得 θ。

> $p_\theta(\mathbf{z}_t|\mathbf{z}_{t-1}, \mathbf{x}_t^{k_t}, k_t)$ $p_\theta(\mathbf{x}_t^0|\mathbf{z}_t)$ $\mathbf{z}_{t-1}$ $\mathbf{x}_t^{k_t}$ $\mathbf{x}_t = \mathbf{x}_t^0$ ϵ^{k_t} $\epsilon_\theta(\mathbf{z}_{t-1}, \mathbf{x}_t^{k_t}, k_t)$ θ

$$\mathop{\mathbb{E}}_{\substack{k_t, \mathbf{x}_t, \epsilon_t \\ \mathbf{z}_t \sim p_\theta(\mathbf{z}_t|\mathbf{z}_{t-1}, \mathbf{x}_t^{k_t}, k_t)}} \sum_{t=1}^{T} \left[\|\epsilon_t - \epsilon_\theta(\mathbf{z}_{t-1}, \mathbf{x}_t^{k_t}, k_t)\|^2 \right], \tag{3.1}$$

其中 $k_{1:T}$ 从 $[K]^T$ 均匀采样；干净序列来自训练集；噪声按正向过程的方差采样（伪代码见算法1）。重要的是，式(3.1)结合了贝叶斯滤波和条件扩散的核心。附录D.1把常见扩散训练技巧重新推导到 DF，在视频预测中极有用；附录B.2讨论均匀采样噪声序列的必要性。附录A证明以下非正式定理3.1。

> $k_{1:T}$ $[K]^T$ $\mathbf{x}_{1:T}$ $\epsilon_t \sim \mathcal{N}(0, \sigma_{k_t}^2 I)$ $k_{1:T}$

定理 3.1 (Informal).

定理3.1：DF 训练过程（算法1）优化了部分加噪序列期望对数似然的证据下界（ELBO）的一个重新加权版本。期望对均匀采样的噪声等级序列取平均，各噪声 token 按正向过程产生。此外，在适当条件下，优化式(3.1)可同时最大化所有噪声等级序列似然的一个下界。

> $\ln p_{\boldsymbol{\theta}}((\mathbf{x}_t^{k_t})_{1 \le t \le T})$ $k_{1:T} \sim [K]^T$ $\mathbf{x}_t^{k_t}$

“所有噪声等级序列”的特殊情形是：每个 k_t 取 0 或 K。因此，可掩去任意过去 token，DF 仍会学习正确条件分布，建模训练集所有可能子序列的分布。

> $k_t = 0$ $k_t = K$

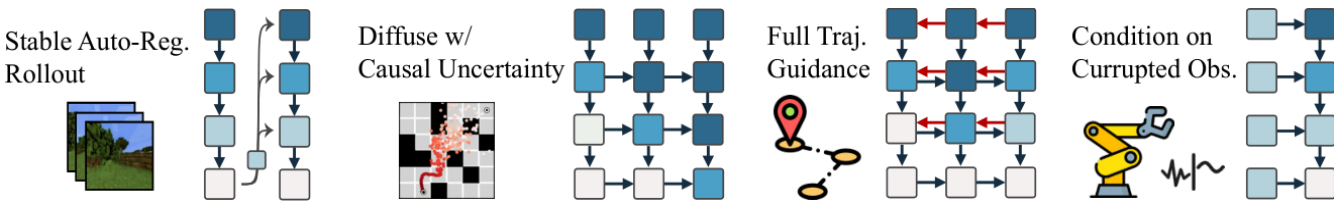
采样。

算法2通过二维 M×T 日程网格 $\mathcal{K}$ 定义采样：列为时间 t，行 m 规定噪声等级，$\mathcal{K}_{m,t}$ 是该行第 t 个 token 的目标噪声等级。生成长度 T 的序列时，先把所有 token 初始化为 k=K 的白噪声；逐行向下迭代，每行从左到右去噪至网格指定等级。最后 m=0 行所有等级均为0，得到干净 token。边界情形见D.5，α、$\bar{\alpha}$、σ 使用标准设定 [30]。$\mathcal{K}$ 决定每个 token 在每个扩散步骤的去噪速度。模型训练覆盖所有噪声等级序列，因此只需设计不同 $\mathcal{K}$，即可灵活改变行为，而不必重新训练。

> $M \times T$ $\mathcal{K} \in [K]^{M \times T}$ t m $\mathcal{K}_{m,t}$ t m T $\mathbf{x}_{1:T}$ $k = K$ $\mathcal{K}$ $m = 0$ $\mathcal{K}_{0,t} \equiv 0$ $(\alpha_k, \bar{\alpha}_k, \sigma_k)$ $\mathcal{K}$ $\mathcal{K}$

3.3 序列生成的新能力

下面解释这一灵活采样范式产生的新能力。



稳定自回归生成。

视频等高维连续序列的自回归模型容易发散，尤其当采样长度超过训练时域。DF 则把先前 token 按很小的正噪声等级 0<k≪K 处理，并使用与此略带噪声 token 相应的先前潜状态更新后续潜状态，从而稳定生成超出训练长度的长序列。4.1节展示长时域生成的显著改善；B.4给出进一步直觉。

> $0 < k \ll K$

让未来保持不确定。

从三个白噪声 token 出发，可先将第一个完全去噪、第二个去噪到 K/2、第三个保持 K；再让前两个完全去噪、第三个保持 K/2；最后全变为零噪声。将噪声等级理解为不确定性，这种“之字形”日程使近未来比远未来更确定。3.4节说明它如何改善序列引导。

> $[\mathbf{x}_1^K, \mathbf{x}_2^K, \mathbf{x}_3^K]^\top \quad [\mathbf{x}_1^0, \mathbf{x}_2^{K/2}, \mathbf{x}_3^K]^\top \quad [\mathbf{x}_1^0, \mathbf{x}_2^0, \mathbf{x}_3^{K/2}]^\top \quad [\mathbf{x}_1^0, \mathbf{x}_2^0, \mathbf{x}_3^0]^\top$

长时域引导。

可在算法2第10行按第2节，对部分扩散的轨迹加入引导。未来依赖过去，所以未来 token 的引导梯度能够沿时间反向传播。DF 的独特优势是：无需把过去完全扩散就能扩散未来，因此梯度可以引导过去 token 的采样，在尊重因果关系的同时实现长时域引导。实现细节见B.3；4.2节表明这种规划显著优于带引导的全序列扩散。

> $\mathbf{x}_{1:T}$

3.4 用于灵活序贯决策的 Diffusion Forcing

DF 的能力促成了一种面向机器人和自主智能体的序贯决策框架。考虑由动态 $p(s_{t+1}|s_t,a_t)$、观测 $p(o_t|s_t)$ 和奖励 $p(r_t|s_t,a_t)$ 定义的马尔可夫决策过程。目标是训练 $\pi(a_t|o_{1:t})$，最大化轨迹的期望累积奖励。令 token $x_t=[a_t,r_t,o_{t+1}]$；轨迹是可能变长的 token 序列，按算法1训练。执行到 t 时，无噪声过去由 z_{t-1} 概括；以其为条件，通过算法2采样计划 $\hat{x}_{t:t+h}$，包含动作、奖励及观测预测。H 类似模型预测控制的前瞻窗 [20]。执行计划动作后，环境返回实际奖励和新观测，组成真实新 token，再用零噪声后验更新潜状态。这使框架可同时充当策略与规划器。

> $p(\mathbf{s}_{t+1}|\mathbf{s}_t,\mathbf{a}_t) \quad p(\mathbf{o}_t|\mathbf{s}_t) \quad p(\mathbf{r}_t|\mathbf{s}_t,\mathbf{a}_t) \quad \pi(\mathbf{a}_t \mid \mathbf{o}_{1:t}) \quad \mathbb{E}[\sum_{t=1}^{T} \mathbf{r}_t] \quad \mathbf{x}_t=[\mathbf{a}_t,\mathbf{r}_t,\mathbf{o}_{t+1}] \quad \mathbf{x}_{1:T} \quad t \quad \mathbf{x}_{1:t-1} \quad \mathbf{z}_{t-1}$
> $\hat{\mathbf{x}}_{t:t+H} \quad \hat{\mathbf{x}}_t=[\hat{\mathbf{a}}_t,\hat{\mathbf{r}}_t,\hat{\mathbf{o}}_{t+1}]^\top \quad H \quad \hat{\mathbf{a}}_t \quad \mathbf{r}_t \quad \mathbf{o}_{t+1} \quad \mathbf{x}_t=[\hat{\mathbf{a}}_t,\mathbf{r}_t,\mathbf{o}_{t+1}]^\top \quad p_\theta(\mathbf{z}_t|\mathbf{z}_{t-1},\mathbf{x}_t,0)$

灵活规划时域。

DF（a）逐步选择动作，因此能用于可变总时域任务；（b）可缩短前瞻 H 降低延迟，作为策略使用，也可增大 H 配合引导做长时域规划，不必重训或改架构。Diffuser [37] 等全轨迹生成模型无法实现（a），而扩散策略 [10] 通常需要固定的小前瞻窗，无法实现（b）。

> H

灵活奖励引导。

如B.3所述，可用任意定义在未来步骤上的奖励代替 log c：包括整条轨迹的稠密逐步奖励、未来前瞻窗内的稠密奖励，以及表示目标完成的稀疏奖励 $-||o_t-g||^2$。逐时刻策略无法利用后一类长时域引导。

> $\log c \quad \sum_{t=1}^{T} \mathbf{r}_t \quad \sum_{t'=t}^{t+H} \mathbf{r}_t \quad -\|\mathbf{o}_T-\mathbf{g}\|^2$

蒙特卡洛引导（MCG）与未来不确定性。

CDF 可根据整个未来分布的引导来影响当前 token x_t^k。不只抽取一条未来轨迹算梯度，而是多次采样未来，再平均引导梯度，这就是 MCG。与 MPPI [64] 等打靶方法的思想一致，当前 token 被所有未来结果分布的期望奖励引导，而非某个特定结果。当日程在对近期 token 去噪时令远期 token 维持高噪声（如3.3节之字形日程），MCG 效果更强，因为它考虑了远未来更大的不确定性。B.5进一步解释其意义及 DF 如何独特地利用它。

> $\mathbf{x}_t^k \quad \mathbf{x}_{t+1:T} \quad \mathbf{x}_t^k$

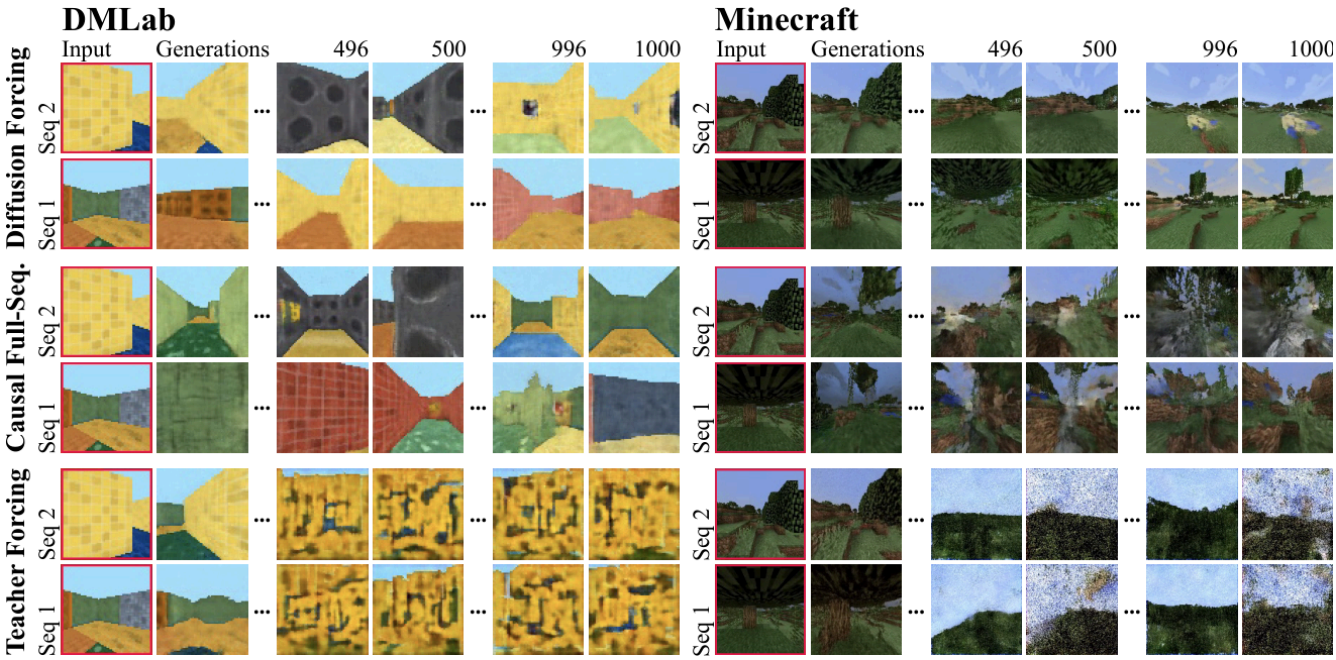


图3：视频生成。在测试方法中，只有 DF 的生成既保持时间一致，又在远超训练时域的滚动生成中不发散。动态视频结果见项目网站。

4 实验

我们在视频与时间序列预测、规划和模仿学习等多类应用中广泛评估 DF 作为生成序列模型的优点。数据集与复现细节见附录，视频见项目网站。

4.1 视频预测：一致、稳定的序列生成与无限滚动

我们在 Minecraft 游戏视频和 DMLab 导航视频 [69] 上训练卷积 RNN 版 CDF。采样时用3.3节的稳定化方法自回归滚动。两个基线采用完全相同 RNN 架构：教师强制训练的下一帧扩散 [65]，以及因果全序列扩散。图3展示从未见过的起始帧生成的结果。DF 即使远超训练时域（如1000帧）仍稳定，而教师强制和全序列扩散很快发散。在训练时域以内，全序列扩散也有明显帧间跳变；DF 则表现为在一致三维环境中的自运动。这突出了其抑制高维序列误差累积的能力。

> 1000

表1：DF 用于规划。（上）采样时每个时间步可按不同噪声日程去噪，从而在引导规划中考虑因果不确定性；DF 让远未来比近未来更不确定，而 Diffuser [37] 对各时刻使用相同噪声。（下）DF 的跨运行平均奖励最高。直接执行实际生成动作时，Diffuser 会严重失败；它需要忽略生成动作并使用手工 PD 控制器（星号所示）。

4.2 扩散规划：MCG、因果不确定性与灵活时域控制

决策尤其能受益于 DF。我们在标准离线强化学习基准 D4RL [18] 的稀疏奖励二维迷宫中评估：智能体从随机起点到达指定目标。环境细节见附录F.2。基准给出迷宫随机行走数据，因此具有随机性；每个迷宫单独训练一个模型。

我们将3.4节决策框架与领先离线强化学习方法及新近的扩散规划器 Diffuser [37] 比较。定性和定量结果见表1：DF 在全部6个环境上均超过 Diffuser 与各基线。

> 6

蒙特卡洛引导的收益。

强化学习通常要找最大化未来期望奖励的动作，MCG 实现了这一点。B.5形式化说明，Diffuser 等全序列扩散不支持按这种方式最大化期望奖励。为检验 MCG 的重要性，我们做去除消融；性能降低，但 DF 仍有竞争力。

因果建模的收益。

不同于纯生成建模，序贯决策要执行动作并接收反馈。因为不确定性累积，近期动作比远期更重要。尽管 Diffuser 等训练生成动作—奖励—状态三元组，直接执行所生成动作却会使真实轨迹明显偏离所生成状态，即二者不满足因果一致性。Diffuser 的实现因此忽略生成动作，用手工 PD 控制器由生成状态推断动作。表1显示，直接执行其动作会使性能剧降。相反，DF 原生生成的动作自洽，甚至优于 Diffuser 状态预测加手工 PD 控制器。

> $[\mathbf{a}_t, \mathbf{r}_t, \mathbf{o}_t]$

灵活时域的收益。

许多强化学习任务具有固定总时长，智能体推进任务时，剩余规划时域应缩短。DF 天然支持这一点；Diffuser 等全序列模型即使作调整仍表现较差，原因见B.6。

4.3 可控序贯组合生成

只修改采样方案，就能灵活组合训练中所见序列的子序列。数据集由二维正方形平面上从一角到对角的轨迹构成，整体呈叉形。不需要组合时，DF 保留完整记忆，重现叉形分布；需要组合时，让模型以无记忆 MPC 生成较短计划，就能拼接叉形的子轨迹，形成 V 形轨迹。篇幅所限，结果见E.2。

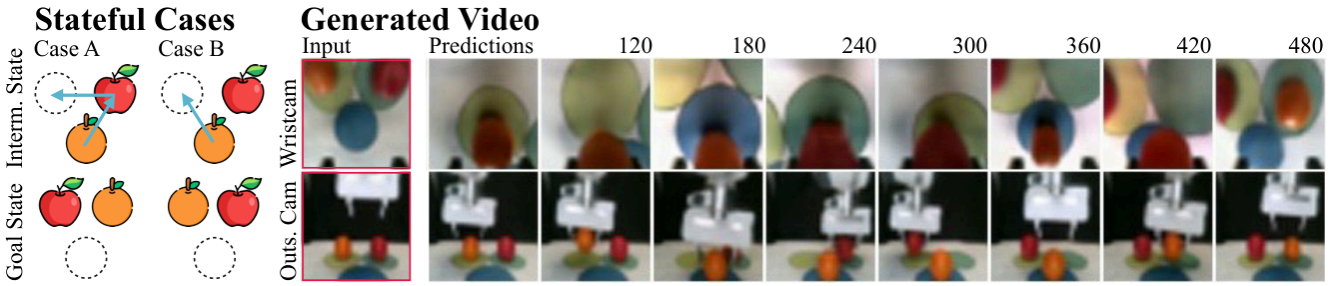


图4：真实机器人任务要求机械臂借助第三个槽位交换两个水果。初始槽位随机，所以如果不知道最初布局，仅凭单帧无法确定下一动作。如(a)(b)所示，上方观测相同，下方期望结果却可不同，必须记忆初始配置。此外(c)表明，生成动作的同一模型还能仅凭一帧合成真实感视频。

4.4 机器人：长时域模仿学习与鲁棒视觉运动控制

最后，我们说明 DF 为真实机器人视觉运动控制带来的新机会。模仿学习 [10] 从专家示范学习观测到动作的映射，广泛用于操作，但缺乏记忆常阻碍长时域任务。DF 不仅缓解这一缺点，还使模仿学习更鲁棒。

带记忆的模仿学习。

我们遥操作 Franka 机器人收集视频和动作，让它借第三槽交换苹果和橙子（图4）。随机初始布局产生两种可能目标。当一个水果位于第三槽时，仅当前观测无法判断目标，策略必须记住初始配置来决定移动哪个水果。不同于普通行为克隆，DF 的潜状态自然包含记忆。DF 成功率为80%，而没有记忆的领先模仿学习算法扩散策略 [10] 失败。

> 80%

对缺失或噪声观测的鲁棒性。

DF 结合贝叶斯滤波原理，可以在模仿学习中抵抗观测噪声与缺失。我们在执行中加入视觉干扰，甚至完全遮挡相机。只需用 k>0 标明观测“有噪声”，DF 就会更依赖先验来预测动作；成功率仅下降4个百分点，至76%。相比之下，下一帧扩散基线必须把扰动观测当作真值，遭遇分布外误差，成功率为48%。

> $k > 0$ 4% 76% 48%

视频预训练的潜力。

除生成动作外，图4显示 DF 仅凭初始帧还能生成机器人执行任务的视频。这统一了扩散策略 / 模仿学习与视频生成建模，并为未标注视频预训练铺路。

4.5 时间序列预测：DF 是有竞争力的通用序列模型

附录E遵循 [54] 实验设置，展示 DF 在多变量时间序列预测上可与既有扩散 [50] 和 Transformer [51] 方法竞争。

5 讨论

局限性。

当前因果实现基于 RNN。更高分辨率视频或更复杂分布，可能需要按B.1所述使用大型 Transformer。本文未研究 DF 在互联网规模数据与任务上的扩展行为。

结论。

本文提出 DF：对具有逐 token 独立噪声等级的 token 集合去噪的新训练范式。在时间序列上，经 DF 训练的下一 token 模型兼具下一 token 和全序列扩散的优点。我们提出新的采样与引导方案，在序贯决策中显著改善性能。后续可将其用于时间序列生成之外的领域，并扩展到更大数据集。

致谢。

本工作获美国国家科学基金会项目2211259、新加坡 DSTA 项目 DST00OECI20300823（面向标签高效视觉的三维自监督学习）、美国情报高级研究计划局 IARPA 经内政部 / 内政业务中心 DOI/IBC 项目 140D0423C0075，以及 Amazon Science Hub 支持。

参考文献（英文原书目）

- **[1]**
  A. Ajay, Y. Du, A. Gupta, J. Tenenbaum, T. Jaakkola, and P. Agrawal.
  Is conditional generative modeling all you need for decision-making?
  arXiv preprint arXiv:2211.15657, 2022.
- **[2]**
  A. Alexandrov, K. Benidis, M. Bohlke-Schneider, V. Flunkert, J. Gasthaus, T. Januschowski, D. C. Maddix, S. Rangapuram, D. Salinas, J. Schulz, L. Stella, A. C. Türkmen, and Y. Wang.
  Gluonts: Probabilistic and neural time series modeling in python.
  Journal of Machine Learning Research, 21(116):1–6, 2020.
- **[3]**
  C. Berner, G. Brockman, B. Chan, V. Cheung, P. Debiak, C. Dennison, D. Farhi, Q. Fischer, S. Hashme, C. Hesse, R. Józefowicz, S. Gray, C. Olsson, J. Pachocki, M. Petrov, H. P. de Oliveira Pinto, J. Raiman, T. Salimans, J. Schlatter, J. Schneider, S. Sidor, I. Sutskever, J. Tang, F. Wolski, and S. Zhang.
  Dota 2 with large scale deep reinforcement learning.
  CoRR, abs/1912.06680, 2019.
- **[4]**
  A. Blattmann, T. Dockhorn, S. Kulal, D. Mendelevitch, M. Kilian, D. Lorenz, Y. Levi, Z. English, V. Voleti, A. Letts, V. Jampani, and R. Rombach.
  Stable video diffusion: Scaling latent video diffusion models to large datasets, 2023.
- **[5]**
  A. Block, A. Jadbabaie, D. Pfrommer, M. Simchowitz, and R. Tedrake.
  Provable guarantees for generative behavior cloning: Bridging low-level stability and high-level behavior.
  In Thirty-seventh Conference on Neural Information Processing Systems, 2023.
- **[6]**
  T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei.
  Language models are few-shot learners.
  CoRR, abs/2005.14165, 2020.

