arXiv:2603.00935v1 [stat.ML] 1 Mar 2026

时间感知潜在空间贝叶斯优化

Tuan A. Vu阿尔托大学 tuan.t.vu@aalto.fi

Julien Martinelli阿尔托大学

Harri Lähdesmäki阿尔托大学

摘要

潜在空间贝叶斯优化（LSBO）通过在生成模型的连续潜在空间中进行搜索，将贝叶斯优化扩展到分子设计等结构化领域。然而，大多数 LSBO 方法都假设目标函数固定不变，而实际设计活动往往会面临时间漂移（例如偏好变化或目标转移）。将时变贝叶斯优化引入 LSBO 并非易事：漂移不仅会影响代理模型，还会影响由表征所诱导的潜在搜索空间几何结构。我们提出了时间感知潜在空间贝叶斯优化（TALBO），通过高斯过程先验变分自编码器，在代理模型和学习到的生成式表征中都纳入时间因素，从而使潜在空间与目标函数的演变保持一致。为系统评估时变 LSBO，我们将广泛使用的分子设计任务调整为针对漂移的多属性目标，并引入适用于变化目标的评估指标。在这些基准测试中，TALBO 始终优于表现强劲的 LSBO 基线方法，并且在不同漂移速度和设计选择下保持稳健；即使面对实际上的时不变目标函数，其表现也仍具竞争力。

1 引言

许多现代设计问题都可归结为在大规模结构化空间中优化一个黑箱函数。典型示例包括 从头 分子设计 [Gómez-Bombarelli 等人、 2018], 蛋白质和抗体工程 [Torres 等人、2024、Amin 等人、 2024], 以及晶体设计 [Boyar 等人， 2025]。在这些领域中，候选设计是离散的高维对象，需要通过昂贵的模拟、实验室测定或人类专家进行评估，因此目标函数只能通过此类查询获得。

应对这类问题的首选框架是贝叶斯优化（BO）。这是一种样本高效的方法，它维护一个概率代理模型，通常为高斯过程（GP，Rasmussen 和 Williams 2006），并通过采集函数结合后验均值与不确定性，以指导连续评估 [Garnett，2023]。然而，原始贝叶斯优化方法并不容易适用于分子或生物序列等离散且高度结构化的领域。潜空间贝叶斯优化（LSBO）通过将贝叶斯优化与深度生成模型相结合，克服了这一局限：深度生成模型将结构化对象嵌入连续潜在空间，在该空间中进行优化，然后再将候选项解码回原始领域 [Gómez-Bombarelli 等人，2018，Tripp 等人，2020，Maus 等人，2022，Lee 等人， 2025]。这一表述已成为从头 分子和蛋白质设计的标准范式。

LSBO 方法所依据的一个关键假设是，目标函数在整个优化过程中保持不变。然而，在许多实际设计场景中，由于底层系统或优化标准本身发生变化，目标函数会随时间演变。例如，针对快速变异病原体进行抗体发现时，新出现的变体会改变抗体结合景观，因此必须更新设计或增强其鲁棒性 [Frei 等人， 2025]人在回路优化 [Sundin 等人，2022，Xu 等人， 2024]；以及多目标设计场景，其中偏好（标量化权重）会随时间演变 [Vallerio 等人，2015，Rácz 等人， 2025]。未能考虑这种时间变化，可能导致潜在表征过时，以及代理模型不再反映当前目标，进而降低优化性能。尽管时变贝叶斯优化已在连续域中得到研究 [Bogunovic 等人，2016年，Bardou 等人， 2024], 但由于搜索过程由学习得到的生成式表征介导，对 LSBO 中时间变化的细致处理仍在很大程度上尚未得到探索。