- **[7]**
  S. H. Chan.
  Tutorial on diffusion models for imaging and vision.
  arXiv preprint arXiv:2403.18103, 2024.
- **[8]**
  M. Chen, A. Radford, R. Child, J. Wu, H. Jun, D. Luan, and I. Sutskever.
  Generative pretraining from pixels.
  In International conference on machine learning, pages 1691–1703. PMLR, 2020.
- **[9]**
  T. Chen.
  On the importance of noise scheduling for diffusion models, 2023.
- **[10]**
  C. Chi, Z. Xu, S. Feng, E. Cousineau, Y. Du, B. Burchfiel, R. Tedrake, and S. Song.
  Diffusion policy: Visuomotor policy learning via action diffusion, 2024.
- **[11]**
  K. Cho, B. van Merrienboer, Ç. Gülçehre, F. Bougares, H. Schwenk, and Y. Bengio.
  Learning phrase representations using RNN encoder-decoder for statistical machine translation.
  CoRR, abs/1406.1078, 2014.
- **[12]**
  J. Chung, K. Kastner, L. Dinh, K. Goel, A. C. Courville, and Y. Bengio.
  A recurrent latent variable model for sequential data.
  Advances in neural information processing systems, 28, 2015.
- **[13]**
  J. Cohen, E. Rosenfeld, and Z. Kolter.
  Certified adversarial robustness via randomized smoothing.
  In international conference on machine learning, pages 1310–1320. PMLR, 2019.
- **[14]**
  E. de Bézenac, S. S. Rangapuram, K. Benidis, M. Bohlke-Schneider, R. Kurle, L. Stella, H. Hasson, P. Gallinari, and T. Januschowski.
  Normalizing kalman filters for multivariate time series analysis.
  In Advances in Neural Information Processing Systems, volume 33, 2020.
- **[15]**
  J. Devlin, M. Chang, K. Lee, and K. Toutanova.
  BERT: pre-training of deep bidirectional transformers for language understanding.
  CoRR, abs/1810.04805, 2018.
- **[16]**
  P. Dhariwal and A. Nichol.
  Diffusion models beat gans on image synthesis.
  CoRR, abs/2105.05233, 2021.
- **[17]**
  C. Feichtenhofer, Y. Li, K. He, et al.
  Masked autoencoders as spatiotemporal learners.
  Advances in neural information processing systems, 35:35946–35958, 2022.
- **[18]**
  J. Fu, A. Kumar, O. Nachum, G. Tucker, and S. Levine.
  D4RL: datasets for deep data-driven reinforcement learning.
  CoRR, abs/2004.07219, 2020.
- **[19]**
  S. Gao, P. Zhou, M.-M. Cheng, and S. Yan.
  Masked diffusion transformer is a strong image synthesizer.
  In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 23164–23173, 2023.
- **[20]**
  C. E. Garcia, D. M. Prett, and M. Morari.
  Model predictive control: Theory and practice—a survey.

Automatica, 25(3):335–348, 1989.

- **[21]**
F. Gers, J. Schmidhuber, and F. Cummins.
Learning to forget: continual prediction with lstm.
In 1999 Ninth International Conference on Artificial Neural Networks ICANN 99. (Conf. Publ. No. 470), volume 2, pages 850–855 vol.2, 1999.

- **[22]**
D. Hafner, T. P. Lillicrap, J. Ba, and M. Norouzi.
Dream to control: Learning behaviors by latent imagination.
CoRR, abs/1912.01603, 2019.

- **[23]**
D. Hafner, T. P. Lillicrap, I. Fischer, R. Villegas, D. Ha, H. Lee, and J. Davidson.
Learning latent dynamics for planning from pixels.
CoRR, abs/1811.04551, 2018.

- **[24]**
T. Hang, S. Gu, C. Li, J. Bao, D. Chen, H. Hu, X. Geng, and B. Guo.
Efficient diffusion training via min-snr weighting strategy, 2024.

- **[25]**
N. Hansen, X. Wang, and H. Su.
Temporal difference learning for model predictive control, 2022.

- **[26]**
W. Harvey, S. Naderiparizi, V. Masrani, C. Weilbach, and F. Wood.
Flexible diffusion modeling of long videos.
In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, Advances in Neural Information Processing Systems, volume 35, pages 27953–27965. Curran Associates, Inc., 2022.

- **[27]**
K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick.
Masked autoencoders are scalable vision learners.
In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 16000–16009, 2022.

- **[28]**
K. He, X. Zhang, S. Ren, and J. Sun.
Deep residual learning for image recognition.
CoRR, abs/1512.03385, 2015.

- **[29]**
J. Ho, A. Jain, and P. Abbeel.
Denoising diffusion probabilistic models.
Advances in Neural Information Processing Systems (NeurIPS), 33:6840–6851, 2020.

- **[30]**
J. Ho, A. Jain, and P. Abbeel.
Denoising diffusion probabilistic models.
CoRR, abs/2006.11239, 2020.

- **[31]**
J. Ho and T. Salimans.
Classifier-free diffusion guidance, 2022.

- **[32]**
J. Ho, T. Salimans, A. Gritsenko, W. Chan, M. Norouzi, and D. J. Fleet.
Video diffusion models, 2022.

- **[33]**
S. Hochreiter and J. Schmidhuber.
Long short-term memory.
Neural Comput., 9(8):1735–1780, nov 1997.

- **[34]**
A. Hu, L. Russell, H. Yeo, Z. Murez, G. Fedoseev, A. Kendall, J. Shotton, and G. Corrado.

Gaia-1: A generative world model for autonomous driving.
arXiv preprint arXiv:2309.17080, 2023.

- **[35]**
  S. Huang, Z. Wang, P. Li, B. Jia, T. Liu, Y. Zhu, W. Liang, and S.-C. Zhu.
  Diffusion-based generation, optimization, and planning in 3d scenes.
  In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 16750–16761, 2023.
- **[36]**
  R. Hyndman, A. B. Koehler, J. K. Ord, and R. D. Snyder.
  Forecasting with Exponential Smoothing: The State Space Approach.
  Springer Science & Business Media, 2008.
- **[37]**
  M. Janner, Y. Du, J. B. Tenenbaum, and S. Levine.
  Planning with diffusion for flexible behavior synthesis.
  Proceedings of the International Conference on Machine Learning (ICML), 2022.
- **[38]**
  A. Katharopoulos, A. Vyas, N. Pappas, and F. Fleuret.
  Transformers are rnns: Fast autoregressive transformers with linear attention.
  CoRR, abs/2006.16236, 2020.
- **[39]**
  L. Ke, J. Wang, T. Bhattacharjee, B. Boots, and S. Srinivasa.
  Grasping with chopsticks: Combating covariate shift in model-free imitation learning for fine manipulation.
  In 2021 IEEE International Conference on Robotics and Automation (ICRA), pages 6185–6191. IEEE, 2021.
- **[40]**
  D. Kingma and R. Gao.
  Understanding diffusion objectives as the elbo with simple data augmentation.
  Advances in Neural Information Processing Systems, 36, 2024.
- **[41]**
  R. G. Krishnan, U. Shalit, and D. Sontag.
  Structured inference networks for nonlinear state space models.
  In AAAI, 2017.
- **[42]**
  G. Lai, W. Chang, Y. Yang, and H. Liu.
  Modeling long- and short-term temporal patterns with deep neural networks.
  CoRR, abs/1703.07015, 2017.
- **[43]**
  M. Laskey, J. Lee, R. Fox, A. Dragan, and K. Goldberg.
  Dart: Noise injection for robust imitation learning.
  In Conference on robot learning, pages 143–156. PMLR, 2017.
- **[44]**
  X. L. Li, J. Thickstun, I. Gulrajani, P. Liang, and T. B. Hashimoto.
  Diffusion-lm improves controllable text generation, 2022.
- **[45]**
  H. Lütkepohl.
  New Introduction to Multiple Time Series Analysis.
  Springer Science & Business Media, 2005.
- **[46]**
  J. E. Matheson and R. L. Winkler.
  Scoring rules for continuous probability distributions.
  Management Science, 22(10):1087–1096, 1976.
- **[47]**
  A. Nichol and P. Dhariwal.
  Improved denoising diffusion probabilistic models.
  CoRR, abs/2102.09672, 2021.

- **[48]**
  B. Peng, E. Alcaide, Q. Anthony, A. Albalak, S. Arcadinho, S. Biderman, H. Cao, X. Cheng, M. Chung, L. Derczynski, X. Du, M. Grella, K. Gv, X. He, H. Hou, P. Kazienko, J. Kocon, J. Kong, B. Koptyra, H. Lau, J. Lin, K. S. I. Mantri, F. Mom, A. Saito, G. Song, X. Tang, J. Wind, S. Woźniak, Z. Zhang, Q. Zhou, J. Zhu, and R.-J. Zhu.
  RWKV: Reinventing RNNs for the transformer era.
  In H. Bouamor, J. Pino, and K. Bali, editors, Findings of the Association for Computational Linguistics: EMNLP 2023, pages 14048–14077, Singapore, Dec. 2023. Association for Computational Linguistics.
- **[49]**
  C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu.
  Exploring the limits of transfer learning with a unified text-to-text transformer.
  CoRR, abs/1910.10683, 2019.
- **[50]**
  K. Rasul, C. Seward, I. Schuster, and R. Vollgraf.
  Autoregressive Denoising Diffusion Models for Multivariate Probabilistic Time Series Forecasting.
  In Proceedings of the 38th International Conference on Machine Learning, volume 139 of Proceedings of Machine Learning Research, 2021.
- **[51]**
  K. Rasul, A.-S. Sheikh, I. Schuster, U. M. Bergmann, and R. Vollgraf.
  Multivariate probabilistic time series forecasting via conditioned normalizing flows.
  In International Conference on Learning Representations, 2021.
- **[52]**
  D. Ruhe, J. Heek, T. Salimans, and E. Hoogeboom.
  Rolling diffusion models.
  arXiv preprint arXiv:2402.09470, 2024.
- **[53]**
  T. Salimans and J. Ho.
  Progressive distillation for fast sampling of diffusion models.
  CoRR, abs/2202.00512, 2022.
- **[54]**
  D. Salinas, M. Bohlke-Schneider, L. Callot, R. Medico, J. Gasthaus, and R. Medico.
  High-dimensional multivariate forecasting with low-rank gaussian copula processes.
  In NeurIPS, 2019.
- **[55]**
  D. Salinas, V. Flunkert, J. Gasthaus, and T. Januschowski.
  Deepar: Probabilistic forecasting with autoregressive recurrent networks.
  International Journal of Forecasting, 36(3):1181–1191, 2020.
- **[56]**
  D. Salinas, V. Flunkert, J. Gasthaus, and T. Januschowski.
  Deepar: Probabilistic forecasting with autoregressive recurrent networks.
  International Journal of Forecasting, 36(3):1181–1191, 2020.
- **[57]**
  J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli.
  Deep unsupervised learning using nonequilibrium thermodynamics.
  In Proceedings of the International Conference on Machine Learning (ICML), 2015.
- **[58]**
  J. Song, C. Meng, and S. Ermon.
  Denoising diffusion implicit models.
  CoRR, abs/2010.02502, 2020.
- **[59]**
  B. Tang and D. S. Matteson.
  Probabilistic transformer for time series analysis.
  In A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, editors, Advances in Neural Information Processing Systems, 2021.
- **[60]**
  H. Touvron, P. Bojanowski, M. Caron, M. Cord, A. El-Nouby, E. Grave, A. Joulin, G. Synnaeve, J. Verbeek, and H. Jégou.

Resmlp: Feedforward networks for image classification with data-efficient training.
CoRR, abs/2105.03404, 2021.

- **[61]**
  A. Van den Oord, N. Kalchbrenner, L. Espeholt, O. Vinyals, A. Graves, et al.
  Conditional image generation with pixelcnn decoders.
  Advances in neural information processing systems, 29, 2016.
- **[62]**
  R. van der Weide.
  Go-garch: A multivariate generalized orthogonal garch model.
  Journal of Applied Econometrics, 17(5):549–564, 2002.
- **[63]**
  C. Wei, K. Mangalam, P.-Y. Huang, Y. Li, H. Fan, H. Xu, H. Wang, C. Xie, A. Yuille, and C. Feichtenhofer.
  Diffusion models as masked autoencoders.
  In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 16284–16294, 2023.
- **[64]**
  G. Williams, A. Aldrich, and E. Theodorou.
  Model predictive path integral control using covariance variable importance sampling.
  arXiv preprint arXiv:1509.01149, 2015.
- **[65]**
  R. J. Williams and D. Zipser.
  A Learning Algorithm for Continually Running Fully Recurrent Neural Networks.
  Neural Computation, 1(2):270–280, 06 1989.
- **[66]**
  T. Wu, Z. Fan, X. Liu, Y. Gong, Y. Shen, J. Jiao, H.-T. Zheng, J. Li, Z. Wei, J. Guo, N. Duan, and W. Chen.
  Ar-diffusion: Auto-regressive diffusion model for text generation, 2023.
- **[67]**
  T. Wu, Z. Fan, X. Liu, H.-T. Zheng, Y. Gong, J. Jiao, J. Li, J. Guo, N. Duan, W. Chen, et al.
  Ar-diffusion: Auto-regressive diffusion model for text generation.
  Advances in Neural Information Processing Systems, 36:39957–39974, 2023.
- **[68]**
  T. Yan, H. Zhang, T. Zhou, Y. Zhan, and Y. Xia.
  Scoregrad: Multivariate probabilistic time series forecasting with continuous energy-based generative models, 2021.
- **[69]**
  W. Yan, D. Hafner, S. James, and P. Abbeel.
  Temporally consistent transformers for video generation, 2023.
- **[70]**
  R. Yang, P. Srivastava, and S. Mandt.
  Diffusion probabilistic modeling for video generation.
  Entropy, 25(10):1469, 2023.
- **[71]**
  S. Yao, D. Yu, J. Zhao, I. Shafran, T. L. Griffiths, Y. Cao, and K. Narasimhan.
  Tree of thoughts: Deliberate problem solving with large language models, 2023.
- **[72]**
  H. Yu, N. Rao, and I. S. Dhillon.
  Temporal regularized matrix factorization.
  CoRR, abs/1509.08333, 2015.

附录A 理论论证

本节为 DF 训练提供理论论证，主要贡献如下。

• 证明训练方法优化数据平均对数似然 ELBO 的重新加权形式。先给出一般结果（定理A.1），再特化为高斯扩散（推论A.2）。结果各项能够解耦；当潜变量和模型具有充分表达力时，重新加权不改变最优解。

• 对任意噪声等级序列分布，其期望似然均有由非负项之和组成的下界；这些项重加权后正是 DF 训练所优化的项。因此，若网络足够有表达力并能将所有项降至各自最小值，DF 就同时优化所有噪声等级序列似然的有效替代目标。

先针对一般马尔可夫正向过程 q 与生成模型 pθ 建立 ELBO，再特化为高斯扩散，恢复本文损失。正向过程 q 定义如下。

> $q(\cdot)$ $p_{\boldsymbol{\theta}}(\cdot)$ $q(\cdot)$

$$q\left(\mathbf{x}^{1:K}|\mathbf{x}^{0}\right)=\prod_{k=1}^{K}q\left(\mathbf{x}^{k}|\mathbf{x}^{k-1}\right), \tag{A.1}$$

参数化概率模型定义如下。

$$p_{\boldsymbol{\theta}}\left(\left(\left(\mathbf{x}_t^k\right)_{1\le k\le K},\mathbf{z}_t\right)_{t\ge 1}\right) \tag{A.2}$$

假设 pθ 满足以下马尔可夫性质。

> $p_{\boldsymbol{\theta}}$

$$p_{\boldsymbol{\theta}}\left(\mathbf{z}_t,\mathbf{x}_t^{k_t}|\mathbf{z}_{1:t-1},\left(\mathbf{x}_s^{k_s}\right)_{1\le s<t}\right)=p_{\boldsymbol{\theta}}\left(\mathbf{z}_t,\mathbf{x}^{k_t}|\mathbf{z}_{t-1}\right) \tag{A.3}$$

即给定历史时，潜编码 z_{t-1} 是 x_t^{kt} 的充分统计量。若潜变量的条件分布为狄拉克 δ 分布，则称 pθ 具有确定性潜变量。

> $\mathbf{z}_{t-1}$ $\mathbf{x}^{k_t}$ $p_{\boldsymbol{\theta}}$ $p_{\boldsymbol{\theta}}\left(\mathbf{z}_t|\mathbf{z}_{1:t-1},\left(\mathbf{x}_s^{k_s}\right)_{1\le s<t},\mathbf{x}_t^{k_t}\right)$

说明 1.

为了使确定性潜变量对应有效概率分布，不能把 z_t 视为一个孤立变量，而应视为由时间 t 及噪声历史 $k_{1:t}$ 索引的变量集合 $z_t(k_{1:t})$。简单地把完整噪声历史与相应噪声 token 一起存入 z，就可平凡地得到确定性潜变量。必须以噪声历史索引，是因为若同时对同一时刻不同噪声等级的观测作条件化，除非各等级完全相同，否则潜变量条件分布会失去明确定义，pθ 也就不对应联合概率测度。下文定理允许这种对噪声历史的依赖，但为减少符号混乱，将其省略。

> $p_{\boldsymbol{\theta}}$ $\mathbf{z}_t$ $\mathbf{z}_t(k_{1:t})$ $t\in[T]$ $k_{1:t}\in\{0,1,\ldots,K\}^t$ $\mathbf{z}_t(k_{1:t})=\left(k_{1:t},\left(\mathbf{x}_s^{k_s}\right)_{1\le s\le t}\right.$ $\mathbf{z}_t(k_{1:t})$ $k_{1:t}$
> $\left.p_{\boldsymbol{\theta}}\left(\mathbf{z}_t|\left(\left(\mathbf{x}_s^{k_s}\right)_{1\le s\le t},\left(\mathbf{x}_s^{k_s'}\right)_{1\le s\le t}\right)\right)\right.$ $k_s=k_s'$ $1\le s\le t$ $p_{\boldsymbol{\theta}}$ $\mathbf{z}_t(k_{1:t})$ $k_{1:t}$ $k_{1:t}$

A.1 主要结果

下面给出主要定理：当各 k_t 均匀采样，并按式(A.1)用 q 加噪时，部分噪声序列期望对数似然具有一个 ELBO。此表述不要求 q 或 pθ 的具体形式；下一节再特化为高斯扩散。

> $\left(\mathbf{x}_t^{k_t}\right)_{1\le t\le T}$ k_t $\mathbf{x}_t^{k_t}$ $q(\cdot)$ $q(\cdot)$ $p_{\boldsymbol{\theta}}$

定理 A.1.