贡献。 我们提出了 时间感知潜在空间贝叶斯优化 (TALBO)，这是一种用于时变潜空间优化的框架，能够同时对 代理模型和学习到的表示中的时间漂移进行建模。图 1

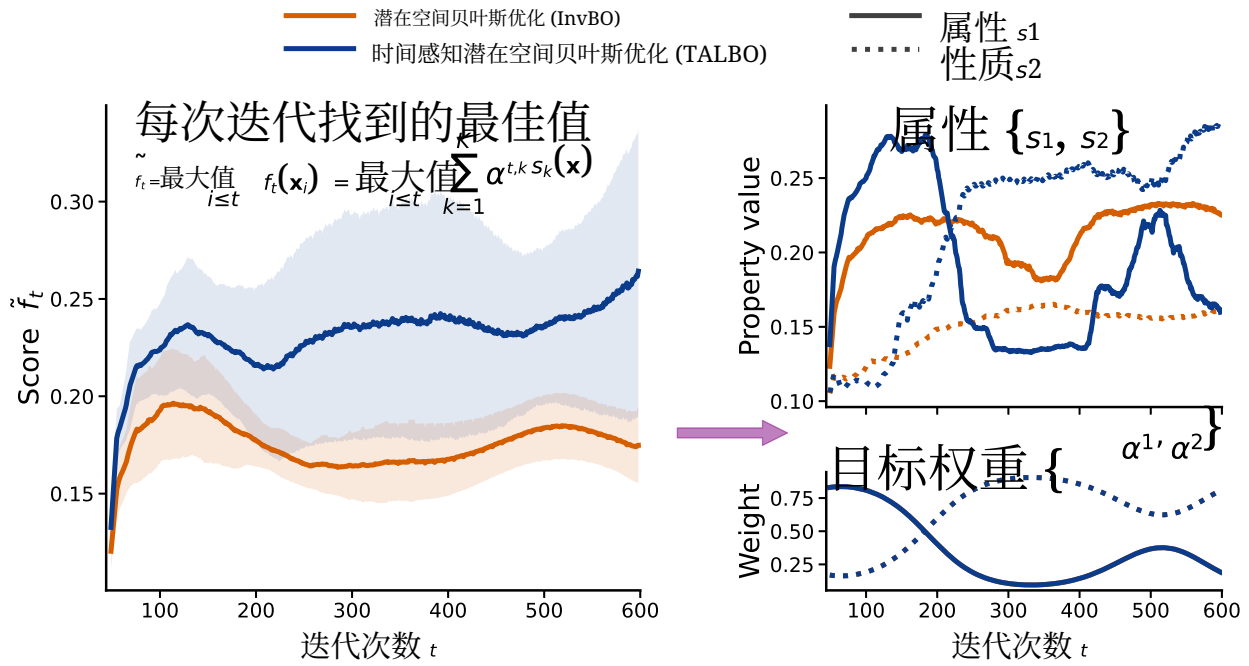

图1：**我们的设置概览**。在每次迭代 t中，黑箱目标函数 f_t 定义为两种分子性质的加权组合，并采用预先设定的时变权重计划（详情见第6.1节）。左：当前目标函数 f_t 下，截至迭代 t已评估的所有分子中的最佳得分（± 均值 ± 标准差，基于 10 个采用不同初始数据集的随机种子）。右上：性质分量的均值。右下：目标权重（所有随机种子均相同）。**我们的时间感知方法 TALBO 能更迅速地适应目标重要性的变化，从而取得˜始终更高的目标值** f_t。

提供了示意图　　览了该设置，并强调时间感知对于追踪不断变化的
结构化设计中的目标函数　　至关重要。我们的贡献如下：

- **概念层面**。我们对时变潜空间贝叶斯优化进行了形式化：在这一优化设置中，目标函数会随轮次变化，而搜索则在学习得到的潜在空间中进行。这一形式化揭示了标准潜空间贝叶斯优化中不存在的一项关键挑战——时间既会影响目标模型，也会影响潜在嵌入——并由此推动了应对时间漂移的原则性机制。
- **方法学**。 我们提出 TALBO，它将时空代理模型与由时间及可能的其他协变量引导的条件生成表示相结合，使潜在搜索空间与当前目标函数对齐。
- **实证**。 我们将广泛使用的分子设计基准适配到受控时变情境，并在固定预算下开展大量实验。TALBO 持续优于强劲的 LSBO 基线，一系列广泛的受控研究表明，这些增益在不同设计选择和实验设置下均具有稳健性。

2 预备知识

设 $\mathcal{X}$为输入空间， $\mathcal{Y} \subset \mathbb{R}$为输出空间。我们研究时变黑盒优化，其中在每次迭代 $t \in \{1, \dots, T\}$时，学习器观测目标函数 f_t 的带噪评估： $\mathcal{X} \to \mathcal{Y}$,

$$y_{tb} = f_t(\boldsymbol{x}_{tb}) + \varepsilon_t, \qquad \varepsilon_t \sim \mathcal{N}(0, \sigma^2), \tag{1}$$

对于批次 $\{\boldsymbol{x}_{t1}, \dots, \boldsymbol{x}_{tB}\} \in \mathcal{X}^B$，旨在选择一个在不断演化的目标函数下表现良好的查询 $\boldsymbol{x}_{t+1}$（或批次）。在方法描述中，我们可以假设 $B = 1$。由于 $\operatorname{argmax}_{\boldsymbol{x} \in \mathcal{X}} f_t(\boldsymbol{x})$ 可能随 t 变化，目标并非恢复单一的静态最大化点，而是自适应地跟踪高价值输入，随着 f_t 漂移。

$\mathcal{X}$可能是离散的、高维的且结构化的（例如分子），从而使优化变得困难。这带来两项挑战：处理时变目标，以及在复杂搜索空间中导航。接下来，我们介绍用于应对这些问题的组件：时变贝叶斯优化，以及面向结构化设计的学习表示。

2.1 时变贝叶斯优化

贝叶斯优化提供 一种用于黑盒优化的序贯策略，并且可以扩展到时变目标函数。 s.

贝叶斯优化。 给定 $\mathcal{D}_t = \{(\boldsymbol{x}_i,\ t_i,\ y_i)\}_{i=1}^{t}$，贝叶斯优化会针对目标函数维护一个概率代理模型，记为 $p(f \mid \mathcal{D}_t)$，该模型会为 $f(\boldsymbol{x})$ 给出预测分布。下一次查询通过最大化依赖于该代理模型的采集函数 来选取，

$$\boldsymbol{x}_{t+1} \in \arg\max_{\boldsymbol{x}\in\mathcal{X}} \alpha\big(\boldsymbol{x};\, p(f \mid \mathcal{D}_t)\big), \tag{2}$$

其中，α 在代理模型下权衡预测值（利用）与不确定性（探索）。

高斯过程。 概率代理模型的一种常见选择是高斯过程（GP），

$$f \sim \mathcal{GP}\big(\mu(\cdot), k(\cdot,\cdot)\big), \tag{3}$$

其中 μ 通常设为零，而 k 是一个半正定核函数，用于编码先验平滑性。假设采用高斯似然且超参数固定，在给定 $\mathcal{D}_t$ 的条件下，可得到 $f(\boldsymbol{x})$ 的高斯预测分布，其均值和方差均有闭式解。然而，由于精确推断的计算复杂度随 $|\mathcal{D}_t|$呈立方增长，实践中通常采用稀疏近似或变分近似Titsias [2009]。

时变贝叶斯优化。在这种情况下，目标函数会随轮次变化，即 $f_t(\boldsymbol{x})$，因此代理模型必须对时间演化进行建模，并调整历史观测的相关性。一种常见方法是在时空函数 $f_t(\boldsymbol{x})$ 上设置高斯过程先验：$= f(\boldsymbol{x}, t)$，并使用定义在 $(\boldsymbol{x}, t)$ 上的核函数，例如 $k\big((\boldsymbol{x}, t)、(\boldsymbol{x}', t')\big) := k_x(\boldsymbol{x}, \boldsymbol{x}')\ k_t(t, t')$。第 t 轮的采集函数根据当前轮次 $f(\cdot, t)$ 的后验预测分布计算 [Bogunovic 等人， 2016]。

评估指标。 随着目标函数随时间变化，标准的迄今最优指标已不足以满足需求。因此，我们考虑两项评估标准。(i)当前目标下的迄今最优值，

$$\tilde{f}_t = \max_{i\le t} f_t(\boldsymbol{x}_i) \tag{4}$$

它会评估此前查询过的所有设 计在当前目标函数 f_t 下的表现，并报告在时 f_t 间 t
的可达到最优值。(ii) 累积 遗憾，

$$R_T := \sum_{t=1}^{T} \big(f_t^\star - \tilde{f}_t\big), \tag{5}$$

˜其中 $f_t^\star$：$=$ 最大值$_{\boldsymbol{x}\in\mathcal{X}}\ f_t(\boldsymbol{x})$ 表示时间 t下目标函数的最优值。值得注意的是，f_t 通常并不单调；当目标函数发生变化、先前表现良好的设计与新的权衡不再那么契合时，其值可能会下降，如图1所示。

2.2 潜在空间贝叶斯优化

当 $\mathcal{X}$ 是离散的、具有结构或维度很高时，直接优化会很困难。潜空间贝叶斯优化通过学习连续表示来解决这一问题，例如使用变分自编码器（VAE） [Kingma 和 Welling， 2013]。

VAE 定义了概率编码器 $q_\phi(\boldsymbol{z} \mid \boldsymbol{x})$ 和解码器 $p_\theta(\boldsymbol{x} \mid \boldsymbol{z})$。在实际应用中，潜空间贝叶斯优化基于确定性嵌入进行操作，例如后验均值 $\mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x})}[\boldsymbol{z}]$，以及解码映射

$$\boldsymbol{\Gamma}(\boldsymbol{z}) := \arg\max_{\boldsymbol{x}} p_\theta(\boldsymbol{x} \mid \boldsymbol{z}), \tag{6}$$

它将潜在点映射回设计空间。这会形成一个连续潜在空间 $\mathcal{Z} \subset \mathbb{R}^d$，可在其中执行贝叶斯优化：

$$\boldsymbol{z}_{t+1} \in \arg\max_{\boldsymbol{z}\in\mathcal{Z}} \alpha\big(\boldsymbol{z}; p(g \mid \mathcal{D}_t^z)\big), \tag{7}$$

其中

$$g(\boldsymbol{z}) := f(\boldsymbol{\Gamma}(\boldsymbol{z})), \qquad \mathcal{D}_t^z = \{(\boldsymbol{z}_i, t_i, y_i)\}_{i=1}^{t}. \tag{8}$$

所选潜在点解码为

$$\boldsymbol{x}_{t+1} = \boldsymbol{\Gamma}(\boldsymbol{z}_{t+1}). \tag{9}$$

表征更新。原始潜在空间贝叶斯优化在优化过程中保持生成模型固定。一些变体会定期重新训练该模型，通常与代理模型联合训练 [Chu 等人，2024，Maus 等人， 2022]，以便使潜在几何与观测值更好地对齐。然而，这些方法假设目标函数是静态的，且无法应对时间漂移。

3 问题陈述

将潜空间贝叶斯优化扩展到时变目标会带来结构性挑战。在潜空间贝叶斯优化中，优化是在学习得到的潜在空间 $\mathcal{Z}$ 中进行的，其几何结构由生成模型决定。在时间漂移的情况下，静态的潜在表示可能会与不断变化的目标函数失去对齐。