固定干净序列 $x_{1:T}^0$。带随机噪声等级的正向过程期望定义如下。

$$\mathbf{x}_{1:T}^0 \qquad k_{1:T}$$

$$\mathop{\mathbb{E}}_{\text{forward}}[\cdot] := \mathop{\mathbb{E}}_{k_1,\ldots,k_T \overset{\text{unif}}{\sim} [K]} \mathop{\mathbb{E}}_{\mathbf{x}_s^{k_s} \sim q\left(\mathbf{x}_s^{k_s} | \mathbf{x}_s^0\right), 1 \le s \le T} [\cdot], \tag{A.4}$$

给定噪声等级及相应噪声观测，pθ 下潜变量的条件期望定义如下。

$$p_{\boldsymbol{\theta}}(\cdot) \qquad k_{1:T}, \left(\mathbf{x}_s^{k_t}\right)_{1 \le t \le T}$$

$$\mathop{\mathbb{E}}_{p, \mathbf{z}_{1:T}}[\cdot] := \mathop{\mathbb{E}}_{\mathbf{z}_S \sim p\left(\mathbf{z}_S | \mathbf{z}_{s-1}, \mathbf{x}_s^{k_s}\right), s \le T} \left[\cdot \mid k_{1:T}, \left(\mathbf{x}_t^{k_t}\right)_{1 \le t \le T}\right] \tag{A.5}$$

只要 pθ 满足马尔可夫性质，就有以下结论。

$$p_{\boldsymbol{\theta}}$$

$$\mathop{\mathbb{E}}_{\text{forward}}\left[\ln p_{\boldsymbol{\theta}}\left(\left(\mathbf{x}_t^{k_t}\right)_{1 \le t \le T}\right)\right] \ge C\left(\mathbf{x}_{1:T}^0\right)$$
$$+ \mathop{\mathbb{E}}_{\text{forward}} \mathop{\mathbb{E}}_{p, \mathbf{z}_{1:T}}\left[\sum_{t=1}^{T}\left(\frac{1}{K+1}\ln p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^0 | \mathbf{x}_t^1, \mathbf{z}_{t-1}\right) + \sum_{j=2}^{K} \frac{j}{K+1} D_{\mathbb{KL}}\left(q\left(\mathbf{x}_t^{j-1} | \mathbf{x}_t^j, \mathbf{x}_t^0\right) \| p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^j | \mathbf{x}_t^{j-1}, \mathbf{z}_{t-1}\right)\right)\right)\right],$$

其中 C 仅依赖无噪声数据。如果潜变量确定，即相应条件分布为狄拉克分布，那么当且仅当所列 q 条件分布与 pθ 条件分布相同，也就是变分近似精确时，下界取等号。（底本此处“inequality with inequality”按上下文译为等号条件。）

$$C(\mathbf{x}_{1:T}^0) \qquad \mathbf{x}_{1:T}^0 \qquad p_{\boldsymbol{\theta}}(\mathbf{z}_t | \mathbf{z}_{t-1}, \mathbf{x}_t^{k_t}) \qquad q(\mathbf{x}_t^{k_t+1:T} | \mathbf{x}_t^{k_t}) \equiv p_{\boldsymbol{\theta}}(\mathbf{x}_t^{k_t+1:T} | \mathbf{x}_t^{k_t}, \mathbf{z}_{t-1})$$

证明见A.2。值得注意的是，证明仅涉及两次不等式：第一次在潜变量确定时取等号；第二次当且仅当变分近似精确时取等号。因此，该 ELBO 的紧致性说明定理A.1中的表达式是优化似然的较强替代目标。

$$q\left(\mathbf{x}_t^{k_t+1:T} | \mathbf{x}_t^{k_t}\right) \equiv p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^{k_t+1:T} | \mathbf{x}_t^{k_t}, \mathbf{z}_{t-1}\right)$$

A.1.1 特化为高斯扩散

现在将定理A.1特化为高斯扩散。先考虑实现实际采用的 x 预测形式；正文使用的 ε 预测形式可类似推导（清晰说明见 [7] 第2节）。将 DDPM [29,7] 的标准似然和 KL 散度计算应用于定理A.1，直接得到如下定理。

$$\mathbf{x} \qquad \epsilon$$

推论 A.2.

令正向高斯过程为下式。

$$q\left(\mathbf{x}^{k+1} | \mathbf{x}_t^k\right) = \mathcal{N}\left(\mathbf{x}^k, \sqrt{1-\beta_k}\mathbf{x}^{k-1}, \beta_k \mathbf{I}\right), \tag{A.6}$$

定义 $\alpha_k = 1 - \beta_k$，$\bar{\alpha}_k$ 为各 α 的累乘。假设逆过程参数化为均值 μθ、方差 σ_j^2 的高斯分布，并令均值满足下式。

$$\alpha_k = (1-\beta_k) \qquad \bar{\alpha}_k = \prod_{j=1}^{k} \alpha_j \qquad p_{\boldsymbol{\theta}}(\mathbf{x}_t^j | \mathbf{x}_t^{j+1}, \mathbf{z}_{t-1}) = \mathcal{N}(\mu_{\boldsymbol{\theta}}(\mathbf{x}_t^{j+1}, \mathbf{z}_{t-1}, j), \sigma_j^2)$$

$$\mu_{\boldsymbol{\theta}}\left(\mathbf{x}_t^j, \mathbf{z}_{t-1}, j\right) = \frac{\left(1-\bar{\alpha}_{j-1}\right)\sqrt{\alpha_j}}{1-\bar{\alpha}_j}\mathbf{x}_t^j + \frac{\left(1-\alpha_j\right)\sqrt{\bar{\alpha}_{j-1}}}{1-\bar{\alpha}_j}\hat{\mathbf{x}}_{\boldsymbol{\theta}}\left(\mathbf{x}_t^j, \mathbf{z}_{t-1}, j\right), \quad \sigma_j^2 := \frac{\left(1-\alpha_j\right)\left(1-\sqrt{\bar{\alpha}_{j-1}}\right)}{1-\bar{\alpha}_j}.$$

只要 pθ 满足马尔可夫性质，就得到下列界。

> $p_{\boldsymbol{\theta}}$

$$\begin{aligned}\mathop{\mathbb{E}}_{\text{forward}}\left[\ln p_{\boldsymbol{\theta}}\left(\left(\mathbf{x}_t^{k_t}\right)_{1\le t\le T}\right)\right]+C\left(\mathbf{x}_{1:T}^0\right) &\ge \mathop{\mathbb{E}}_{\text{forward}}\mathop{\mathbb{E}}_{p,\mathbf{z}_{1:T}}\left[\sum_{t=1}^{T}\frac{j}{K+1}\sum_{j=1}^{K}c_j\left\|\hat{\mathbf{x}}_{\boldsymbol{\theta}}^0\left(\mathbf{x}_t^j,\mathbf{z}_{t-1},j\right)-\mathbf{x}_t^0\right\|^2\right]\\ &= \mathop{\mathbb{E}}_{\text{forward}}\mathop{\mathbb{E}}_{p,\mathbf{z}_{1:T}}\left[\sum_{t=1}^{T}\mathbf{1}\{k_t\ge 1\}\cdot k_t c_{k_t}\left\|\hat{\mathbf{x}}_{\boldsymbol{\theta}}^0\left(\mathbf{x}_t^{k_t},\mathbf{z}_{t-1},k_t\right)-\mathbf{x}_t^0\right\|^2\right],\end{aligned}$$

其中 c_j 的定义为下列公式中给出的正权重，即 $(1-\alpha_j)^2\bar{\alpha}_{j-1}$ 除以 $2\sigma^2(1-\bar{\alpha}_j)^2$。

> $$c_j=\frac{(1-\alpha_j)^2\bar{\alpha}_{j-1}}{2\sigma^2(1-\bar{\alpha}_j)^2}$$

证明.

第一项不等式来自扩散 x 预测的标准计算（见 [7] 第2.7节及其引文）；第二项把对 j 的求和改写成 k_t 在 {0,...,K} 上均匀采样的期望。证毕。

> $\mathbf{x}$ j $k_t\overset{\text{unif}}{\sim}\{0,1,\ldots,K\}$

有以下几点说明。

• 推论A.2同样可写成 ε 预测或 v 预测形式，因为三者由仿射变换相联。

> ϵ $\mathbf{v}$

• 定义理想潜变量，包含所有过去噪声 token 及其噪声等级；它是原潜变量的充分统计量，因此原潜变量可视为对完整历史的压缩。对推论A.2两侧再取真实数据分布 q 的期望，并在所有可能函数逼近器上取下确界，所列最优平方误差均等于“给定全部噪声历史时干净 token 的条件方差”。这说明：当潜变量和 pθ 足够有表达力时，可分别优化推论中各时刻的项；不存在跨项耦合，权重就不影响最优解，因而可以忽略。对应的三行下确界等式完整保留如下。

> $\tilde{\mathbf{z}}_{t-1}$ $\left(\mathbf{x}_t^{k_t}\right)$ k_t $\mathbf{z}_{t-1}$ $\hat{\mathbf{x}}_{\boldsymbol{\theta}}^0\left(\mathbf{x}_t^{k_t},\mathbf{z}_{t-1},k_t\right)=\hat{\mathbf{x}}_{\boldsymbol{\theta}}^0\left(\mathbf{x}_t^{k_t},\bar{\mathbf{z}}_{t-1},k_t\right)$ $\mathbf{z}_{t-1}$ $\bar{\mathbf{z}}_{t-1}$ $\mathbf{x}_{1:T}\sim q$ $\hat{\mathbf{x}}_\theta^0$
>
> $\inf_{p_\theta}\mathbb{E}_q\mathop{\mathbb{E}}_{\text{forward}}\mathbb{E}_{p,\mathbf{z}_{1:T}}\left\|\hat{\mathbf{x}}_{\boldsymbol{\theta}}^0\left(\mathbf{x}_t^{k_t},\mathbf{z}_{t-1},k_t\right)-\mathbf{x}_t^0\right\|^2$ $=\inf_{p_\theta}\mathbb{E}_q\mathop{\mathbb{E}}_{\text{forward}}\mathbb{E}_{p,\mathbf{z}_{1:T}}\left\|\hat{\mathbf{x}}_{\boldsymbol{\theta}}^0\left(\mathbf{x}_t^{k_t},\bar{\mathbf{z}}_{t-1}\right)-\mathbf{x}_t^0\right\|^2$
>
> $=\mathbf{Var}_q[\mathbf{x}_t^0\mid\left(\mathbf{x}_s^{k_s}\right)_{1\le s\le t'},k_1,\ldots,k_t].$ p_θ

• 因此，可对推论A.2的目标求梯度来优化 ELBO，并按需要删除或重缩放权重。因为潜变量确定，通过其期望反向传播很直接。这论证了式(3.1)和算法1的正确性。

> $\mathbb{E}_{p,\mathbf{z}_{1:T}}$

A.1.2 覆盖所有子序列

定理A.1说明：忽略重加权，DF 优化均匀噪声等级下期望对数似然的有效 ELBO。对其证明稍作修改，便可推广到任意、可能在时间上相关的噪声等级序列。

定理 A.3.

令 $\mathcal{D}$ 为 $[K]^T$ 上任意分布，$P_t(j|k_{1:t-1})$ 为给定过去等级后当前等级取 j 的条件概率。固定干净序列，将随机等级正向过程的期望定义如下。

$\mathcal{D}$ $[K]^T$ $P_t(j|\,k_{1:t-1}) := \Pr_{\mathcal{D}}[k_t = j|\,k_{1:t-1}]$ $\mathbf{x}_{1:T}{}^0$ $k_{1:T}$

$$\mathop{\mathbb{E}}_{\text{forward},\mathcal{D}}[\cdot] := \mathop{\mathbb{E}}_{k_1,\ldots,k_T\sim\mathcal{D}}\ \mathop{\mathbb{E}}_{\mathbf{x}_s^{k_s}\sim q\left(\mathbf{x}_s^{k_s}|\mathbf{x}_s^0\right),1\le s\le T}[\cdot], \tag{A.7}$$

在给定等级及其噪声观测条件下，pθ 潜变量的期望定义如下。

$p_{\boldsymbol{\theta}}(\cdot)$ $k_{1:T}, (\mathbf{x}_s^{k_t})_{1\le t\le T}$

$$\mathop{\mathbb{E}}_{p,\mathbf{z}_{1:T}}[\cdot] := \mathop{\mathbb{E}}_{\mathbf{z}_S\sim p\left(\mathbf{z}_S|\mathbf{z}_{s-1},\mathbf{x}_s^{k_s}\right),s\le T}\left[\cdot|\,k_{1:T},\left(\mathbf{x}_t^{k_t}\right)_{1\le t\le T}\right] \tag{A.8}$$

只要 pθ 满足马尔可夫性质，就有下列界。

$p_{\boldsymbol{\theta}}$

$$\mathop{\mathbb{E}}_{\text{forward},\mathcal{D}}\left[\ln p_{\boldsymbol{\theta}}\left(\left(\mathbf{x}_t^{kt}\right)_{1\le t\le T}\right)\right] \ge C\left(\mathbf{x}_{1:T}^0\right) + \mathop{\mathbb{E}}_{\text{forward},\mathcal{D}}\mathop{\mathbb{E}}_{p,\mathbf{z}_{1:T}}\left[\sum_{t=1}^{T}\Xi_t\right], \text{where}$$
$$\Xi_t := \left(P_t\left(1|\,k_{1:t-1}\right)\ln p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^0|\,\mathbf{x}_t^1,\mathbf{z}_{t-1}\right) + \sum_{j=2}^{K} jP_t\left(j|\,k_{1:t-1}\right)D_{\mathbb{KL}}\left(q\left(\mathbf{x}_t^{j-1}|\,\mathbf{x}_t^j,\mathbf{x}_t^0\right)\|\,p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^j|\,\mathbf{x}_t^{j-1},\mathbf{z}_{t-1}\right)\right)\right),$$

其中 C 仅依赖无噪声数据；当且仅当（a）潜变量确定，（b）所列 q 与 pθ 条件分布一致，即变分近似紧致时，该不等式取等号。

$C(\mathbf{x}_{1:T}^0)$ $\mathbf{x}_{1:T}{}^0$ $p_{\boldsymbol{\theta}}(\mathbf{z}_t|\mathbf{z}_{t-1},\mathbf{x}_t^{k_t})$ $q(\mathbf{x}_t^{k_t+1:T}|\,\mathbf{x}_t^{k_t}) \equiv p_{\boldsymbol{\theta}}(\mathbf{x}_t^{k_t+1:T}|\,\mathbf{x}_t^{k_t},\mathbf{z}_{t-1})$

尤其对推论A.2的高斯情形，有以下形式。

$$\mathop{\mathbb{E}}_{\text{forward},\mathcal{D}}\left[\ln p_{\boldsymbol{\theta}}\left(\left(\mathbf{x}_t^{k_t}\right)_{1\le t\le T}\right)\right] + C\left(\mathbf{x}_{1:T}^0\right) \ge \mathop{\mathbb{E}}_{\text{forward},\mathcal{D}}\mathop{\mathbb{E}}_{p,\mathbf{z}_{1:T}}\left[\sum_{t=1}^{T}\mathbf{1}\{k_t\ge 1\}k_t c_{k_t}\|\hat{\mathbf{x}}_{\boldsymbol{\theta}}^0\left(\mathbf{x}_t^{k_t},\mathbf{z}_{t-1},k_t\right)-\mathbf{x}_t^0\|^2\right],$$

最重要的情形是把 $\mathcal{D}$ 限制为某条固定噪声等级序列，即 $[K]^T$ 上的狄拉克分布。此时除 $j=k_t$ 外，各 P_t 为0，所以训练目标本身未必是该固定序列期望对数似然的下界。但真正下界中的项，经重加权后是训练目标所优化各项的子集。由推论A.2后的说明，足够有表达力的网络可以同时优化全部损失项。因此，对充分表达的神经网络，优化式(3.1)是最大化所有可能噪声序列似然的有效替代。

$\mathcal{D}$ $k_1,\ldots,k_T$ $[K]^T$ $P_t\left(j|\,k_{1:t-1}\right)=0$ $j=k_t$ $\mathbb{E}_{\text{forward},\mathcal{D}}\left[\ln p_{\boldsymbol{\theta}}\left(\left(\mathbf{x}_t^{k_t}\right)_{1\le t\le T}\right)\right]$

A.2 定理A.1的证明

用 $\mathbb{E}_{<t}$ 简记以下复合期望：对过去均匀噪声等级、由正向过程生成的噪声 token，以及模型条件潜变量依次取期望。先证明以下命题。

$\mathbb{E}_{<t}[\cdot]$ $\mathbb{E}_{k_{1:s}\overset{\text{unif}}{\sim}[K]}\ \mathbb{E}_{\mathbf{x}_s^{k_s}\sim q\left(\mathbf{x}_s^{k_s}|\mathbf{x}_s^0\right),1\le s\le t-1}\ \mathbb{E}_{\mathbf{z}_S\sim p\left(\mathbf{z}_S|\mathbf{z}_{s-1},\mathbf{x}_s^{k_s}\right),s\le t}[\cdot]$

命题 1 (展开潜变量).

下列下界成立。

$$\mathop{\mathbb{E}}_{\text{forward}}\left[\ln p_{\boldsymbol{\theta}}\left(\left(\mathbf{x}_t^{k_t}\right)_{1\le t\le T}\right)\right] \ge \sum_{t=1}^{T}\mathop{\mathbb{E}}_{<t}\ \mathop{\mathbb{E}}_{k_t\overset{\text{unif}}{\sim}\{0,1,\ldots,K\}}\ \mathop{\mathbb{E}}_{\mathbf{x}_t^{k_t}\sim q\left(\mathbf{x}_t^{k_t}|\mathbf{x}_t^0\right)}\left[\ln p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^{k_t}|\mathbf{z}_{t-1}\right)\right], \tag{A.9}$$

此外，如果各潜变量的条件分布为狄拉克分布，即潜变量确定，那么该下界取等号。

$\mathbf{z}_s \sim p(\mathbf{z}_s|\mathbf{z}_{s-1},\mathbf{x}_s^{k_s})$

证明.