经典时变贝叶斯优化通过使用时空核对 $f(\boldsymbol{x}, t)$ 进行建模，在代理模型层面处理漂移。然而，在潜空间贝叶斯优化中，代理模型通过以下方式对潜在编码进行操作：

$$g(\boldsymbol{z}, t) := f_t(\boldsymbol{\Gamma}(\boldsymbol{z})), \tag{10}$$

其中， $\boldsymbol{\Gamma}$表示解码器。因此，如果潜在几何本身保持静态，仅在代理模型中对时间进行建模可能并不足够。

双重时间建模。 我们在两个层面引入时间：时空高斯过程在潜空间中对 $g(\boldsymbol{z}, t)$ 进行建模，同时映射 $\boldsymbol{x} \mapsto \boldsymbol{z}$ 采用潜函数上的高斯过程先验。这种联合处理使不确定性和潜在几何都能随时间漂移而自适应。

4 基于高斯过程先验生成模型的时间感知潜空间贝叶斯优化

我们提出 时间感知潜在空间贝叶斯优化 （TALBO），通过采用具有可扩展基函数近似和全局变分参数的高斯过程先验变分自编码器（DG-BFGP）来实现双重时间建模 [Balık 等人， 2025]。据我们所知，这是时间深度潜变量模型首次应用于贝叶斯优化。第4.1 节介绍潜在表示，第4.2 节介绍目标函数，第4.3 节介绍时间的处理方式。算法1总结了完整的 TALBO 循环。

4.1 结构化潜在表示

每个设计 $\boldsymbol{x} \in \mathcal{X}$ 都与 R 辅助协变量 $\mathbf{c}(\boldsymbol{x}, \ t) \in \mathbb{R}^R$相关联，这些协变量可能包括连续描述符和时间。对这些协变量进行显式建模，使潜在几何能够捕捉结构化且在时间上连贯的变化；这对于应对目标漂移至关重要。给定协变量 $\mathbf{c} = (c^{(1)}, \ \ldots\ldots, \ c^{(R)}) := \mathbf{c}(\boldsymbol{x}, \ t)$，DGBFGP 定义了潜在编码 $\boldsymbol{z}(\mathbf{c}) \in \mathbb{R}^d$ 和解码器 $p_\theta(\boldsymbol{x} \mid \boldsymbol{z})$。潜在几何由定义在协变量依赖的潜函数上的加性高斯过程先验诱导。

基函数高斯过程先验。对于每个协变量分量 $c^{(r)}$，我们对单变量潜在函数 $h^{(r)}$设定一个高斯过程先验，

$$h^{(r)}(c^{(r)}) \sim \mathcal{GP}\big(0, k^{(r)}(\cdot, \cdot)\big), \tag{11}$$

其中 $k^{(r)}$ 是一个协方差核，其超参数为 σ_r， ℓ_r。为获得可扩展的参数化形式，DGBFGP 采用希尔伯特空间基函数近似 [Solin 和 Särkkä， 2020], 从而得到线性形式

$$h^{(r)}(c^{(r)}) \approx \boldsymbol{a}^{(r)\top} \boldsymbol{\phi}^{(r)}(c^{(r)}), \tag{12}$$

$$\boldsymbol{a}^{(r)} \sim \mathcal{N}\big(\mathbf{0}, \boldsymbol{S}^{(r)}(\sigma_r, \ell_r)\big), \tag{13}$$

其中 $\boldsymbol{\phi}^{(r)}(\cdot) \in \mathbb{R}^M$ 表示希尔伯特特征，且 $\boldsymbol{S}^{(r)}(\sigma_r, \ell_r)$ 为对角矩阵。对于连续协变量，我们使用平方指数核。希尔伯特空间特征的详细构造见附录 C.2。

高斯过程先验变分自编码器模型假设每个潜在维度都具有相互独立的高斯过程先验 [Casale 等人， 2018]。定义$\boldsymbol{A}^{(r)} = [\boldsymbol{a}^{(r,1)\top}, \ldots, \boldsymbol{a}^{(r,d)\top}]^\top \in \mathbb{R}^{d \times M}$，所有潜在维度的高斯过程先验的基函数近似为

$$\boldsymbol{z}(\mathbf{c}) = \boldsymbol{A}^{(r)} \boldsymbol{\phi}^{(r)}(c^{(r)}) \tag{14}$$

$$\boldsymbol{a}^{(r,l)} \sim \mathcal{N}\big(\mathbf{0}, \boldsymbol{S}^{(r)}(\sigma_r, \ell_r)\big), \tag{15}$$

其中，我们假设所有潜在维度共享核超参数。

加性潜在编码。DGBFGP 将潜在编码表示为各协变量效应之和：

$$\boldsymbol{z}(\mathbf{c}; \boldsymbol{A}) = \sum_{r=1}^{R} \boldsymbol{A}^{(r)} \boldsymbol{\phi}^{(r)}(c^{(r)}), \qquad \boldsymbol{A}^{(r)} \in \mathbb{R}^{d \times M}, \tag{16}$$

其中 $\boldsymbol{A} = [\boldsymbol{A}^{(1)}, \ldots \ \boldsymbol{A}^{(R)}]$。DGBFGP 通过为核超参数设置适当的先验来确保平滑性 :

$$\sigma_r, \ell_r \sim \text{Lognormal}(0, 1). \tag{17}$$

请注意 将时间作为协变量之一纳入，使潜在几何显式依赖于时间 t.

算法 1 TALBO（简化版）

要求：初始评估集 $\{(\boldsymbol{x}_i, t_i, y_i)\}_{i=1}^{n_0}$。**1：在初始语料库上训练 DGBFGP，以获得** $q_0(\boldsymbol{A}, \sigma, \ell)$。**2：构建初始贝叶斯优化数据集** $\mathcal{D}_0 = \{(\boldsymbol{z}_i, t_i, y_i)\}_{i=1}^{n_0}$, $\boldsymbol{z}_i = \mathbb{E}_{q_0}[\boldsymbol{z}(\mathbf{c}(\boldsymbol{x}_i, t_i); \boldsymbol{A})]$。**3：对于** $t = 1$ **到** T，**或直到满足条件为止，执行以下操作：4：在** $\mathcal{D}_{t-1}$ **上拟合时空代理模型（式 S20）**。**5：选择** $\hat{\boldsymbol{z}} \in \arg\max_{\boldsymbol{z}\in\mathcal{Z}} \alpha(\boldsymbol{z};\ p(g(\cdot, t) \mid \mathcal{D}_{t-1}))$。**6：解码设计** $\boldsymbol{x}_t \sim p_{\theta_{t-1}}(\cdot \mid \hat{\boldsymbol{z}})$。**7：观测** $y_t = f_t(\boldsymbol{x}_t) + \varepsilon_t$ **并计算** $\mathbf{c}_t = \mathbf{c}(\boldsymbol{x}_t, t)$。**8：计算嵌入** $\boldsymbol{z}_t \leftarrow \mathbb{E}_{q_{t-1}}[\boldsymbol{z}(\mathbf{c}_t; \boldsymbol{A})]$。**9：更新贝叶斯优化数据集** $\mathcal{D}_t \leftarrow \mathcal{D}_{t-1} \cup \{(\boldsymbol{z}_t, t, y_t)\}$。**10：如果触发表示更新（算法 2-L18），则执行以下操作：11：通过在当前数据上进行热启动随机变分推断，更新 DGBFGP 后验分布和解码器参数。**(q_t, θ_t)。**12：通过使用** q_t **重新嵌入设计，重新计算** $\mathcal{D}_t$。**13：结束 if**。**14：结束 for**。

4.2 变分学习

给定训练集 $\{(\boldsymbol{x}_n,\ \mathbf{c}_n)\}_{n=1}^{N}$，我们通过解码器似然对设计进行建模

$$p_\theta(\boldsymbol{x}_n \mid \boldsymbol{A}, \mathbf{c}_n) := p_\theta\big(\boldsymbol{x}_n \mid \boldsymbol{z}(\mathbf{c}_n; \boldsymbol{A})\big). \tag{18}$$

我们使用因子化变分后验学习全局参数

$$q(\boldsymbol{A}, \sigma, \ell) = q(\boldsymbol{A})\, q(\sigma)\, q(\ell), \tag{19}$$

其中 $q(\boldsymbol{A})$ 服从高斯分布，$q(\sigma)$ 和 $q(\ell)$ 服从对数正态分布。证据下界的形式为

$$\begin{aligned} \mathcal{L}_{\mathrm{DGBFGP}} = \mathbb{E}_{q(\boldsymbol{A})}\Big[\sum_{n=1}^{N} \log p_\theta\big(\boldsymbol{x}_n \mid \boldsymbol{z}(\mathbf{c}_n; \boldsymbol{A})\big)\Big] \\ - \mathbb{E}_{q(\sigma)q(\ell)}\Big[\mathrm{KL}\big(q(\boldsymbol{A}) \,\|\, p(\boldsymbol{A} \mid \sigma, \ell)\big)\Big] \\ - \mathrm{KL}\big(q(\sigma) \,\|\, p(\sigma)\big) - \mathrm{KL}\big(q(\ell) \,\|\, p(\ell)\big), \end{aligned} \tag{20}$$

并通过小批量处理的随机梯度方法进行优化 [Balık 等人， 2025]。

除公式（20）中的变分目标函数外，我们的时间感知表示更新还纳入了两种互补机制。首先，我们加入对齐正则化器，以鼓励潜在几何相对于观测到的目标值保持平滑且尺度适当 [Lee 等人， 2023]。其次，我们应用潜空间反演，以缓解已评估设计与其潜在表征之间的不对齐 [Chu 等人， 2024]。最后，我们在潜在观测上训练变分高斯过程代理模型，并采用 SVGP 目标函数 [Hensman 等人， 2015]。这些组成部分分别详见附录 B.2、B.3和B.4。

4.3 时间条件化的潜在表示

由于时间 t 作为额外的连续协变量纳入 $\mathbf{c}(\boldsymbol{x},\ t)$，潜在编码 $\boldsymbol{z}(\mathbf{c}(\boldsymbol{x},\ t); \boldsymbol{A})$会依赖于时间，从而形成时间连贯的潜在几何。这使得表示能够随目标函数漂移而演变。