固定噪声等级序列 $k_{1:t}$。有以下恒等关系。

> $k_{1:T}$

$$
\begin{aligned}
p_{\boldsymbol{\theta}}\left(\left(\mathbf{x}_t^{k_t}\right)_{1\le t\le T}\right) &= \int_{\mathbf{z}_{1:T}} \prod_{t=1}^{T} p\left(\mathbf{x}_t^{k_t}, \mathbf{z}_t \middle| \left(\mathbf{x}_s^{k_s}, \mathbf{z}_s\right)_{s<t}\right) \\
&= \int_{\mathbf{z}_{1:T}} \prod_{t=1}^{T} p\left(\mathbf{x}_t^{k_t}, \mathbf{z}_t \middle| \mathbf{z}_{t-1}\right) && \text{(马尔可夫性质)} \\
&= \int_{\mathbf{z}_{1:T}(\mathbf{k})} \prod_{t=1}^{T} p\left(\mathbf{z}_t \middle| \mathbf{z}_{t-1}, \mathbf{x}_t^{k_t}\right) p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^{k_t} \middle| \mathbf{z}_{t-1}\right) \\
&= \mathop{\mathbb{E}}_{\mathbf{z}_S \sim p\left(\mathbf{z}_S \middle| \mathbf{z}_{s-1}, \mathbf{x}_S^{k_s}\right), s\le T} \prod_{t=1}^{T} p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^{k_t} \middle| \mathbf{z}_{t-1}\right). && \text{(重要性采样)}
\end{aligned}
$$

由 Jensen 不等式可得下式。

$$
\ln p_{\boldsymbol{\theta}}\left(\left(\mathbf{x}_t^{k_t}\right)_{1\le t\le T}\right) \ge \mathop{\mathbb{E}}_{\mathbf{z}_S \sim p\left(\mathbf{z}_S \middle| \mathbf{z}_{s-1}, \mathbf{x}_S^{k_s}\right), s\le T} \sum_{t=1}^{T} \ln p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^{k_t} \middle| \mathbf{z}_{t-1}\right) = \mathop{\mathbb{E}}_{p, \mathbf{z}_{1:T}}\left[\sum_{t=1}^{T} \ln p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^{k_t} \middle| \mathbf{z}_{t-1}\right)\right],
$$

当潜变量条件分布为狄拉克分布时，不等式取等号。对上式两侧取正向过程期望，再用潜变量的马尔可夫性质，得到以下结论。

> $p_{\boldsymbol{\theta}}(\mathbf{z}_s|\mathbf{z}_{s-1}, \mathbf{x}_s^{k_s})$　　$\mathbb{E}_{\text{forward}}$

$$
\begin{aligned}
\mathop{\mathbb{E}}_{\text{forward}}\left[\ln p_{\boldsymbol{\theta}}\left(\left(\mathbf{x}_t^{k_t}\right)_{1\le t\le T}\right)\right] &\ge \mathop{\mathbb{E}}_{\text{forward}} \mathop{\mathbb{E}}_{p, \mathbf{z}_{1:T}}\left[\sum_{t=1}^{T} \ln p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^{k_t} \middle| \mathbf{z}_{t-1}\right)\right] \\
&= \sum_{t=1}^{T} \mathop{\mathbb{E}}_{<t} \mathop{\mathbb{E}}_{k_t \overset{\text{unif}}{\sim} \{0,1,\ldots,K\}} \mathop{\mathbb{E}}_{\mathbf{x}_t^{k_t} \sim q\left(\mathbf{x}_t^{k_t} \middle| \mathbf{x}_t^0\right)} \left[\ln p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^{k_t} \middle| \mathbf{z}_{t-1}\right)\right].
\end{aligned}
$$

■

现在展开上一命题得到的各项。

命题 2 (ELBO w.r.t. q).

有下式。

$$
\mathop{\mathbb{E}}_{\mathbf{x}_t^{k_t} \sim q\left(\mathbf{x}_t^{k_t} \middle| \mathbf{x}_t^0\right)} \left[\ln p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^{k_t} \middle| \mathbf{z}_{t-1}\right)\right] \ge C_1\left(\mathbf{x}_0, k_t\right) + \left[\mathop{\mathbb{E}}_{\mathbf{x}_t{}^{k_t:K} \sim q\left(\mathbf{x}_t{}^{k_t:K} \middle| \mathbf{x}_t{}^0\right)} \ln \frac{p_{\boldsymbol{\theta}}\left(\mathbf{x}_t{}^{k_t:K} \middle| \mathbf{z}_{t-1}\right)}{q\left(\mathbf{x}_t{}^{k_t+1:K} \middle| \mathbf{x}_t{}^0\right)}\right].
$$

其中 C_1 仅依赖干净数据与 k_t。该不等式当且仅当所列 q 与 pθ 条件分布相同，即变分近似精确时取等号。

> $C_1(\mathbf{x}_0, k_t)$　　$\mathbf{x}_0$　　k_t　　$q(\mathbf{x}_t^{k_t+1:T} | \mathbf{x}_t^{k_t}) \equiv p_{\boldsymbol{\theta}}(\mathbf{x}_t^{k_t+1:T} | \mathbf{x}_t^{k_t}, \mathbf{z}_{t-1})$

证明.

我们有以下推导。

$$
\begin{aligned}
&\mathop{\mathbb{E}}_{\mathbf{x}_t^{k_t} \sim q\left(\mathbf{x}_t^{k_t} \middle| \mathbf{x}_t^0\right)} \left[\ln p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^{k_t} \middle| \mathbf{z}_{t-1}\right)\right] \\
&= \mathop{\mathbb{E}}_{\mathbf{x}_t^{k_t} \sim q\left(\mathbf{x}_t^{k_t} \middle| \mathbf{x}_t^0\right)} \left[\ln \int p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^{k_t:K} \middle| \mathbf{z}_{t-1}\right) d\mathbf{x}_t^{k_t+1:K}\right] \\
&= \mathop{\mathbb{E}}_{\mathbf{x}_t^{k_t} \sim q\left(\mathbf{x}_t^{k_t} \middle| \mathbf{x}_t^0\right)} \left[\ln\left(\mathop{\mathbb{E}}_{\mathbf{x}_t{}^{k_t+1:K} \sim q\left(\mathbf{x}_t{}^{k_t+1:K} \middle| \mathbf{x}_t{}^{k_t}\right)} \left[\frac{p_{\boldsymbol{\theta}}\left(\mathbf{x}_t{}^{k_t:K} \middle| \mathbf{z}_{t-1}\right)}{q\left(\mathbf{x}_t{}^{k_t+1:K} \middle| \mathbf{x}_t{}^{k_t}\right)}\right]\right)\right]
\end{aligned}
$$

$$\geq \mathop{\mathbb{E}}_{\mathbf{x}_t^{k_t} \sim q\left(\mathbf{x}_t^{k_t} \mid \mathbf{x}_t^0\right)}\left[\mathop{\mathbb{E}}_{\mathbf{x}_t{}^{k_t+1:K} \sim q\left(\mathbf{x}_t{}^{k_t+1:K} \mid \mathbf{x}_t{}^{k_t}\right)}\left[\ln \frac{p_{\boldsymbol{\theta}}\left(\mathbf{x}_t{}^{k_t:K} \mid \mathbf{z}_{t-1}\right)}{q\left(\mathbf{x}_t{}^{k_t+1:K} \mid \mathbf{x}_t{}^{k_t}\right)}\right]\right] \quad \text{((Jensen不等式))}$$

$$= \mathop{\mathbb{E}}_{\mathbf{x}_t{}^{k_t:K} \sim q\left(\mathbf{x}_t{}^{k_t:K} \mid \mathbf{x}_t{}^0\right)}\left[\ln \frac{p_{\boldsymbol{\theta}}\left(\mathbf{x}_t{}^{k_t:K} \mid \mathbf{z}_{t-1}\right)}{q\left(\mathbf{x}_t{}^{k_t+1:K} \mid \mathbf{x}_t{}^{k_t}\right)}\right] \quad \text{(以下分布的马尔可夫性质 } q(\cdot)\text{)}$$

$$= C_1\left(\mathbf{x}_0, k_t\right) + \left[\mathop{\mathbb{E}}_{\mathbf{x}_t{}^{k_t:K} \sim q\left(\mathbf{x}_t{}^{k_t:K} \mid \mathbf{x}_t{}^0\right)} \ln \frac{p_{\boldsymbol{\theta}}\left(\mathbf{x}_t{}^{k_t:K} \mid \mathbf{z}_{t-1}\right)}{q\left(\mathbf{x}_t{}^{k_t+1:K} \mid \mathbf{x}_t{}^0\right)}\right],$$

其中所定义 C_1 是正向过程下对数密度比的期望，只依赖干净数据和 k_t。为检验等号条件，若所列 q 与 pθ 条件分布完全相同，则有下列恒等式。

> $C_1(\mathbf{x}_0, k_t) = \mathbb{E}_{\mathbf{x}_t{}^{k_t:K} \sim q(\mathbf{x}_t{}^{k_t:K} \mid \mathbf{x}_t{}^0)}\left[\ln \frac{q(\mathbf{x}_t{}^{k_t+1:K} \mid \mathbf{x}_t{}^0)}{q(\mathbf{x}_t{}^{k_t+1:K} \mid \mathbf{x}_t{}^{k_t})}\right]$ $\mathbf{x}_0$ k_t $q(\mathbf{x}_t^{k_t+1:T} \mid \mathbf{x}_t^{k_t}) \equiv p_{\boldsymbol{\theta}}(\mathbf{x}_t^{k_t+1:T} \mid \mathbf{x}_t^{k_t}, \mathbf{z}_{t-1})$

$$\mathop{\mathbb{E}}_{\mathbf{x}_t{}^{k_t+1:K} \sim q\left(\mathbf{x}_t{}^{k_t+1:K} \mid \mathbf{x}_t{}^{k_t}\right)}\left[\ln \frac{p_{\boldsymbol{\theta}}\left(\mathbf{x}_t{}^{k_t:K} \mid \mathbf{z}_{t-1}\right)}{q\left(\mathbf{x}_t{}^{k_t+1:K} \mid \mathbf{x}_t{}^{k_t}\right)}\right] = \ln p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^{k_t} \mid \mathbf{z}_{t-1}\right) + \mathop{\mathbb{E}}_{\mathbf{x}_t{}^{k_t+1:K} \sim q\left(\mathbf{x}_t{}^{k_t+1:K} \mid \mathbf{x}_t{}^{k_t}\right)}\left[\ln p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^{k_t+1:K} \mid \mathbf{z}_{t-1}, \mathbf{x}_t^{k_t}\right)\right]$$

因为 log 严格凹，所列期望中的 Jensen 等号当且仅当相应 pθ 后验与 q 后验相同才成立。证毕。（底本该句的零值表达按原公式保留，等号条件与上文一致。）

> $\ln(\cdot)$ $\mathbb{E}_{\mathbf{x}_t{}^{k_t+1:K} \sim q(\mathbf{x}_t{}^{k_t+1:K} \mid \mathbf{x}_t{}^{k_t})}\left[\ln p_{\boldsymbol{\theta}}(\mathbf{x}_t^{k_t} \mid \mathbf{z}_{t-1})\right] = 0$ $p_{\boldsymbol{\theta}}(\mathbf{x}_t^{k_t+1:K} \mid \mathbf{z}_{t-1}, \mathbf{x}_t^{k_t}) = q(\mathbf{x}_t^{k_t+1:K} \mid \mathbf{x}_t^{k_t})$

命题 3 (计算期望ELBO).

$$\mathbb{E}_{\mathbf{x}_t{}^{k_t:K} \sim q\left(\mathbf{x}_t{}^{k_t:K} \mid \mathbf{x}_t{}^0\right)} \ln \frac{p_{\boldsymbol{\theta}}\left(\mathbf{x}_t{}^{k_t:K} \mid \mathbf{z}_{t-1}\right)}{q\left(\mathbf{x}_t{}^{k_t+1:K} \mid \mathbf{x}^0{}_t\right)}$$

$$= C_3\left(\mathbf{x}_0, k_t\right) + \mathbf{1}\{k_t = 0\} \ln p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^0 \mid \mathbf{x}_t^1, \mathbf{z}_{t-1}\right) + \sum_{j=1}^{K-1} \mathbf{1}\{j \geq k_t\} D_{\mathbb{KL}}\left(q\left(\mathbf{x}_t^j \mid \mathbf{x}_t^{j+1}, \mathbf{x}_t^0\right) \| p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^j \mid \mathbf{x}_t^{j+1}, \mathbf{z}_{t-1}\right)\right),$$

其中 C_2 是另一个只依赖干净数据和 k_t 的常数。

> $C_2(\mathbf{x}_0, k_t)$ $\mathbf{x}_0$ k_t

证明.

证明使用标准扩散 ELBO 推导的类似变换，但要仔细处理正向过程只加噪至 k_t 这一点。照常，先用以下恒等式。

> k_t

$$q\left(\mathbf{x}_t^j \mid \mathbf{x}_t^{j-1}, \mathbf{x}_t^0\right) = q\left(\mathbf{x}_t^{j-1} \mid \mathbf{x}_t^j, \mathbf{x}_t^0\right) \cdot \frac{q\left(\mathbf{x}_t^j \mid \mathbf{x}_t^0\right)}{q\left(\mathbf{x}_t^{j-1} \mid \mathbf{x}_t^0\right)}. \quad \text{(A.10)}$$

第1部分：展开似然比

利用上述恒等式，可得下式。

$$\ln \frac{p_{\boldsymbol{\theta}}\left(\mathbf{x}_t{}^{k_t:K} \mid \mathbf{z}_{t-1}\right)}{q\left(\mathbf{x}_t{}^{k_t+1:K} \mid \mathbf{x}^0{}_t\right)}$$

$$= \ln p\left(\mathbf{x}_t^K \mid \mathbf{z}_{t-1}\right) + \ln \frac{p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^{k_t} \mid \mathbf{x}_t^{k_t+1}, \mathbf{z}_{t-1}\right)}{q\left(\mathbf{x}_t^{k_t+1} \mid \mathbf{x}_t^0\right)} + \sum_{j=k_t+2}^{K} \ln \frac{p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^{j-1} \mid \mathbf{x}_t^j, \mathbf{z}_{t-1}\right)}{q\left(\mathbf{x}_t^j \mid \mathbf{x}_t^{j-1}, \mathbf{x}_t^0\right)}$$

$$\overset{(i)}{=} \ln p\left(\mathbf{x}_t^K \mid \mathbf{z}_{t-1}\right) + \ln \frac{p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^{k_t} \mid \mathbf{x}_t^{k_t+1}, \mathbf{z}_{t-1}\right)}{q\left(\mathbf{x}_t^{k_t+1} \mid \mathbf{x}_t^0\right)} + \sum_{j=k_t+2}^{K} \left(\ln \frac{p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^{j-1} \mid \mathbf{x}_t^j, \mathbf{z}_{t-1}\right)}{q\left(\mathbf{x}_t^{j-1} \mid \mathbf{x}_t^j, \mathbf{x}_t^{k_t}\right)} + \ln \frac{q\left(\mathbf{x}_t^{j-1} \mid \mathbf{x}_t^0\right)}{q\left(\mathbf{x}_t^j \mid \mathbf{x}_t^0\right)}\right)$$

$$
\begin{aligned}
&\overset{(ii)}{=} \ln p\left(\mathbf{x}_t^K|\mathbf{z}_{t-1}\right) + \ln \frac{p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^{k_t}|\mathbf{x}_t^{k_t+1},\mathbf{z}_{t-1}\right)}{q\left(\mathbf{x}_t^{k_t+1}|\mathbf{x}_t^0\right)} + \ln \frac{q\left(\mathbf{x}_t^{k_t+1}|\mathbf{x}_t^{k_t}\right)}{q\left(\mathbf{x}_t^K|\mathbf{x}_t^{k_t}\right)} + \sum_{j=k_t+1}^{K-1} \ln \frac{p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^j|\mathbf{x}_t^{j+1},\mathbf{z}_{t-1}\right)}{q\left(\mathbf{x}_t^j|\mathbf{x}_t^{j+1},\mathbf{x}_t^0\right)} \\
&= \frac{\ln p\left(\mathbf{x}_t^K|\mathbf{z}_{t-1}\right)}{q\left(\mathbf{x}_t^K|\mathbf{x}_t^{k_t}\right)} + \ln p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^{k_t}|\mathbf{x}_t^{k_t+1},\mathbf{z}_{t-1}\right) + \sum_{j=k_t+1}^{K-1} \ln \frac{p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^j|\mathbf{x}_t^{j+1},\mathbf{z}_{t-1}\right)}{q\left(\mathbf{x}_t^j|\mathbf{x}_t^{j+1},\mathbf{x}_t^0\right)} \\
&= \ln\left(q\left(\mathbf{x}_t{}^{k_t}|\,\mathbf{x}_t{}^{k_t+1}\right)^{\mathbf{1}\{k_t\geq 1\}}\right) + \ln \frac{p\left(\mathbf{x}_t^K|\mathbf{z}_{t-1}\right)}{q\left(\mathbf{x}_t^K|\mathbf{x}_t^{k_t}\right)} + \ln \frac{p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^{k_t}|\mathbf{x}_t^{k_t+1},\mathbf{z}_{t-1}\right)}{q\left(\mathbf{x}_t{}^{k_t}|\,\mathbf{x}_t{}^{k_t+1}\right)^{\mathbf{1}\{k_t\geq 1\}}} + \sum_{j=k_t+1}^{K-1} \ln \frac{p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^j|\mathbf{x}_t^{j+1},\mathbf{z}_{t-1}\right)}{q\left(\mathbf{x}_t^j|\mathbf{x}_t^{j+1},\mathbf{x}_t^0\right)},
\end{aligned}
$$

其中 (i) 使用式(A.10)，(ii) 用望远镜求和的抵消，最后一式来自下列计算。

> (i) (ii)

$$
q\left(\mathbf{x}_t^{k_t}|\,\mathbf{x}_t^{k_t+1}\right)^{\mathbf{1}\{k_t\geq 1\}} = \begin{cases} 1 & k_t = 0 \\ q\left(\mathbf{x}_t^{k_t}|\mathbf{x}_t^{k_t+1}\right) & k_t \geq 1 \end{cases}. \tag{A.11}
$$

注意，我们并未参数化最大噪声等级处的 $p(x_t^K|z_{t-1})$，所以所列项都可视为仅依赖对应噪声变量的常数 C'。于是得到下式。

> $p\left(\mathbf{x}_t^K|\mathbf{z}_{t-1}\right)$ $\ln\left(q\left(\mathbf{x}_t{}^{k_t}|\,\mathbf{x}_t{}^{k_t+1}\right)^{\mathbf{1}\{k_t\geq 1\}}\right) + \frac{\ln p\left(\mathbf{x}_t^K|\mathbf{z}_{t-1}\right)}{q\left(\mathbf{x}_t^K|\mathbf{x}_t^{k_t}\right)}$ $C'\left(\mathbf{x}_t^{k_t},\mathbf{x}_t^{k_t+1},\mathbf{x}_t^K\right)$

$$
\ln \frac{p_{\boldsymbol{\theta}}\left(\mathbf{x}_t{}^{k_t:K}|\,\mathbf{z}_{t-1}\right)}{q\left(\mathbf{x}_t{}^{k_t+1:K}|\,\mathbf{x}0_t\right)} = C'\left(\mathbf{x}_t^{k_t},\mathbf{x}_t^{k_t+1},\mathbf{x}_t^K\right) + \ln \frac{p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^{k_t}|\mathbf{x}_t^{k_t+1},\mathbf{z}_{t-1}\right)}{q\left(\mathbf{x}_t{}^{k_t}|\,\mathbf{x}_t{}^{k_t+1}\right)^{\mathbf{1}\{k_t\geq 1\}}} + \sum_{j=k_t+1}^{K-1} \ln \frac{p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^j|\mathbf{x}_t^{j+1},\mathbf{z}_{t-1}\right)}{q\left(\mathbf{x}_t^j|\mathbf{x}_t^{j+1},\mathbf{x}_t^0\right)} \tag{A.12}
$$

第2部分：取期望

现在可通过取期望简化。注意下列关系。

$$
\mathbb{E}_{\mathbf{x}_t{}^{k_t:K}\sim q\left(\mathbf{x}_t{}^{k_t:K}|\mathbf{x}_t{}^0\right)} \ln \frac{p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^j|\mathbf{x}_t^{j+1},\mathbf{z}_{t-1}\right)}{q\left(\mathbf{x}_t^j|\mathbf{x}_t^{j+1},\mathbf{x}_t^0\right)} = D_{\mathbb{KL}}\left(q\left(\mathbf{x}_t^j|\,\mathbf{x}_t^{j+1},\mathbf{x}_t^0\right)\|\,p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^j|\,\mathbf{x}_t^{j+1},\mathbf{z}_{t-1}\right)\right),
$$

类似地，有下式。

$$
\mathbb{E}_{\mathbf{x}_t{}^{k_t:K}\sim q\left(\mathbf{x}_t{}^{k_t:K}|\mathbf{x}_t{}^0\right)} \ln \frac{p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^{k_t}|\mathbf{x}_t^{k_t+1},\mathbf{z}_{t-1}\right)}{q\left(\mathbf{x}_t{}^{k_t}|\,\mathbf{x}_t{}^{k_t+1}\right)^{\mathbf{1}\{k_t\geq 1\}}} = \begin{cases} \ln p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^0|\mathbf{x}_t^1,\mathbf{z}_{t-1}\right) & k_t = 0 \\ D_{\mathbb{KL}}\left(q\left(\mathbf{x}_t{}^{k_t}|\,\mathbf{x}_t{}^{k_t+1},\mathbf{x}_t{}^0\right)\|\,p_{\boldsymbol{\theta}}\left(\mathbf{x}_t{}^{k_t}|\,\mathbf{x}_t{}^{j+1},\mathbf{z}_{t-1}\right)\right) & k_t \geq 1. \end{cases}
$$

最后，C' 对正向过程取期望后，是仅依赖 k_t 和干净数据的常数 C_2。因此，由式(A.12)得到以下结果。

> $\mathbb{E}_{\mathbf{x}_t{}^{k_t:K}\sim q\left(\mathbf{x}_t{}^{k_t:K}|\mathbf{x}_t{}^0\right)}$ $C'\left(\mathbf{x}_t^{k_t},\mathbf{x}_t^{k_t+1},\mathbf{x}_t^K\right)$ $C_2\left(k_t,\mathbf{x}_0\right)$ $k_t,\mathbf{x}_0$

$$
\begin{aligned}
&\mathbb{E}_{\mathbf{x}_t{}^{k_t:K}\sim q\left(\mathbf{x}_t{}^{k_t:K}|\mathbf{x}_t{}^0\right)} \ln \frac{p_{\boldsymbol{\theta}}\left(\mathbf{x}_t{}^{k_t:K}|\,\mathbf{z}_{t-1}\right)}{q\left(\mathbf{x}_t{}^{k_t+1:K}|\,\mathbf{x}^0{}_t\right)} \\
&= C_2\left(k_t,\mathbf{x}_0\right) + \mathbf{1}\{k_t=0\}\ln p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^0|\,\mathbf{x}_t^1,\mathbf{z}_{t-1}\right) + \sum_{j=\max\{1,k_t\}}^{K-1} D_{\mathbb{KL}}\left(q\left(\mathbf{x}_t^j|\,\mathbf{x}_t^{j+1},\mathbf{x}_t^0\right)\|\,p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^j|\,\mathbf{x}_t^{j+1},\mathbf{z}_{t-1}\right)\right) \\
&= C_2\left(k_t,\mathbf{x}_0\right) + \mathbf{1}\{k_t=0\}\ln p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^0|\,\mathbf{x}_t^1,\mathbf{z}_{t-1}\right) + \sum_{j=1}^{K-1} \mathbf{1}\{j\geq k_t\} D_{\mathbb{KL}}\left(q\left(\mathbf{x}_t^j|\,\mathbf{x}_t^{j+1},\mathbf{x}_t^0\right)\|\,p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^j|\,\mathbf{x}_t^{j+1},\mathbf{z}_{t-1}\right)\right).
\end{aligned}
$$

■

完成 ELBO 证明。

现在合并前两个命题，得到下式。

$$
\mathbb{E}_{\mathbf{x}_t^{k_t}\sim q\left(\mathbf{x}_t^{k_t}|\mathbf{x}_t^0\right)} \left[\ln p_{\boldsymbol{\theta}}\left(\mathbf{x}_t^{k_t}|\mathbf{z}_{t-1}\right)\right]
$$

$$\geq C_3(\mathbf{x}_0, k_t) + \mathbf{1}\{k_t = 0\} \ln p_{\boldsymbol{\theta}}(\mathbf{x}_t^0 | \mathbf{x}_t^1, \mathbf{z}_{t-1}) + \sum_{j=1}^{K-1} \mathbf{1}\{j \geq k_t\} D_{\mathbb{KL}}\left(q(\mathbf{x}_t^j | \mathbf{x}_t^{j+1}, \mathbf{x}_t^0) \| p_{\boldsymbol{\theta}}(\mathbf{x}_t^j | \mathbf{x}_t^{j+1}, \mathbf{z}_{t-1}) \right),$$

其中 $C_3=C_1+C_2$；同样，当所列 q 与 pθ 条件分布一致时取等号。再对 {0,…,K} 上均匀的 k_t 取期望，得到下式。

$$C_3(\mathbf{x}_0, k_t) = C_1(\mathbf{x}_0, k_t) + C_2(\mathbf{x}_0, k_t) \qquad q(\mathbf{x}_t^{k_t+1:T} | \mathbf{x}_t^{k_t}) \equiv p_{\boldsymbol{\theta}}(\mathbf{x}_t^{k_t+1:T} | \mathbf{x}_t^{k_t}, \mathbf{z}_{t-1}) \qquad k_t \overset{\text{unif}}{\sim} \{0, 1, \ldots, K\}$$

$$\mathop{\mathbb{E}}_{k_t \overset{\text{unif}}{\sim} \{0,1,\ldots,K\}} \left[\mathbf{1}\{k_t = 0\} \right] = \frac{1}{K+1}, \qquad \mathop{\mathbb{E}}_{k_t \overset{\text{unif}}{\sim} \{0,1,\ldots,K\}} \mathbf{1}\{j \geq k_t\} = \frac{j+1}{K+1}. \tag{A.13}$$

因此得到以下结果。

$$\begin{aligned}
&\mathop{\mathbb{E}}_{k_t \overset{\text{unif}}{\sim} \{0,1,\ldots,K\}} \mathop{\mathbb{E}}_{\mathbf{x}_t^{k_t} \sim q(\mathbf{x}_t^{k_t} | \mathbf{x}_t^0), 1 \leq t \leq T} \ln p_{\boldsymbol{\theta}}\left((\mathbf{x}_t^{k_t})_{1 \leq t \leq T} \right) \\
&\geq C_4(\mathbf{x}_t^0) + \frac{1}{K+1} \ln p_{\boldsymbol{\theta}}(\mathbf{x}_t^0 | \mathbf{x}_t^1, \mathbf{z}_{t-1}) + \sum_{j=1}^{K-1} \frac{j+1}{K+1} D_{\mathbb{KL}}\left(q(\mathbf{x}_t^j | \mathbf{x}_t^{j+1}, \mathbf{x}_t^0) \| p_{\boldsymbol{\theta}}(\mathbf{x}_t^j | \mathbf{x}_t^{j+1}, \mathbf{z}_{t-1}) \right)
\end{aligned}$$

应用命题1，可得下式。

$$\begin{aligned}
&\mathop{\mathbb{E}}_{\text{forward}} \left[\ln p_{\boldsymbol{\theta}}\left((\mathbf{x}_t^{k_t})_{1 \leq t \leq T} \right) \right] \\
&\geq \sum_{t=1}^{T} \mathop{\mathbb{E}}_{<t} \mathop{\mathbb{E}}_{k_t \overset{\text{unif}}{\sim} \{0,1,\ldots,K\}} \mathop{\mathbb{E}}_{\mathbf{x}_t^{k_t} \sim q(\mathbf{x}_t^{k_t} | \mathbf{x}_t^0)} \left[\ln p_{\boldsymbol{\theta}}(\mathbf{x}_t^{k_t} | \mathbf{z}_{t-1}) \right] \\
&= \sum_{t=1}^{T} \mathop{\mathbb{E}}_{<t} \left[C_4(\mathbf{x}_t^0) + \frac{1}{K+1} \ln p_{\boldsymbol{\theta}}(\mathbf{x}_t^0 | \mathbf{x}_t^1, \mathbf{z}_{t-1}) + \sum_{j=1}^{K-1} \frac{j+1}{K+1} D_{\mathbb{KL}}\left(q(\mathbf{x}_t^j | \mathbf{x}_t^{j+1}, \mathbf{x}_t^0) \| p_{\boldsymbol{\theta}}(\mathbf{x}_t^j | \mathbf{x}_t^{j+1}, \mathbf{z}_{t-1}) \right) \right]
\end{aligned}$$

最后注意，各时刻 C_4 的过去期望之和是仅依赖整条干净序列的常数 C；并且有以下关系。

$$\sum_{t=1}^{T} \mathop{\mathbb{E}}_{<t} \left[C_4(\mathbf{x}_t^0) \right] \qquad C(\mathbf{x}_{1:T}^0)$$

$$\begin{aligned}
&\mathop{\mathbb{E}}_{<t} \left[\ln p_{\boldsymbol{\theta}}(\mathbf{x}_t^0 | \mathbf{x}_t^1, \mathbf{z}_{t-1}) \right] = \mathop{\mathbb{E}}_{\text{forward}} \mathop{\mathbb{E}}_{p, \mathbf{z}_{1:T}} \left[\ln p_{\boldsymbol{\theta}}(\mathbf{x}_t^0 | \mathbf{x}_t^1, \mathbf{z}_{t-1}) \right] \\
&\mathop{\mathbb{E}}_{<t} \left[D_{\mathbb{KL}}\left(q(\mathbf{x}_t^j | \mathbf{x}_t^{j+1}, \mathbf{x}_t^0) \| p_{\boldsymbol{\theta}}(\mathbf{x}_t^j | \mathbf{x}_t^{j+1}, \mathbf{z}_{t-1}) \right) \right] \\
&= \mathop{\mathbb{E}}_{\text{forward}} \mathop{\mathbb{E}}_{p, \mathbf{z}_{1:T}} \left[D_{\mathbb{KL}}\left(q(\mathbf{x}_t^j | \mathbf{x}_t^{j+1}, \mathbf{x}_t^0) \| p_{\boldsymbol{\theta}}(\mathbf{x}_t^j | \mathbf{x}_t^{j+1}, \mathbf{z}_{t-1}) \right) \right],
\end{aligned}$$

原因是两项都只依赖过去噪声等级、过去噪声 token 与过去潜变量。因此得到下列结论。

$$k_{1:t-1}, (\mathbf{x}_s^{k_s})_{1 \leq s \leq t-1} \qquad \mathbf{z}_{1:t-1}$$

$$\begin{aligned}
&\mathop{\mathbb{E}}_{\text{forward}} \left[\ln p_{\boldsymbol{\theta}}\left((\mathbf{x}_t^{k_t})_{1 \leq t \leq T} \right) \right] \geq C(\mathbf{x}_{1:T}^0) \\
&+ \mathop{\mathbb{E}}_{\text{forward}} \mathop{\mathbb{E}}_{p, \mathbf{z}_{1:T}} \left[\sum_{t=1}^{T} \left(\frac{1}{K+1} \ln p_{\boldsymbol{\theta}}(\mathbf{x}_t^0 | \mathbf{x}_t^1, \mathbf{z}_{t-1}) + \sum_{j=1}^{K-1} \frac{j+1}{K+1} D_{\mathbb{KL}}\left(q(\mathbf{x}_t^j | \mathbf{x}_t^{j+1}, \mathbf{x}_t^0) \| p_{\boldsymbol{\theta}}(\mathbf{x}_t^j | \mathbf{x}_t^{j+1}, \mathbf{z}_{t-1}) \right) \right) \right],
\end{aligned}$$

这就是所需结论。回顾等号条件：（a）潜变量条件分布为狄拉克分布；（b）所列 q 和 pθ 条件分布完全一致。最后重新编号 j←j+1，以与标准扩散 ELBO 的下标一致。

$$p_{\boldsymbol{\theta}}(\mathbf{z}_t | \mathbf{z}_{t-1}, \mathbf{x}_t^{k_t}) \qquad q(\mathbf{x}_t^{k_t+1:T} | \mathbf{x}_t^{k_t}) \equiv p_{\boldsymbol{\theta}}(\mathbf{x}_t^{k_t+1:T} | \mathbf{x}_t^{k_t}, \mathbf{z}_{t-1}) \qquad j \leftarrow j+1$$

■

附录B 更多直觉与解释

B.1 扩展至 Transformer 主干

虽然本文关注基于 RNN 的因果实现，但 DF 很容易适配现代 Transformer。只需让基于 Transformer 的序列扩散训练采用 token 独立噪声等级，并遵循D.1技巧。严格的因果 DF 要用因果注意力掩码。不过，部分掩码还有更有趣的用法：即使用没有因果掩码的 Transformer，也能通过控制噪声实现因果性。把未来标成白噪声，就不会向过去泄漏信息；把未来标成无噪声，模型就完全非因果；把未来标成部分噪声，则允许少量未来信息用于预测过去。因此，仅用非因果架构并控制未来噪声程度，就可能实现部分或完全因果性。这些扩展超出本文范围，但我们已验证有效，故作为后续工作的直觉提供。

B.2 为什么噪声等级必须独立

训练 DF 时，我们令逐 token 噪声等级从 [1,…,K] 独立同分布均匀采样。这个选择是否必要？下面讨论独立噪声的独有能力和新增计算开销。

> $[1, 2 \ldots K]$

独立噪声带来：3.3节的自回归滚动稳定化、因果不确定性建模，以及B.6中以历史为条件时避免昂贵重建引导。全序列扩散无法实现这些能力；AR-diffusion [67] 和 Rolling Diffusion [52] 只支持第一和第三项。还存在灵活插帧等采样期应用。独立噪声也便于调参：只需尝试不同采样日程，找出适合应用者。所有能力只需一次 DF 训练；AR-diffusion 和 Rolling Diffusion 若调采样方案，必须重训。

另一方面，只要严格遵循D.1等训练技巧，我们没有观察到相对全序列扩散明显的额外计算开销。这一经验来自实验性 Transformer 版 DF，并非完全与正文实验一致。高层过程是：独立噪声新增的复杂性在时间维；先用视频扩散常见的图像预训练，把像素本身的复杂性抽离，剩下时间预测；再在视频上继续训练预训练图像模型。第二阶段较少训练步的 DF 已优于收敛后的全序列扩散。我们推测这源于 [40] 所述的数据增强效应。考虑包括图像预训练的总计算，独立噪声增加的开销是值得的。

B.3 以引导实现规划

如第2节，可用分类器对数概率的梯度引导采样至具有属性 y 的结果，例如成功事件。但也可考虑更一般的能量函数 c，其指数可解释为成功的条件概率。一些常用能量候选如下。

> $\log c\left(y \mid \mathbf{x}_t^k\right) \quad y \quad y \quad c\left(\mathbf{x}_t^k\right) \quad \Pr\left(y \mid \mathbf{x}_t^k\right) \quad \Pr[y=1 \mid \mathbf{x}_t^k]=e^{c\left(\mathbf{x}_t^k\right)}$

$$c\left(\mathbf{x}_t^k\right)=\mathbb{E}\left[\sum_{t^{\prime}>t} \mathbf{r}_{\prime}\left(\mathbf{x}_{t^{\prime}}^{k_{t^{\prime}}}\right) \mid \mathbf{x}_t^k\right], \tag{B.1}$$

它们对应剩余代价；可用未来累积奖励构造该梯度的无偏估计。也可用末端位置至目标的负平方距离作为终端奖励。maze2d 实验实际引导函数见D.5。

> $\widetilde{c}\left(\mathbf{x}_t^k\right)= \quad \sum_{t^{\prime}>t} \mathbf{r}_{\prime}\left(\mathbf{x}_{t^{\prime}}^{k_{t^{\prime}}}\right) \quad c=-\left\|\mathbf{x}_T^{k_T}-\mathbf{g}\right\|^2$

B.4 加噪如何稳定长时域生成

每个时刻 t 去噪时，都保留上一时刻小正噪声等级 k_small 对应的潜状态 z_{t-1}。以它为初始潜变量做下一 token 扩散，把当前 token 从 K 逐级去噪至0（即算法2取 T=1，日程 $\mathcal{K}_{m,1}$=m）。这也得到各噪声等级对应的潜变量。选择当前 k_small 的潜变量，重复过程。此带噪声潜状态可解释为自回归过程以略带噪声的当前 token 为条件。若采用B.1的 Transformer，可对已完全去噪 token 再做正向加噪以稳定生成。

> t $\mathbf{z}_{t-1}^{k_{\text{small}}}$ $0 < k_{\text{small}} \ll K$ $\mathbf{x}_t$ $\mathbf{x}_t^K, \mathbf{x}_t^{K-1}, ..., \mathbf{x}_t^0$ $T = 1$ $\mathbf{z}_{t-1}^k$ $\mathcal{K}_{m,1} = m$ $\mathbf{z}_t^K, \mathbf{z}_t^{K-1}, ..., \mathbf{z}_t^0$ $\mathbf{z}_t^{k_{\text{small}}}$ $\mathbf{x}_t^{k_{\text{small}}}$

数据加噪能缓解行为克隆中的长期误差累积 [39,43]，也能增强对非序列对抗攻击的鲁棒性 [13]。自回归视频中，略带噪声的前一帧属于训练分布，因为 DF 的目标本来就以带噪声过去观测训练。它可视为行为克隆 DART [43] 的特例：模仿者（这里为视频生成器）从噪声观测（带噪声前帧）预测动作（下一帧）。更准确地，DF 训练覆盖这两种 token，并在测试自回归时使用略带噪声 token，因此继承 HINT 算法 [5] 的理论保证。

> $\mathbf{x}_t^{k_{\text{small}}}$

B.5 为什么蒙特卡洛引导依赖 DF

MCG 大幅降低剩余代价引导估计的方差。它关键依赖：能从当前 token 滚动采样未来，用多条未来样本估计梯度。全序列扩散必须同步去噪全部 token，固定噪声等级下没有明显可用于蒙特卡洛估计的随机源。也许可以用下一节技巧模拟未来并得到可变时域，但据我们所知，这不是标准做法。

B.6 替换技巧能让全序列扩散具有灵活时域吗?