算法1 概述了由此产生的贝叶斯优化循环。在每次迭代中，我们使用 DGBFGP 协变量嵌入已评估设计，更新关于（$\boldsymbol{z}$，t）的时空代理模型，并选择一个潜在查询进行解码和评估。优化过程中可以更新生成模型；若更新，则会重新计算潜在嵌入以保持对齐。值得注意的是，DGBFGP 仅包含解码器，因此设计 $\boldsymbol{x}_i$ 的后验均值嵌入计算为 $\boldsymbol{z}_i = \mathbb{E}_{q_t(\boldsymbol{A})}[\boldsymbol{z}(\mathbf{c}(\boldsymbol{x}_i,\ t_i); \boldsymbol{A})]$。

大多数潜空间贝叶斯优化论文只描述高层流程，而省略了许多实践细节；在此，我们尽力明确列出忠实复现所需的各个模块和步骤。算法 [Lee 等人， 2023],以及潜空间反演 [Chu 等人， 2024], 的详细流程（包括更新触发条件和对齐正则化器）见算法 2。

5 相关工作

潜在空间贝叶斯优化 [Gómez-Bombarelli 等人， 2018] 借助变分自编码器等生成模型 [Kingma 和 Welling， 2013] 将结构化设计嵌入连续潜在空间，从而将传统贝叶斯优化扩展到离散、结构化或高维输入空间，并可在其中应用标准贝叶斯优化。然而，在固定的变分自编码器表示上进行贝叶斯优化可能会导致样本效率低下，因为潜在几何可能与优化目标不对齐。为缓解这一问题，近期研究探索了在优化过程中调整潜在空间的方法，包括加权重训练方案 [Tripp 等人， 2020], 度量学习目标 [Grosnit 等人， 2021], 以及对代理模型和生成映射进行联合推断的方法，例如 LOL-BO [Maus 等人， 2022]。CoBO [Lee 等人， 2023]提出了利普希茨约束微调目标，以使潜在表示更好地与目标函数对齐；InvBO [Chu 等人， 2024] 则将这类工作扩展到通过潜空间反演缓解解码器引起的错位。其他方法将生成模型与代理模型组件解耦 [Moss 等人， 2025] 并纳入领域专用核函数，例如 Tanimoto 核 [Tripp 等人， 2023]。关于潜空间贝叶斯优化方法的全面综述，参见 González-Duque 等人 [2024] 。尽管已有大量相关研究，时变目标的扩展仍未得到充分探索。在这类场景中，潜在几何的变化不仅可能源于表征更新，也可能由目标函数本身的时间漂移造成。

时变贝叶斯优化 针对目标函数随时间演变的情形。一种常见方法是将时间作为额外输入纳入时空代理核 [Marchant 和 Ramos，2012，Marchant 等人， 2014]。后续研究提出了管理过时观测的方法，包括定期重置数据集，或逐步折扣历史数据 [Bogunovic 等人，2016年，Brunzema 等人， 2022]。近年来，研究者提出了基于分布距离的自适应相关性度量，用于评估历史观测对未来预测的影响 [Bardou 等人， 2024]。尽管取得了这些进展，时变贝叶斯优化几乎完全是在原始输入空间中进行研究的。将其与潜空间优化相结合会带来额外挑战：在自适应潜空间贝叶斯优化中，潜在几何会随着表征更新而演变；而在时变设置下，它还可能因目标函数的时间漂移而发生变化。我们的方法同时考虑了这两种演变来源。

潜在表示的生成式骨干模型始于潜空间贝叶斯优化中的变分自编码器 [Kingma 和 Welling，2013，Gómez-Bombarelli 等人， 2018]。后续扩展将标准高斯先验替换为高斯过程先验，以编码潜在空间中的时间相关性或协变量相关性，例如 GPPVAE [Casale 等人， 2018], GP-VAE [Fortuin 等人， 2020] 以及加性高斯过程先验模型 [Ramchandran 等人， 2021], 及其贝叶斯优化应用 [Ramchandran 等人，2025]。更多背景信息请参见附录 B.7。近期工作，例如 DGBFGP [Balık 等人， 2025], 通过基函数近似和全局参数化提升了可扩展性。研究者也探索了其他骨干模型，例如使用归一化流获得可逆潜映射，并减轻重建引起的差异 [Lee 等人，2025]。

6 实验

基线方法。我们将 TALBO 与具有代表性的潜空间贝叶斯优化方法进行比较：InvBO [Chu 等人、 2024], CoBO [Lee 等人、2023], 以及 LOL-BO [Maus 等人， 2022]。TALBO 和这些基线方法均采用信赖域贝叶斯优化（TuRBO；Eriksson 等人 2019年），该方法通过将搜索限制在以有潜力的当前最优解为中心的自适应局部信赖域内，改善高维贝叶斯优化。此外，我们还报告了三种基线方法：不带信赖域约束的标准潜空间贝叶斯优化、与 TuRBO 结合的潜空间贝叶斯优化，以及一种随机搜索方法，该方法对潜在编码进行均匀采样并将其解码为候选分子。我们的方法分别采用 Maus 等人的[2022] SELFIES变分自编码器和我们提出的 SELFIES DGBFGP模型实现。更多实现细节见附录C。