朴素办法是“替换技巧”：将固定长度序列分成已生成的过去和待生成的未来。在后续去噪中，固定过去部分为先前生成 token，仅用分数梯度更新未来。明显缺点是低效：即使只剩一步，仍要扩散整条序列。此外，[32] 指出这种“通过替换条件化”的方法缺乏数学原则，会使生成序列不一致。[32] 最好的修复是在每一步增加对未来 token 的额外梯度项；但修复仍不完整，并使每次采样多付出一次反向传播开销。

> $\mathbf{x}_{1:T}$ $\mathbf{x}_{1:t-1}, \mathbf{x}_{t:T}]$ $\mathbf{x}_{1:t-1}$ $[\tilde{\mathbf{x}}_{1:t-1}^k, \mathbf{x}_{t:T}^k]$ $\tilde{\mathbf{x}}_{1:t-1}{}^k = \mathbf{x}_{1:t-1}$ $\mathbf{x}_{t:T}{}^k$ $t = T - 1$ $\mathbf{x}_{t:T}$

B.7 与贝叶斯滤波的进一步联系

DF 的核心可理解为用扩散在贝叶斯滤波的先验与后验之间插值。考虑混合条件分布 $p(z_t|z_{t-1},x_t^k)$：k=0 时成为有真实观测的后验；k=K 时输入为独立高斯噪声，不包含潜状态信息，因此恰为先验 $p(z_t|z_{t-1})$。令 k 在 K 与0之间变化，同一网络就能参数化先验与后验之间的全部分布。

> $p(\mathbf{z}_t|\mathbf{z}_{t-1}, \mathbf{x}_t^k)$ $k = 0$ $p(\mathbf{z}_t|\mathbf{z}_{t-1}, \mathbf{x}_t)$ $k = K$ $p(\mathbf{z}_t|\mathbf{z}_{t-1}, \mathbf{n})$ $\mathbf{n} \sim \mathcal{N}(0, \mathbf{I})$ $\mathbf{n}$ $\mathbf{z}$ $p(\mathbf{z}_t|\mathbf{z}_{t-1})$ k K 0

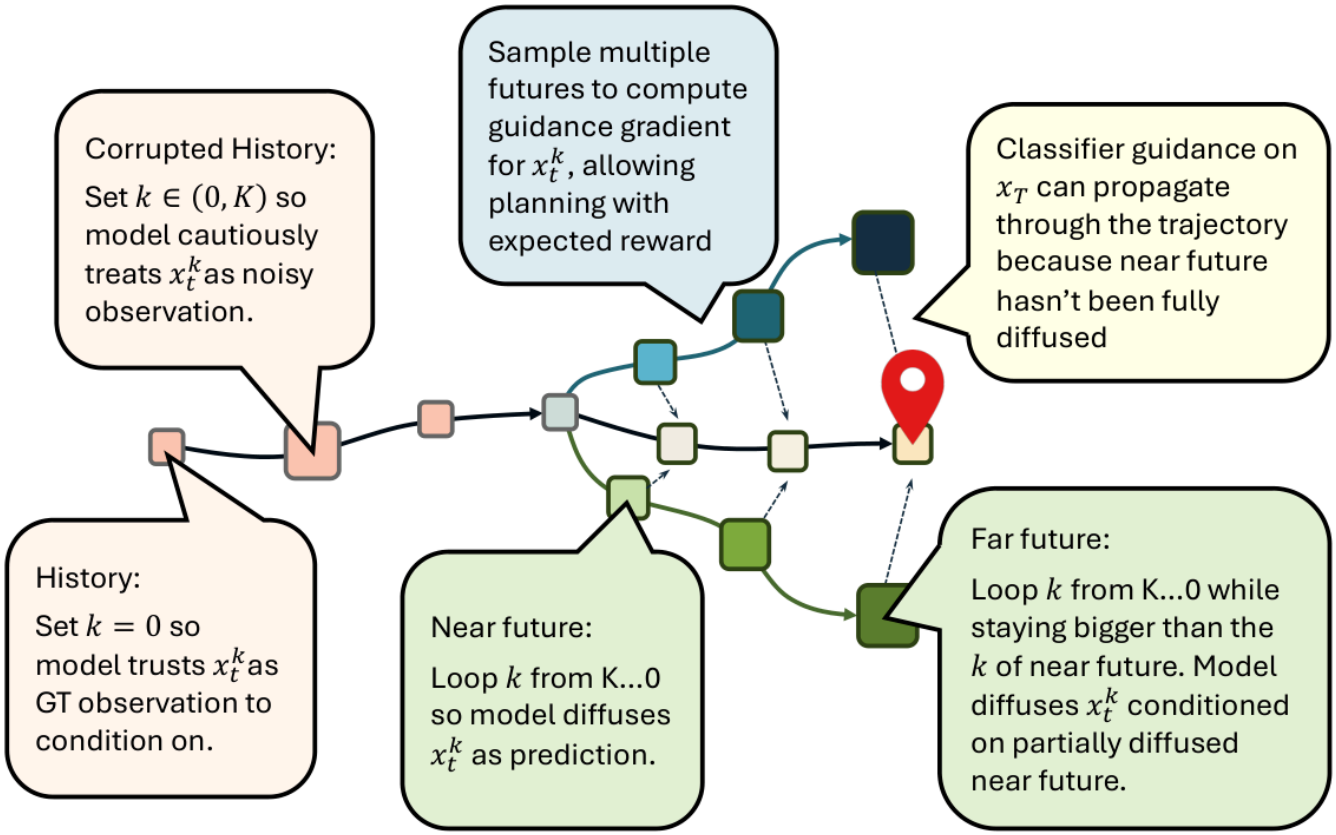


图5：DF 在不同时间步以独立噪声等级训练。因此，可控制噪声等级 k，使条件化和预测产生不同效果。

> k

B.8 与其他序列训练方案的联系

把噪声视为掩码，可以统一各类序列训练方案。以长度3为例：总从三个完全掩码白噪声 token 出发，目标是得到三个零噪声干净 token；假设扩散均用3步 DDIM 采样。

> 3 $[\mathbf{x}_1^K, \mathbf{x}_2^K, \mathbf{x}_3^K]$ $[\mathbf{x}_1^0, \mathbf{x}_2^0, \mathbf{x}_3^0]$ 3

自回归。

教师强制根据过去真实观测训练下一 token 预测。可像 [50] 一样训练下一 token 扩散：输入历史与当前噪声观测，要求预测干净当前观测。例如输入为前两个干净 token 加第三个白噪声 token，目标为三个干净 token。

> $[\mathbf{x}_1^0, \mathbf{x}_2^0, \mathbf{x}_3^K]^\top$ $[\mathbf{x}_1^0, \mathbf{x}_2^0, \mathbf{x}_3^0]^\top$

采样时先完全去噪下一个 token，再将它纳入历史，执行自回归滚动；过程如下。

$$
\begin{aligned}
&[\mathbf{x}_1^K, \mathbf{x}_2^K, \mathbf{x}_3^K]^\top \\
&[\mathbf{x}_1^{K//2}, \mathbf{x}_2^K, \mathbf{x}_3^K]^\top, \\
&[\mathbf{x}_1^0, \mathbf{x}_2^K, \mathbf{x}_3^K]^\top, \\
&[\mathbf{x}_1^0, \mathbf{x}_2^{K//2}, \mathbf{x}_3^K]^\top \\
&[\mathbf{x}_1^0, \mathbf{x}_2^0, \mathbf{x}_3^K]^\top,
\end{aligned}
$$

$$[\mathbf{x}_1^0, \mathbf{x}_2^0, \mathbf{x}_3^{K//2}]^\top,$$
$$[\mathbf{x}_1^0, \mathbf{x}_2^0, \mathbf{x}_3^0]^\top.$$

DF 也支持此采样方案，适合模仿学习等需要尽快生成下一动作的应用。

全序列扩散。

全序列扩散接收整个噪声序列，逐噪声等级同步去噪，如下所示。

$$[\mathbf{x}_1^K, \mathbf{x}_2^K, \mathbf{x}_3^K]^\top$$
$$[\mathbf{x}_1^{K//2}, \mathbf{x}_2^{K//2}, \mathbf{x}_3^{K//2}]^\top,$$
$$[\mathbf{x}_1^0, \mathbf{x}_2^0, \mathbf{x}_3^0]^\top.$$

DF 同样能在采样时采用此方案。

带因果不确定性的 DF

如图2，为建模因果不确定性，DF 在扩散的任一时刻都使远未来噪声等级更大，从而比近未来更不确定。示例如下。

> k

$$[\mathbf{x}_1^K, \mathbf{x}_2^K, \mathbf{x}_3^K]^\top$$
$$[\mathbf{x}_1^{K//2}, \mathbf{x}_2^K, \mathbf{x}_3^K]^\top,$$
$$[\mathbf{x}_1^0, \mathbf{x}_2^{K//2}, \mathbf{x}_3^K]^\top,$$
$$[\mathbf{x}_1^0, \mathbf{x}_2^0, \mathbf{x}_3^{K//2}]^\top$$
$$[\mathbf{x}_1^0, \mathbf{x}_2^0, \mathbf{x}_3^0]^\top$$

值得指出，[66] 首先为因果扩散模型提出这种线性不确定性采样；DF 则将它与其他能力结合，给出了更一般的方案。

带稳定化的 DF

DF 也能做前述自回归采样，但把预测当作真实观测会累积单步误差。DF 通过告诉模型“生成图像应视为有噪声的真值”来缓解（图2）。

> $\mathbf{x}$

先完全去噪第一个 token，如下所示。

$$[\mathbf{x}_1^K, \mathbf{x}_2^K, \mathbf{x}_3^K]^\top$$
$$[\mathbf{x}_1^{K//2}, \mathbf{x}_2^K, \mathbf{x}_3^K]^\top,$$
$$[\mathbf{x}_1^0, \mathbf{x}_2^K, \mathbf{x}_3^K]^\top$$

再把已去噪的第一个 token 输入模型，但告知其处于略高噪声等级1，用来生成第二个 token。

> $\mathbf{x}_1^0 \quad \mathbf{x}_1^1 \quad \mathbf{x}_2$

$$[\mathbf{x}_1^1, \mathbf{x}_2^{K//2}, \mathbf{x}_3^K]^\top$$

$[\mathbf{x}_1^1, \mathbf{x}_2^0, \mathbf{x}_3^K]^\top$

随后输入完全去噪的第二个 token，同样把它标为较高噪声等级1，继续生成。

> $\mathbf{x}_2^0 \quad \mathbf{x}_2^1$

$[\mathbf{x}_1^1, \mathbf{x}_2^1, \mathbf{x}_3^{K//2}]^\top$,
$[\mathbf{x}_1^1, \mathbf{x}_2^1, \mathbf{x}_3^0]^\top$.

附录C 扩展相关工作

重建被掩码 token。

图像 [27] 和视频 [17] 的掩码自编码器，是像素空间表示学习的常用方法；它们还被扩展为以未掩码块为条件，扩散生成被掩码块 [63,19]。

把图像生成转为序列生成。

[61,8] 表明，即使是图像等非序列数据，其生成建模也能有效转化为序列生成。

非扩散的概率序列模型。

[12] 用变分自编码器参数化 token 间转移。这使转移具有概率性，但不直接最大化整个序列的联合概率，而是支持从单步转移分布采样。

具有可变噪声等级的序列扩散。

最相近的是 AR-Diffusion [66]，同样训练下一 token 模型做序列扩散。关键区别是：其噪声等级线性依赖单词位置，而我们令每个等级独立，这使自回归稳定化、损坏观测条件化等采样方案成为可能。AR-Diffusion 只研究语言，没有研究引导；我们把 DF 作为广泛适用的序列模型，特别研究序贯决策，并提出 MCG。另一相关工作 Rolling Diffusion [52] 令近未来较确定、远未来较不确定，类似 DF 的因果不确定性日程。但它与 AR-Diffusion 一样，训练等级线性依赖 token 位置，采样必须沿用完全相同方案，因此也有上述限制。

附录D 更多方法细节

D.1 融合信噪比重新加权

SNR 重新加权 [24] 广泛用于加速图像扩散收敛：简言之，让扩散损失权重正比于噪声样本的信噪比。在 DF 中，除当前噪声 token 外，条件变量 z_{t-1} 也可能包含关于当前干净 token 的大量信息。例如确定性马尔可夫系统中，如果前一观测噪声等级为0，后验潜状态就包含预测当前干净 token 所需的全部信息，而与当前 token 的噪声等级无关。

> $\mathbf{x}^k \quad \mathbf{z}_{t-1} \quad \mathbf{x}_t \quad \mathbf{x}_t^{k_t} \quad \mathbf{x}_{t-1}^{k_{t-1}} \quad k_{t-1}=0 \quad \mathbf{z}_{t-1} \quad \mathbf{x}_t^0 \quad \mathbf{x}_t^{k_t}$

因此，我们重新推导 SNR 权重以反映 DF 的变化，称为融合 SNR 重新加权。沿用原 SNR 的直觉，在逐时刻噪声独立的序列中宽泛定义信噪比。令 S_t 为按普通扩散推导后归一化的当前 SNR 权重；例如 min-SNR [24] 权重在 [0,C]，除以 C 后在 [0,1]。用信号衰减率 $0<\gamma<1$ 表示过去信号对当前去噪的贡献比例，采用简单指数

衰减模型。递归累积信噪比为 $\bar{S}_t=\gamma\bar{S}_{t-1}+(1-\gamma)S_t$，以计入整个噪声历史。另一信息来源是当前噪声观测本身的 S_t。二者都在 [0,1]，可简化解释为“具备完美去噪所需全部信息”的概率。因当前噪声独立于历史噪声，将 S_t 与 $\bar{S}_{t-1}$ 视为独立事件概率，合成为 $S'_t=1-(1-S_t)(1-\bar{S}_{t-1})$，并把它用作扩散训练的融合 SNR 权重。

> S_t $\mathbf{x}_t^{k_t}$ $[0,C]$ C $S_t \in [0,1]$ $0<\gamma<1$ $\mathbf{x}_{t-1}^{k_t-1}$ $\mathbf{x}_t^{k_t}$ S_t $S_t=\gamma S_{t-1}+(1-\gamma)S_t$ t S_t $\mathbf{x}_t^{k_t}$ S_t S_t $[0,1]$ $\mathbf{x}_t^{k_t}$ t S_t S_{t-1} $S_t^{'}=1-(1-S_t)(1-S_{t-1})$ $S_t^{'}$

实验按 min-SNR [24] 得到 S。融合权重对加快视频预测收敛极有用；非图像领域未观察到收益，因而未使用。

> S

D.2 架构

视频扩散

原始图像 x 与潜状态 z 均为含通道、宽和高的二维空间张量，为简便二者宽高相同。转移模型使用典型扩散 U-Net [47]，其输出与上一潜状态一起输入门控循环单元 GRU，输出作为新潜状态。观测模型用一层 ResNet [28] 后接卷积。二者组合为 RNN 层，以前一潜变量和当前噪声图像为输入，输出干净图像估计。因果 Transformer 或许能更好，但 RNN 在在线决策中更高效，所以视频也使用 RNN，结果已令人满意。

> $\mathbf{x}$ $\mathbf{z}$ $\mathbf{x}$ $\mathbf{z}$ $p(\mathbf{x}_t^{k_t}|\mathbf{z}_{t-1})$ $\mathbf{z}_{t-1}$ $\mathbf{z}_t$ $p(\mathbf{x}_t|\mathbf{z}_t)$ 1 $\mathbf{z}_{t-1}$ $\mathbf{x}_t^{k_t}$ $\hat{\mathbf{x}}$

潜状态通道数：DMLab 为16，Minecraft 为32；总参数量分别为2400万和3600万。更多参数可能改善 Minecraft，但为把训练控制在一天以内，留待后续。迷宫规划模型共433万参数。

> $\mathbf{z}$ 16 32 36 24 <1 4.33

非视频扩散

对非视频、非图像、无空间结构的 x，动态模型主干改用残差 MLP [60]，即 MLP 版 ResNet [28]。同视频模型，残差 MLP 输出与前一潜状态送入 GRU 得到新潜状态；另一残差 MLP 充当观测模型。

> $\mathbf{x}$ $\mathbf{z}_{t-1}$ $\mathbf{z}_t$

D.3 扩散参数化

扩散有 x_0、ε [29] 与 v [53] 三种等价预测目标；它们结合 SNR 权重，对不同噪声等级产生不同的损失重新加权。例如 ε 与 v 参数化对强调高频细节的像素生成很重要。

> $\mathbf{x}_0$ ϵ v ϵ v

视频预测使用 v 参数化，它对收敛速度和质量都至关重要。

> v

规划和模仿学习明显更适合 x_0 参数化，可能因为不需要人为强调高频细节。时间序列预测则观察到 v 参数化有益。

> $\mathbf{x}_0$

D.4 噪声日程

视频预测用 sigmoid 噪声日程 [9]，迷宫规划用线性日程，其余用余弦日程。

D.5 带引导采样的实现细节

采样噪声的边界情形

灵活日程矩阵会出现某 token 在一次采样步骤中噪声等级不变的边界情形。此时是否更新 token？一种选择是复制原值；另一种是先逆扩散，再正向扩散回原等级，以重新采样。我们认为这仍是开放问题，但更偏好后者，并用它在 MCG 中生成多个样本。即使采用复制，引导梯度仍可经动态模型传至前一潜状态和更早时间步。

> $\mathcal{K}$ $\mathbf{x}_t^{k_t}$ $\mathbf{x}_t^{k_t}$ t $p(\mathbf{z}_t|\mathbf{x}_t^{k_t}, \mathbf{z}_{t-1})$ $\mathbf{z}_{t-1}$

除 MCG 外，实验中这一情形只出现在 k_t=0 或 K：日程设计为 token 一旦开始去噪，就持续去噪。k_t=K 时保持白噪声，无需重抽；k_t=0 时已完全去噪，两种办法应相同，故为简单起见复制。

> $k_t = 0$ $k_t = K$ $\mathcal{K}$ $k_t = K$ $\mathbf{x}_t^{k_t}$ $k_t = 0$

迷宫规划引导

主要基线 Diffuser [37] 丢弃数据集奖励，直接以目标位置和速度规划；DF 也遵循此约定。可用终端位置与目标距离引导，但灵活时域模型不应要求用户手工指定何时到达，而应在任意可能时域争取到达。因此我们用各时刻目标距离之和作为奖励，使任一时刻都可成为到达终点。二维迷宫的非凸性使该目标困难，但 DF 仍能可靠找到不撞墙计划。也观察到智能体到达后离开目标：数据集只把目标当作可能经过的路点，没有停留示范。我们尝试在 Diffuser 中用同样奖励，即使大量调参也未成功。（底本此段距离及求和公式的符号、下标保持原样；优化方向按“趋近目标”的叙述理解。）

> $\|\mathbf{p}_T - \mathbf{g}\|$ T $\sum_t \|\mathbf{p}_T - \mathbf{g}\|$

D.6 性能优化

加速 DF 采样与普通扩散相似。每个 token 采用 DDIM [58]：训练扩散步数 K=1000，视频仅用100步 DDIM，非视频领域用50步。

> $K = 1000$ 100 50

尽管可用 Transformer，本文使用 RNN，因为它在在线决策中灵活、高效。为节省训练和显存，把多帧堆叠成一个 token：相邻帧可能极相似，为几乎相同帧多次展开动态模型很浪费。视频中，人工检查需要隔多少步才真正需要预测而不是复制帧。另一个原因是：噪声日程等扩散技巧往往面向内部相关、有冗余的数据；低维系统缺乏此冗余时，可能需要完全不同超参数。帧堆叠也让非图像实验可从标准扩散超参出发。DMLab 堆4帧、Minecraft 堆8帧、迷宫规划堆10步。

> $\mathbf{x}$ $\mathbf{x}$ 4 8 10

算法2第8行还有节省计算的选择：直接把新计算潜变量赋给当前潜变量，而不按已经更新的观测和低一级噪声重算后验。因为训练本来就覆盖任意噪声历史，两种方案都有效。选择第8行有两个原因：第一，避免每步重

算后验，将计算减半；第二，它正好实现所需稳定化——简化观测模型下，新潜变量已经包含新干净观测的信息，但估计时采用比新观测更高一级的噪声等级。

> $\mathbf{z}_t^{\text{new}}$ $\mathbf{z}_t$ $\mathbf{z}_t$ $p(\mathbf{z}_t|\mathbf{z}_{t-1}, \mathbf{x}_t^{\text{new}}, k-1)$ $\mathbf{z}_t$ 8 $\mathbf{z}_t^{\text{new}}$ $\mathbf{x}_t^{\text{new}}$ $k = k_t$ $\mathbf{x}_t^{\text{new}}$

D.7 因果不确定性的采样日程

推理见算法2和图2。式(D.1)展示因果规划实际使用的日程矩阵。为简便，假设给定初始潜变量 z_0，目标是生成 $x_{1:h+1}$。

> $\mathcal{K}$ $\mathbf{z}_0$ $\mathbf{x}_{1:H+1}$

$$\mathcal{K}^{\text{pyramid}} = \begin{bmatrix} K & K & K & \dots & K \\ K-1 & K & K & \dots & K \\ K-2 & K-1 & K & \dots & K \\ & & & \ddots & \\ 1 & 2 & 3 & \dots & H \\ 0 & 1 & 2 & \dots & H-1 \\ & & & \ddots & \\ 0 & 0 & 0 & \dots & 1 \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix} \tag{D.1}$$

先以等级 K 的白噪声初始化序列，再按噪声等级递减顺序遍历矩阵各行。每行依次经过各帧，更新潜状态（算法2第5行），再执行部分逆过程去噪至矩阵指定等级（第6—7行）。这种日程称为金字塔日程，因为远未来始终比近未来具有更高噪声。

> K $m = 1, \dots, M$ $\mathcal{K}$ $t = 1, \dots, T$ $k = \mathcal{K}_{m,t}$ $\mathcal{K}$ $\mathcal{K}$

D.8 迷宫规划指标

表1报告各迷宫的回合奖励，但它未必是好指标。直觉上应奖励最快找到目标的聪明智能体，而非慢走到回合末尾才抵达者。数据中没有停留于目标的行为，序列规划方法到达后会离开。Diffuser 生成较慢计划，反而可能在目标附近多停留几步，获得不公平的高奖励。这似乎利用了环境设计缺陷；更合理的设计应惩罚到达耗时。因此建议后续工作使用首次到达目标时间等替代指标，DF 在该方面表现优秀。

D.9 时间序列回归实现细节

遵循 pytorch-ts：验证集是训练集中随机抽取的子序列，数量与测试集相同。验证 CRPS-sum 连续6轮未改善时早停（底本此处写“未增加”，按E.1一致的最小化指标解释）。架构相同，为1个 MLP 和4个 GRU，批量32。

D.10 计算资源

所有实验采用 fp16 混合精度。时间序列、迷宫规划、组合生成和视觉模仿学习均可在11GB显存的单张2080 Ti上训练。批量尽量占满显存：迷宫与组合实验2048，视觉模仿32。时间序列用验证早停；未仔细搜索最少训练步数，一般在5万至10万步收敛。上述环境通常训练4—8小时，仍有很大加速潜力。

> $fp16$ $2080Ti$ 11 2048 32 50 $100k$ $4-8$

视频预测更耗 GPU：两个数据集均使用8张 A100，训练5万步，批量8×16。通常约4万步、12小时收敛，含间歇验证时间。

> $8 \quad 50K \quad 8 \times 16 \quad 12 \quad 40K$

附录E 更多实验结果

E.1 多变量概率时间序列预测

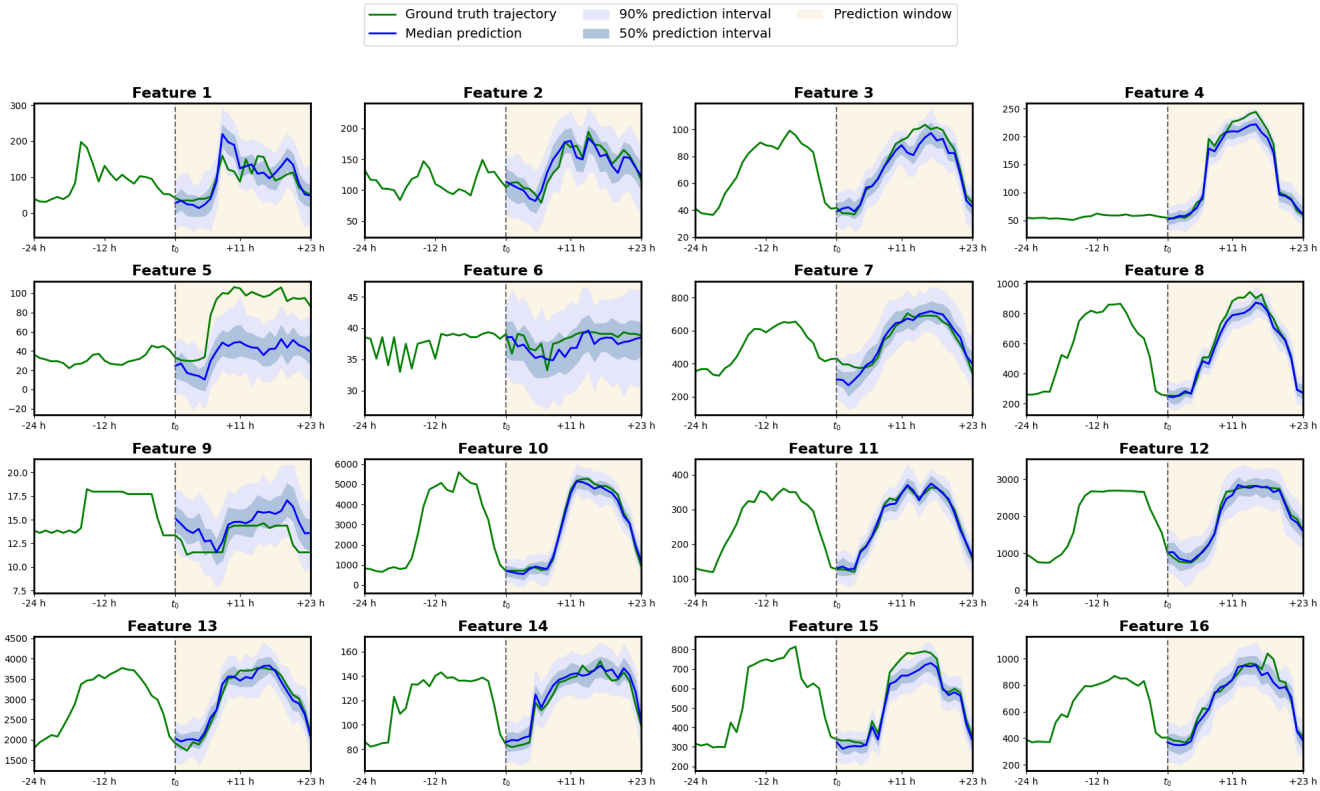


图6：Electricity 测试集第一个预测窗的 DF 预测区间。370个特征中仅绘出前16个。

为说明新目标并不损害 DF 作为通用序列模型的能力，我们评估高维、长时域时间序列预测。采用 GluonTS [2] 的多个真实应用数据集，与强基线按领域标准指标比较。本节集中报告结果和分析；数据与指标细节见 F.4。

问题定义

设 X={xₜ} 为底层动态过程按离散时间采样的 D 维观测序列。概率预测在 t₀ 将其划分为上下文窗 Xc（历史 / 证据，长度 t₀−1）及预测窗 Xp（时域，长度 T−t₀+1）。目标是建模下面预测窗样本的条件联合分布。

> $\boldsymbol{X} = \{\mathbf{x}_t\}_{t=1}^{T} \quad D \quad \mathbf{x}_t \in \mathbb{R}^D \quad t \in \{1, \ldots, T\} \quad T \in \mathbb{N} \quad \boldsymbol{X} = \{\boldsymbol{X}_c, \boldsymbol{X}_p\} \quad t_0 \in \mathbb{N} \quad 1 < t_0 \leq T \quad \boldsymbol{X}_c := \{\mathbf{x}_t\}_{t=1}^{t_0-1} \quad t_0 - 1$
> $\boldsymbol{X}_p := \{\mathbf{x}_t\}_{t=t_0}^{T} \quad T - t_0 + 1$

$$q\left(\mathbf{x}_{t_0:T} | \mathbf{x}_{1:t_0-1}\right) := \prod_{t=t_0}^{T} q\left(\mathbf{x}_t | \mathbf{x}_{1:t-1}\right) \quad \text{(E.1)}$$

若知道式(E.1)的分布，就可根据历史上下文采样预测序列。但自然界时间相关过程动态复杂，通常没有可计算的条件分布形式。我们用统计模型逼近生成过程，通过模拟轨迹的蒙特卡洛采样估计分位数，得到置信水平或不确定性；平均或中位轨迹可作为点预测 [36]。

> $q\left(\mathbf{x}_{t_0:T} \mid \mathbf{x}_{1:t_0-1}\right)$

结果。

方法	Exchange	Solar	Electricity	Traffic	Taxi	Wikipedia
VES [36]	.005±.000	.900±.003	.880±.004	.350±.002	—	—
VAR [45]	.005±.000	.830±.006	.039±.001	.290±.001	—	—
VAR-Lasso [45]	.012±.000	.510±.006	.025±.000	.150±.002	—	3.100±.004
GARCH [62]	.023±.000	.880±.002	.190±.001	.370±.001	—	—
DeepAR [55]	—	.336±.014	.023±.001	.055±.003	—	.127±.042
LSTM-Copula [54]	.007±.000	.319±.011	.064±.008	.103±.006	.326±.007	.241±.033
GP-Copula [54]	.007±.000	.337±.024	.025±.002	.078±.002	.208±.183	.086±.004
KVAE [41]	.014±.002	.340±.025	.051±.019	.100±.005	—	.095±.012
NKF [14]	—	.320±.020	.016±.001	.100±.002	—	.071±.002
Transformer-MAF [51]	.005±.003	.301±.014	.021±.000	.056±.001	.179±.002	.063±.003
TimeGrad [50]	.006±.001	.287±.020	.021±.001	.044±.006	.114±.020	.049±.002
ScoreGrad sub-VP SDE [68]	.006±.001	.256±.015	.019±.001	.041±.004	.101±.004	.043±.002
DF（本文）	.003±.001	.289±.002	.023±.001	.040±.004	.075±.002	.085±.007

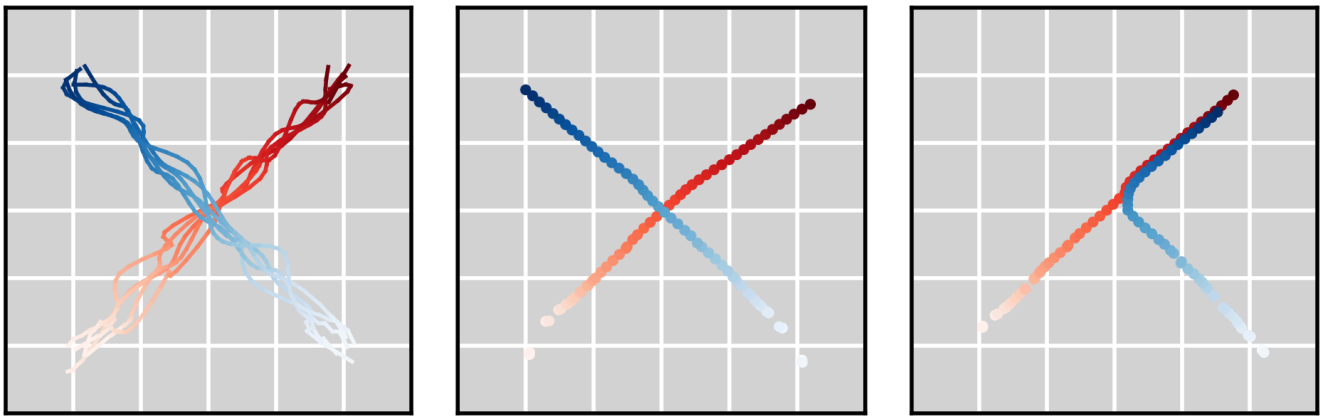
表2：时间序列预测结果。报告六个数据集测试集 CRPS-sum，越低越好；本方法均值和标准差来自5个不同训练种子。表中各方法完整数值以重排后的表格保留。

遵循 [54,51,50,59,68]，在多变量概率时间序列预测这一标准任务中评估 DF，具体包括 Solar、Electricity、Traffic、Taxi、Wikipedia 等数据，维度、领域、采样频率和季节周期均不同。数据来自 GluonTS [2]，每个数据集的上下文窗与预测窗等长，并采用与 [50] 相同协变量。用分位数估计求和连续排序概率分数 CRPS-sum，以衡量预测与真值分布匹配程度，详见F.4。比较包括 TimeGrad [50] 和 Transformer-MAF [51]；关键基线 TimeGrad 是教师强制训练的下一 token 扩散模型。跟踪验证 CRPS-sum，连续6轮未改善则早停；所有数据集每轮固定100批。训练结束后测测试 CRPS-sum。所有时间序列数据和实验采用完全相同架构与超参数。除 ScoreGrad [68] 外，DF 优于此前方法；与其总体相当，但 Wikipedia 上 DF 仅居第四。时间序列并非 DF 核心应用，此处旨在说明该目标可适用于多领域，且相对基线目标未表现出明显性能代价。

E.2 更多组合生成结果

DF 建模序列任意子集的联合分布，因此能从轨迹子集分布采样，再将子轨迹组合为新轨迹。

还可灵活控制组合程度。如图7，二维正方形上的数据轨迹从一角到对角，形成叉形。保留 HMM 完整记忆时，模型重现叉形；需要拼接两个子轨迹形成 V 形时，以无记忆上下文执行短时域 MPC。



(a) 数据集　　(b) 有记忆　　(c) 无记忆

图7：给定轨迹数据集，DF 建模任意长度全部子序列的联合分布。采样时，采用完整时域可从原轨迹分布抽样；忽略过去状态则恢复马尔可夫动态。底本子图引用6(a)—6(c)沿用其内容，图号按图7识别。

E.3 更多视频预测结果（未挑选样本）

不使用滑动窗口的无限滚动

DF 无需滑动窗口即可生成超出最大训练时域的序列：连续运行 RNN，从不重置初始潜变量。这是稳定化带来的令人意外的现象。图8、10生成180帧并稀疏展示；Minecraft 最大训练长度72帧，DMLab 为36帧，因此无滑窗已扩展约2—5倍。我们还尝试2000帧，两个数据集均未见模型崩溃。Minecraft 偶尔卡住，满屏泥土方块，这更像数据集问题；转身后能恢复。