实现细节。我们遵循Chu 等人 [2024] 的做法，并采用汤普森采样作为采集函数，使用变分高斯过程代理模型 [Hensman 等人， 2015] 并使用深度核 [Wilson 等人， 2016]以及径向基函数协方差函数（附录B.4）。代理模型和采集函数组件均在`BoTorch` [Balandat 等人中实现，2020]。在分子发现方面，我们同时考虑了Maus 等人 [2022] 提出的 SELFIES 变分自编码器，以及我们提出的 SELFIES DGBFGP 模型；二者均使用 GuacaMol数据集中 127 万个未标记分子进行预训练 [Brown 等人， 2019]代码将在论文被接收后公开。

6.1 多属性优化与动态变化的优先级

我们考虑一种受控场景，其中 K 属性得分的相对重要性随时间变化。该设定模拟偏好发生转变、从而产生时变目标的情形。

Task	评分函数
中位分子 1	模拟(樟脑, ECFC4) 模拟(薄荷醇, ECFC4)
中位分子 2	模拟(他达拉非, ECFC6) sim(西地那非，ECFC6)
复合相似度 1	sim(西格列汀，ECFC4) sim(雷诺嗪，AP) sim(奥希替尼，FCFC4)
复合相似度 2	模拟(塞来昔布，ECFC4) 模拟(曲格列酮，ECFC4) 模拟(替沃噻吨，ECFC4)

表 1：任务及其对应的评分函数。

令 $s_k(\mathbf{x}) \in [0,1]$ 表示 K 基础属性得分。在时间 t，目标函数定义为一个凸组合 on

$$f_t(\mathbf{x}) = \sum_{k=1}^{K} \alpha_{t,k}\, s_k(\mathbf{x}), \tag{21}$$

其中 $\boldsymbol{\alpha}_t \in \mathbb{R}_+^K$ 位于单纯形上，$\sum_{k=1}^{K} \alpha_{t,\ k} = 1$。

合成权重生成 $\boldsymbol{\alpha}$。 为了使优先级平滑地动态变化，我们对 K 个独立的潜函数进行采样

$$w_k(t) \sim \mathcal{GP}\big(0, k_w(t,t')\big), \qquad k = 1, \ldots, K, \tag{22}$$

其中 k_w 是时间协方差核。然后，我们定义

$$\alpha_{t,k} = \frac{\exp\big(\beta w_k(t)\big)}{\sum_{j=1}^{K} \exp\big(\beta w_j(t)\big)}, \quad k = 1, \ldots, K, \tag{23}$$

它将潜在高斯过程样本映射到概率单纯形。温度参数 $\beta > 0$控制优先级随时间集中分布的程度（附录C.1）。

属性得分 $s_\circ$。 为了构建需要在各组成部分之间进行明确权衡、且其相对重要性会发生变化的任务，我们基于分子相似度评分构造目标函数。遵循 Brown 等人 [2019], 提出的中位分子任务，我们选取了四种源自 GuacaMol 基准套件的评分函数。具体而言，其中两个任务对应于 中位分子 *1* 和 中位分子 *2* 基准测试中的组成部分，另外两个则改编自相似性和再发现任务。每个基础得分 $s_k(\boldsymbol{x}) \in [0,1]$ 和 $f_t(\boldsymbol{x}) = \sum_{k=1}^{K} \alpha_{t,\ k} s_k(\boldsymbol{x})$，因此 $f_t(\boldsymbol{x}) \in [0,1]$ 以及 $f_t^\star = 1$ 对所有 t均成立。完整的评分函数集详见表 1。

实验方案与预算。每次运行包含 $T = 600$ 个贝叶斯优化轮次。在贝叶斯优化循环开始前，我们通过从 GuacaMol 预训练语料库中均匀采样 100 个分子，并在前 50 次迭代下对每个分子进行评估，构建包含 $N_{\text{init}} = 100$ 次评估的初始数据集。对于每个随机种子，这些评估结果在各方法间共享；但时间无关基线方法仅获得相同的 100 个分子在最近时间步的评估结果。随后，在第 t轮中，选取一批 $B = 10$ 潜在点（通过 q批次汤普森采样），将其解码为分子，并在 f_t下进行评估。TALBO 使用附录C.2中所述的协变量。

6.2 结果

总体性能。图 2 通过三个互补视角比较了所有方法：当前目标函数的迄今最优值（公式 4）、与之对应的累积遗憾（公式5）以及基于遗憾的排名。在全部四项任务中（表 1），就单位时间最优值而言（第一行），TALBO 的总体表现最强，在优化时域的大部分时间内均取得最高值。竞争基线方法有时能在早期与 TALBO 持平甚至超过它，但一旦目标优先级发生变化，它们通常会陷入平台期或适应得更慢。累积遗憾（第二行）也呈现出相同趋势；为便于阅读，结果以相对于 InvBO 的百分比改进表示。在大多数任务中，TALBO 的遗憾曲线最有利，且其优势通常会随着优化的推进而扩大。第一项任务（第一列）是个例外，在该任务中，遗憾曲线较为接近，竞争基线方法在优化时域的较长时间内仍具竞争力。为以不受尺度影响的方式总结性能，第三行给出了根据累积遗憾得出的排名（越低越好）。当单位时间最优值或