> z_0　180　72　36　2000

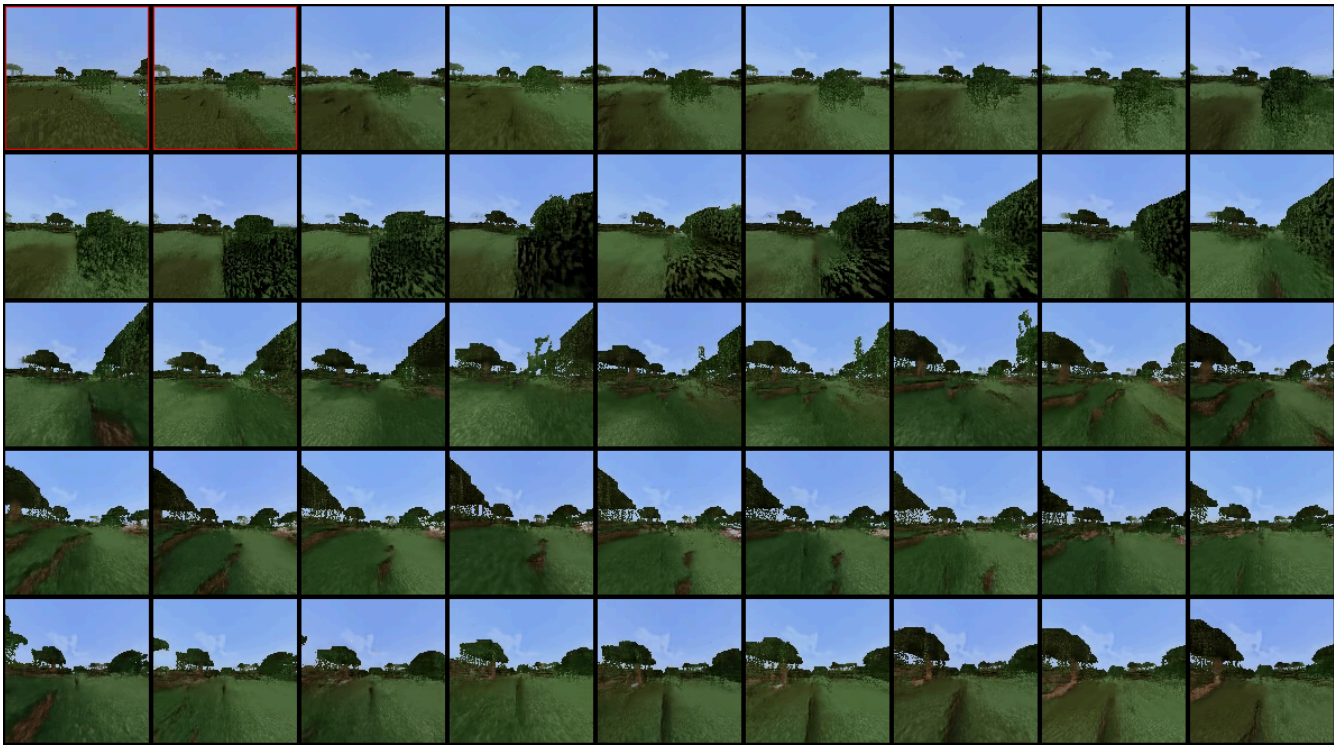


图8：Minecraft 上以72帧训练的 DF 能无滑窗滚动180帧。展示的是对这180帧未挑选的稀疏采样；模型实际还能生成更长，如2000帧。

> 72 180 180

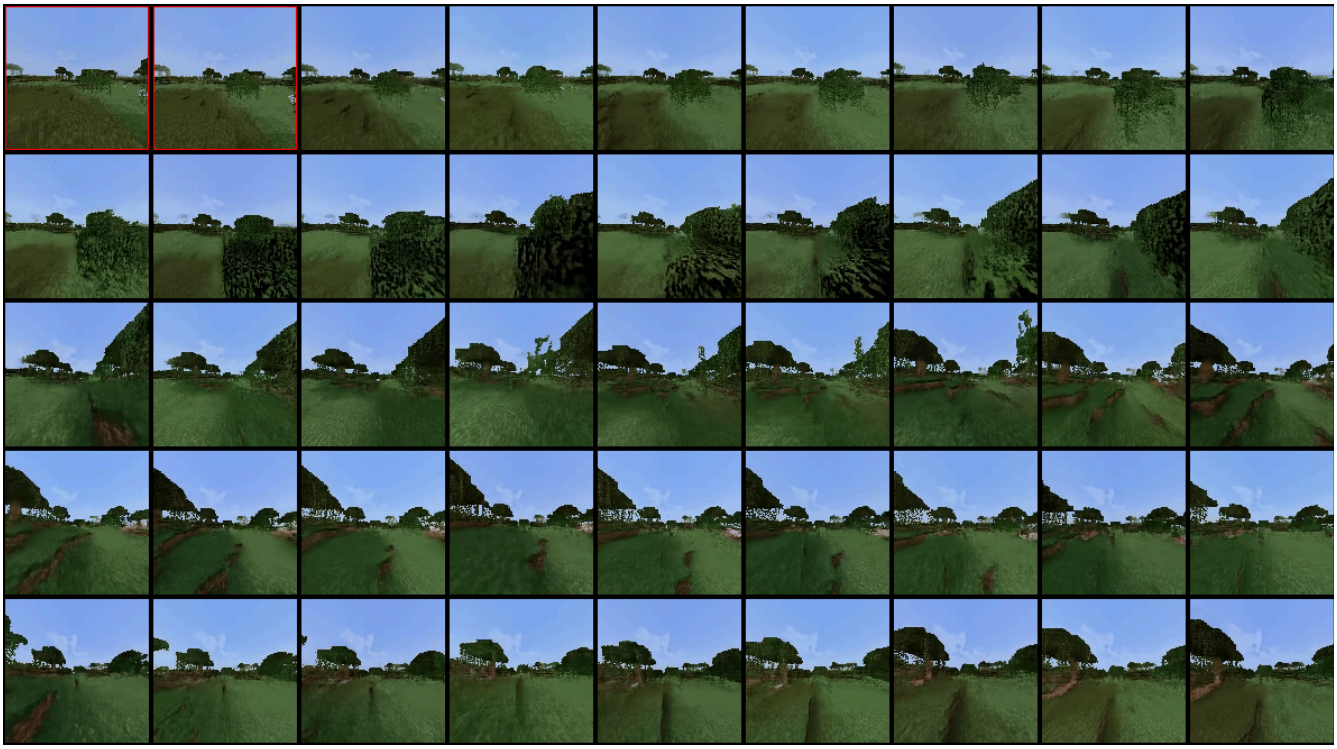


图9：Minecraft 上以72帧训练的 DF 能无滑窗滚动180帧。展示未挑选的稀疏帧；可生成更长，如2000帧。红色标记的前几帧是真实数据，用作条件。

> 72 180 180

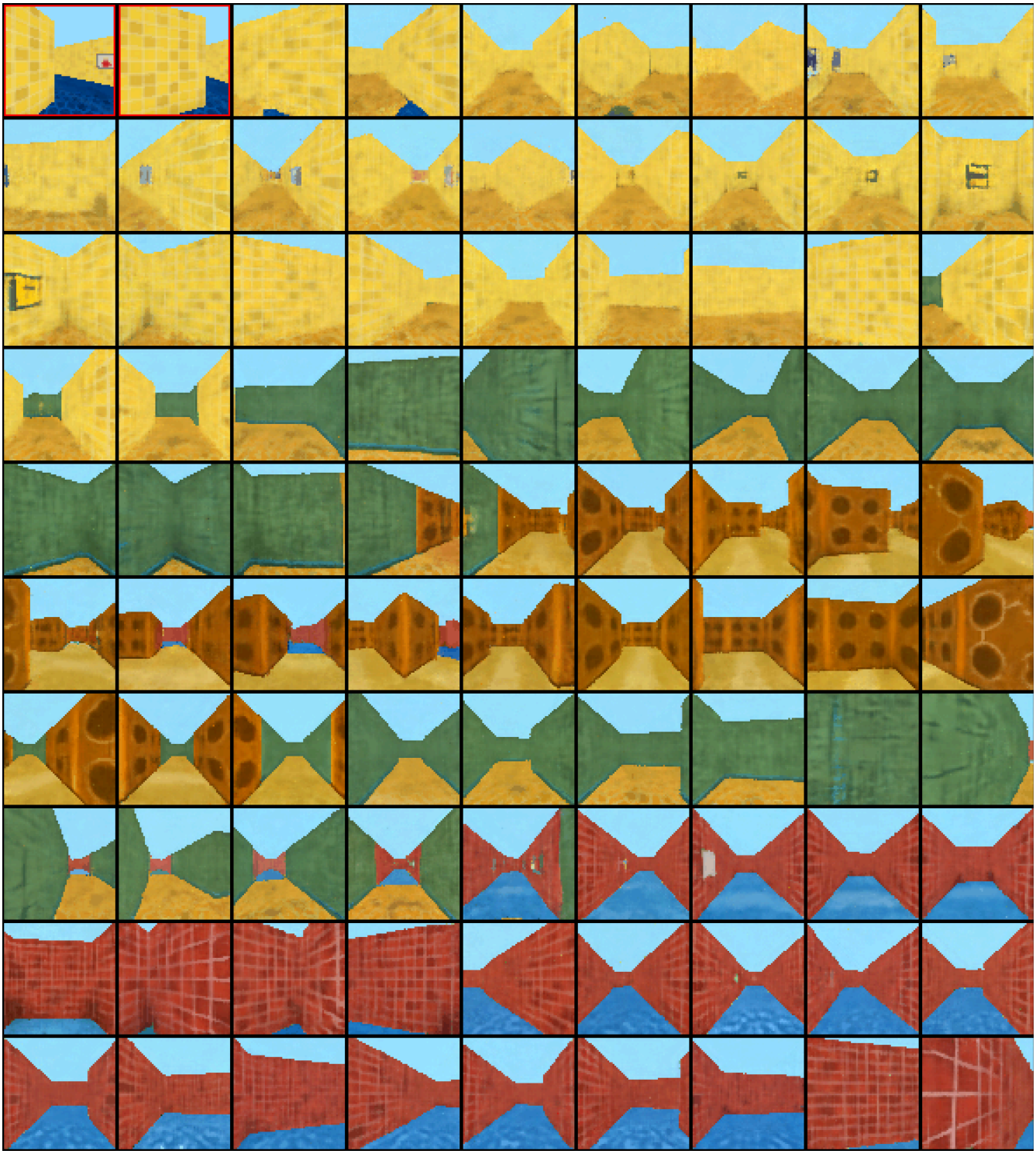


图10：DMLab 上以36帧训练的 DF 能无滑窗滚动180帧。展示未挑选稀疏帧；模型可在该数据上近乎无限滚动。红色前几帧为条件真值。

> 36 180 180

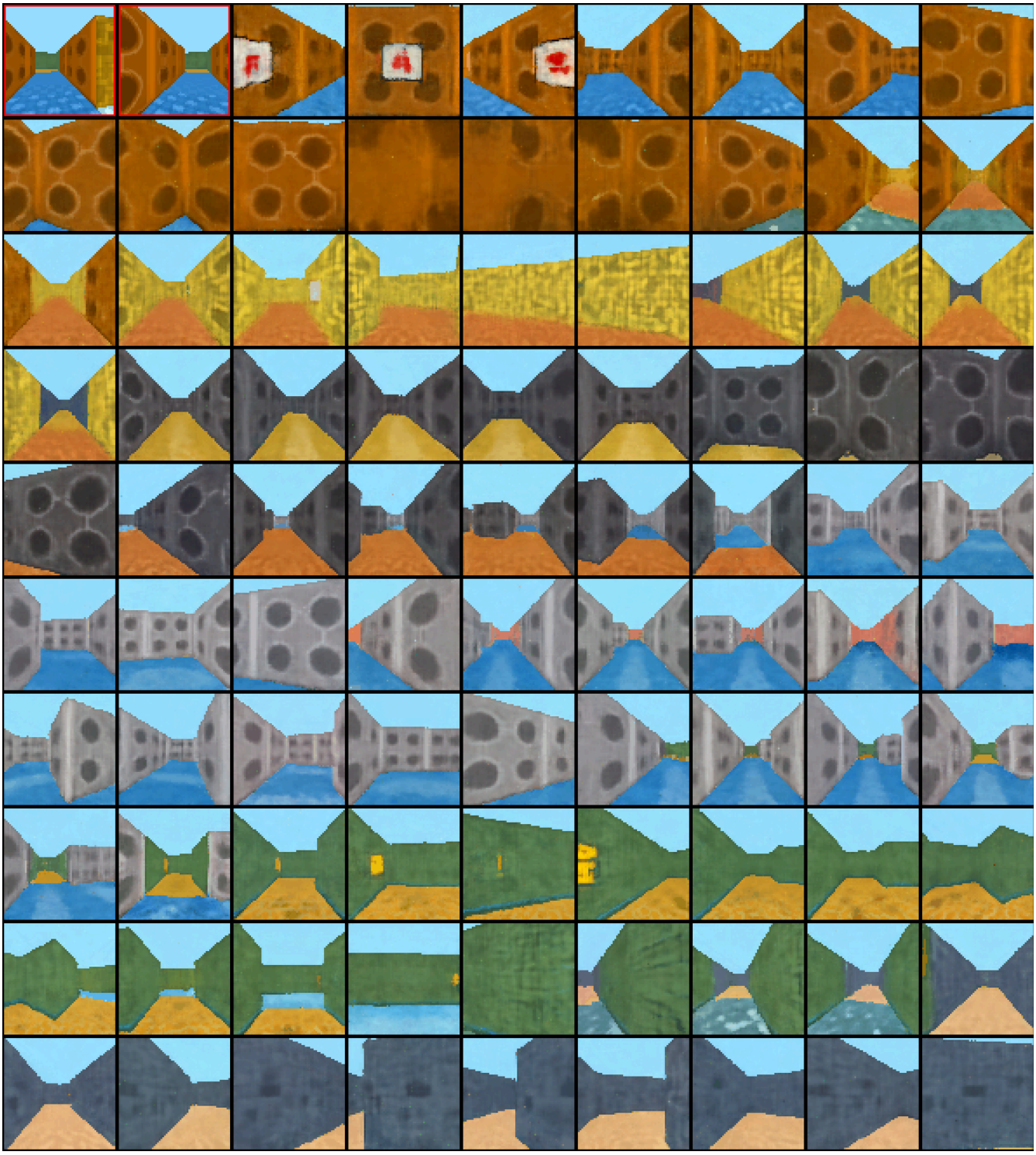


图11：DMLab 上以36帧训练的 DF 能无滑窗滚动180帧。展示未挑选稀疏帧；模型可近乎无限滚动。红色前几帧为条件真值。

> 36 180 180

一致性

我们还提供只在最大训练长度以内生成的结果。如图12、13，DF 能生成一致视频，结果未挑选。

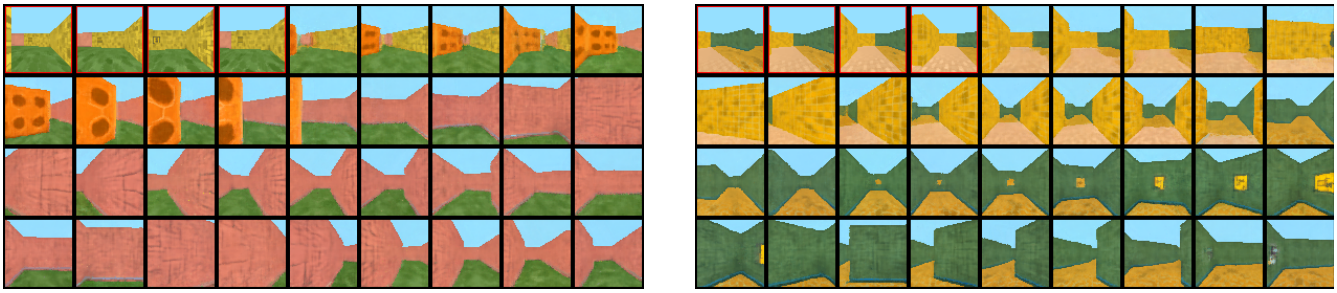


图12：DMLab 在最大训练长度内的额外、未挑选视频预测结果。红色前几帧为条件真值。

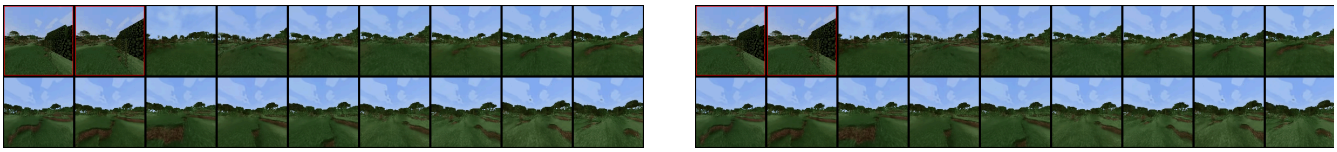


图13：Minecraft 在最大训练长度内的额外、未挑选视频预测结果。红色前几帧为条件真值。

E.4 更多规划结果

图15提供更多因果规划可视化；还展示 DF 实际运行模型预测控制的过程。如图14，由于时域灵活，DF 可生成更短计划。

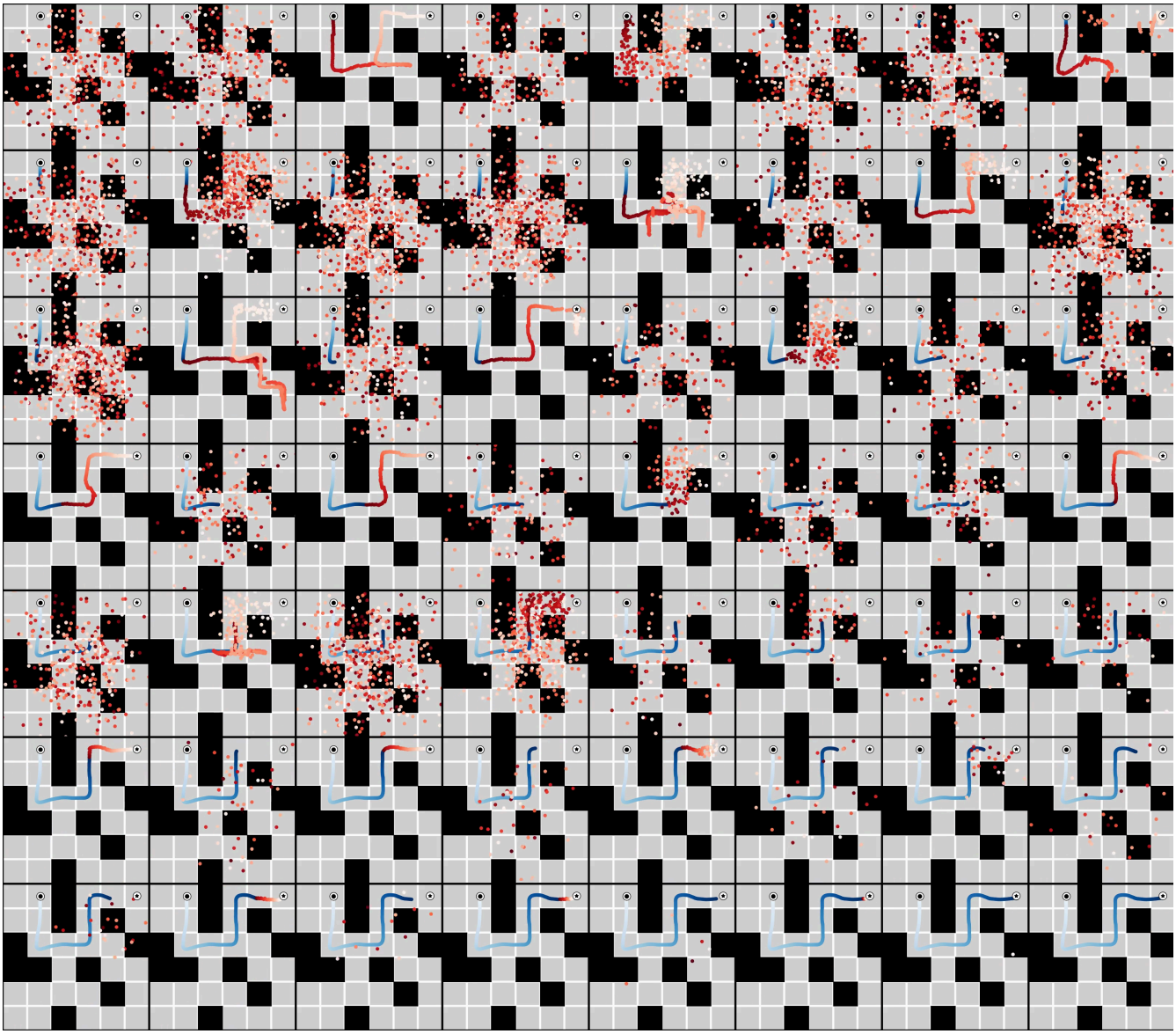


图14：中型迷宫中的 MPC 示例。蓝色为已实际执行轨迹，红色为计划。

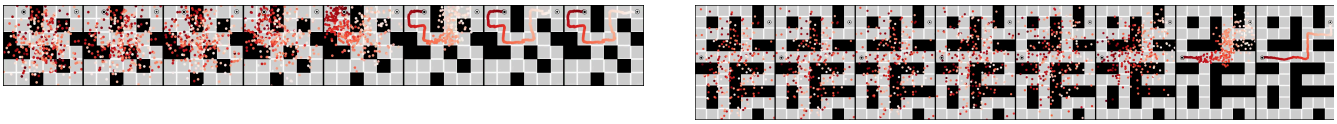


图15：中型迷宫（上）与大型迷宫（下）的计划生成示例。

E.5 真实机器人实验设置

图16展示观测受污染的机器人设置。训练收集数据时桌上没有 Target 购物袋，测试时零样本加入该袋，进行受污染观测模仿实验。典型失败是机器人不再响应物体随机位置的视觉线索；未观察到它因视觉干扰而胡乱行动。



图16：随机把 Target 购物袋扔在桌上，作为强视觉干扰。可提示 DF 把观测视为损坏，而非真值。

附录F 数据集更多细节

F.1 视频扩散数据集

采用 TECO [69] 使用的 Minecraft 与 DMLab 视频预测数据。

Minecraft 导航

数据是在 Minecraft 沼泽生物群系中随机行走的第一人称视频。智能体采用“疾跑跳跃”，能跨过单方块障碍而不被卡住；大多数时间直走，偶尔左右转。视频128×128像素，长视频裁为72帧子序列。虽提供配对动作，我们将其舍弃，以增加预测随机性。计算有限，只训练约10%的子序列。

> $128 \quad 72 \quad 10\%$

数据的问题是遇到两格或更高障碍时，智能体会卡住，整个视频只剩灰色花岗岩或棕色泥土纹理。这类帧很多，使模型预测无意义图像；我们认为这是数据集本身的问题。

DMLab 导航

DeepMind Lab 数据是在三维迷宫中的随机行走。分辨率64×64像素，使用48帧子序列，训练时同样舍弃提供的动作。（E.3所报36帧训练长度与本节48帧存在底本口径差异，均原样保留。）

> 64 48

Stable Video Diffusion [4] 所扩散的 VQ-VAE 潜变量也只有128×128×3，这表明 DF 有潜力借助预训练图像编解码器扩展到高分辨率。数据量极大且计算有限，我们只用约10%序列训练；观察到从测试初始帧已能生成良好视频。

> $128 \times 128 \times 3$ 10%

F.2 规划数据集

D4RL [18] 是标准离线强化学习基准，含多类环境；每个环境提供状态、动作与奖励轨迹组成的离线交互数据。

同 Diffuser [37]，选择三个迷宫，因为它们是长时域、多模态、稀疏奖励问题，适合可视化及评估规划。环境为 maze2d-medium-v1、maze2d-large-v1、maze2d-umaze-v1。控制二维加速度使机器人到达目标；观测4维，包括二维位置与速度；动作2维。起点随机，每个迷宫目标固定。进入以目标为心、半径0.5的圆内奖励1，否则0。

> 4

离线数据由随机行走组成。作者先把交叉口和转弯设为路点，再编程令智能体随机地在路点间导航，因而与墙无碰撞。轨迹不朝特定目标，给数据引入随机性。

沿用主要基线 Diffuser [37] 的选择：舍弃数据奖励，只按目标规划；还评估每环境的多目标变体，目标同起点一样随机，表中标为 multi。

F.3 机器人学习数据集

任务见4.4：桌面三个槽，苹果随机在A或B，橙子在另一个；借空槽C交换两个水果，每次只能移到空槽。例如苹果A、橙子B，可先苹果A→C，再橙子B→A，最后苹果C→B。图4说明其非马尔可夫性：当苹果B、橙子C时，不知道初始布局就无法决定当前动作。

桌上贴纸标出槽位圆形区域，直径约水果两倍；还随机水果在槽内的位置，确保必须视觉反馈。用 VR 遥操作和阻抗控制收集 Franka 的150条专家示范，两种初始配置各半。记录手部相机和拍摄全部三槽的前视相机；每条示范含机械手6自由度动作。成功交换后，终态自然成为另一随机配置的初态，我们利用这点节省采集时间。

> 150 6

每条示范500—600帧及动作。DF 在整条序列上训练，但相邻帧视觉接近，因此填充、降采样为40帧，每帧绑定15个动作。

> $500-600$ 40 15

F.4 时间序列数据集

数据集	维度	数值域	频率	训练步数	预测长度
Exchange	8	正实数	工作日	6,071	30
Solar	137	正实数	小时	7,009	24
Electricity	370	正实数	小时	5,833	24
Traffic	963	(0,1)	小时	4,001	24
Taxi	1,214	自然数	30分钟	1,488	24
Wikipedia	2,000	自然数	日	792	30

表3：用于时间序列预测的 GluonTS 数据特征。表内维度、数值域、频率、训练长度和预测长度完整重排保留。数据由 GluonTS [2] 提供，沿用 [72,42,56] 等既有工作，覆盖汇率、电网等真实高维动态。我们用自己的数据加载器封装 GluonTS 预处理。每个数据集由一条长多变量训练序列与一组短测试序列组成；验证集从训练序列中随机抽子序列，数量与留出测试集相同。全部划分均用训练特征的均值和标准差归一化。

协变量

统计模型常受益于额外人工特征。可构造协变量序列帮助识别季节性及其他时间依赖。遵循 [51]，按采样频率构造：包括滞后输入、学习的嵌入，以及依数据频率而定的小时、日期等手工时间特征。因此即使推理时，整个区间协变量也已知。可按下式纳入概率框架。

> $\boldsymbol{c} = \{\mathbf{c}_t\}_{t=1}^{T} \qquad [1, T]$

$$q\left(\mathbf{x}_{t_0:T} \mid \mathbf{x}_{1:t_0-1}, \mathbf{c}_{1:T}\right) := \prod_{t=t_0}^{T} q\left(\mathbf{x}_t \mid \mathbf{x}_{1:t_0-1}, \mathbf{c}_{1:T}\right). \qquad \text{(F.1)}$$

协变量收益高度依赖数据、模型特征及特征工程方式。

指标

连续排序概率分数 CRPS [46] 衡量预测分布与真值分布匹配程度，定义如下。

$$\mathrm{CRPS}(F, x) = \int_{\mathbb{R}} \left(F(z) - \mathbb{I}\{x \le z\}\right)^2 \mathrm{dz},$$

其中 F(z) 是预测值的一维累积分布，x 为真实观测，指示函数在 x≤z 时为1，否则0。对模拟样本和真值的 D 维序列沿特征求和，可得到预测和的分布与真值和，再报告下列 CRPS-sum。

> $F(z) \qquad x \qquad \mathbb{I}\{x \le z\} \qquad x \le z \qquad D \qquad \hat{F}_{\mathrm{sum}}(t) \qquad \sum_i x_{i,t}^0 \qquad \mathrm{CRPS}_{\mathrm{sum}}$

$$\mathrm{CRPS}_{\mathrm{sum}} = \mathbb{E}_{t \sim \mathcal{U}(t_0, T)}\left[\mathrm{CRPS}\left(\hat{F}_{\mathrm{sum}}(t), \sum_i x_{i,t}^0\right)\right]$$

该指标对预测窗口取平均；数值越低，预测分布越接近数据分布。

> $\mathrm{CRPS}_{\mathrm{sum}}$

具体地，先人工沿特征维求和；每时刻从100条采样轨迹取19个分位等级，估计预测和的累积分布。再用 GluonTS [2] 实现算 CRPS，以 CRPS-sum 报于表中。虽然聚合自行完成，我们验证了相对 GluonTS 实现的数值误差，比所报指标的精度阈值低若干数量级。

> $\hat{F}_{\text{sum}}(t)$ t CRPS CRPS_{sum}