基线	$\ell_w = 0.1$	$\ell_w = 0.2$	$\ell_w = 0.3$
TALBO	**1.98 ± 1.23**	**1.75 ± 1.08**	**1.90 ± 1.15**
InvBO	3.77 ± 1.39	3.90 ± 1.24	3.52 ± 1.38
CoBO	2.48 ± 1.04	2.45 ± 1.30	2.30 ± 1.16
LOL-BO	2.42 ± 1.01	2.73 ± 1.11	2.92 ± 1.10
LS-BO	6.41 ± 0.65	6.62 ± 0.60	6.59 ± 0.65
TurBO	4.76 ± 1.06	4.76 ± 1.19	4.80 ± 1.07
随机	5.35 ± 0.81	5.79 ± 0.97	5.62 ± 0.89

表 2： **消融研究**。 不同 GP 长度尺度下的最终基线排名（所有 $\pm\frac{1}{2}$任务和 10 个随机种子上的均值 $\ell_w \in \{0.1, 0.2, 0.3\}$（方程 22）的标准差）。排名按每个任务和随机种子的累积遗憾计算（越低越好），然后在所有任务和随机种子上取平均。粗体表示每个 ℓ_w的平均排名最佳（最低），带下划线的表示次佳。**TALBO 在所有测试的长度尺度下均取得最佳平均排名，表明其能够适应不同的目标漂移速率**。

遗憾的绝对差距较小且难以解读时，这一视角很有帮助，因为它突显了随时间推移及跨任务的一致相对表现。总体而言，三项评估指标均支持相同结论：与强大的潜空间贝叶斯优化基线相比，显式的时间感知能够更好地跟踪漂移的目标函数。作为补充，图 S6 报告了每次迭代的瞬时目标函数值$f_t(\mathbf{x}_t)$，所得定性结论一致。

将性能与时变权重联系起来。 图2 的最后一行展示了时变目标权重$\alpha_{t,\ k}$ 这些权重定义了标量化。这些权重呈现出明显的状态变化（例如，各分量之间的交叉，以及某个分量占主导的时期），从而改变了化学空间中在 f_t 下最优的区域。˜我们观察到，许多转折点和短暂下降 f_t 都与权重转变相吻合：优先级发生变化时，˜先前最优的分子会根据新的权重组合重新评分，并可能暂时变得不再最优，导致 f_t 下降。我们推测，TALBO 的时序感知代理模型和生成模型有助于其在此类变化后更快地重新聚焦，从而维持更高的单位时间最优值。

要超越预训练集有多难？ 为了说明这些时变目标函数的难度，我们直接根据预训练语料库计算参考曲线。在每个时间点 t，我们在 f_t 整个预训练集（包含 127 万个 GuacaMol 分子）上进行评估，并报告数据集最大值（黑色实线）以及高分位数（虚线；前 5%、1%、0.1% 和 0.01%，见图S1）。这些曲线表明，通过穷举搜索预训练集，在每次迭代中可能达到的水平。此外，图S2报告了同一语料库中各分量的参考轨迹，显示对于每个贝叶斯优化步骤中预训练集里的最佳分子，各个目标函数可达到的最佳值也会随权重方案发生明显转变，证实底层景观会随时间显著变化。

重要的是，与该语料库相比，我们的优化预算非常有限：使用 $T = 600$ 和 $B = 10$，我们总共进行了约 6，000 次预言机查询——约占预训练数据集中 127 万个分子的 0。5%。超过前 0.1% 的基线（即优于预训练集中约 1,270 个分子之外的所有分子）是一个特别突出的结果。

在图 S1中，TALBO 在不同任务和时间窗口中经常超过 1% 和 0.1% 的数据集基线，表明它能够在预算有限的情况下找到具有竞争力的分子。此外，对于 相似性 任务，优化轨迹趋近于数据集导出的参考曲线，表明生成式搜索几乎能够找回预训练语料库中最佳的分子。尽管这一现象此前已在静态设置中得到证实 [Maus 等人， 2022]，我们的结果在更具挑战性的漂移目标机制中重现了同样的定性“超越数据集”效应。相反，当“数据集中的最佳值”曲线始终远高于所有方法时，由此产生的差距凸显了在时变设置中可靠地找到稀有且高度匹配的分子的难度。

6.3 消融研究

双重时间建模。为厘清时间感知在哪些方面发挥作用，我们对 TALBO 的两个时间组件进行消融：（i）一种时序感知代理模型，使用时空高斯过程对 $g(\boldsymbol{z},\ t)$ 进行建模；（ii）一种时间感知表示，时间通过协变量进入生成映射（第4.1节）。我们将完整模型与三种变体进行比较：（*a*）无时间（两个组件都不使用 t）；（*b*）仅代理模型（在（$\boldsymbol{z}$，t）上使用时空高斯过程，但表示与时间无关）；以及（*c*）仅表示（表示与时间相关，但在$\boldsymbol{z}$上使用时间无关高斯过程）。图S3 展示了在与主基准测试相同预算下，沿优化轨迹的基线排名（$\ell_w = 0.2$）。总体而言，显式建模时间能够提升性能，而时间感知在表示与代理模型中的相对影响则因任务而异。

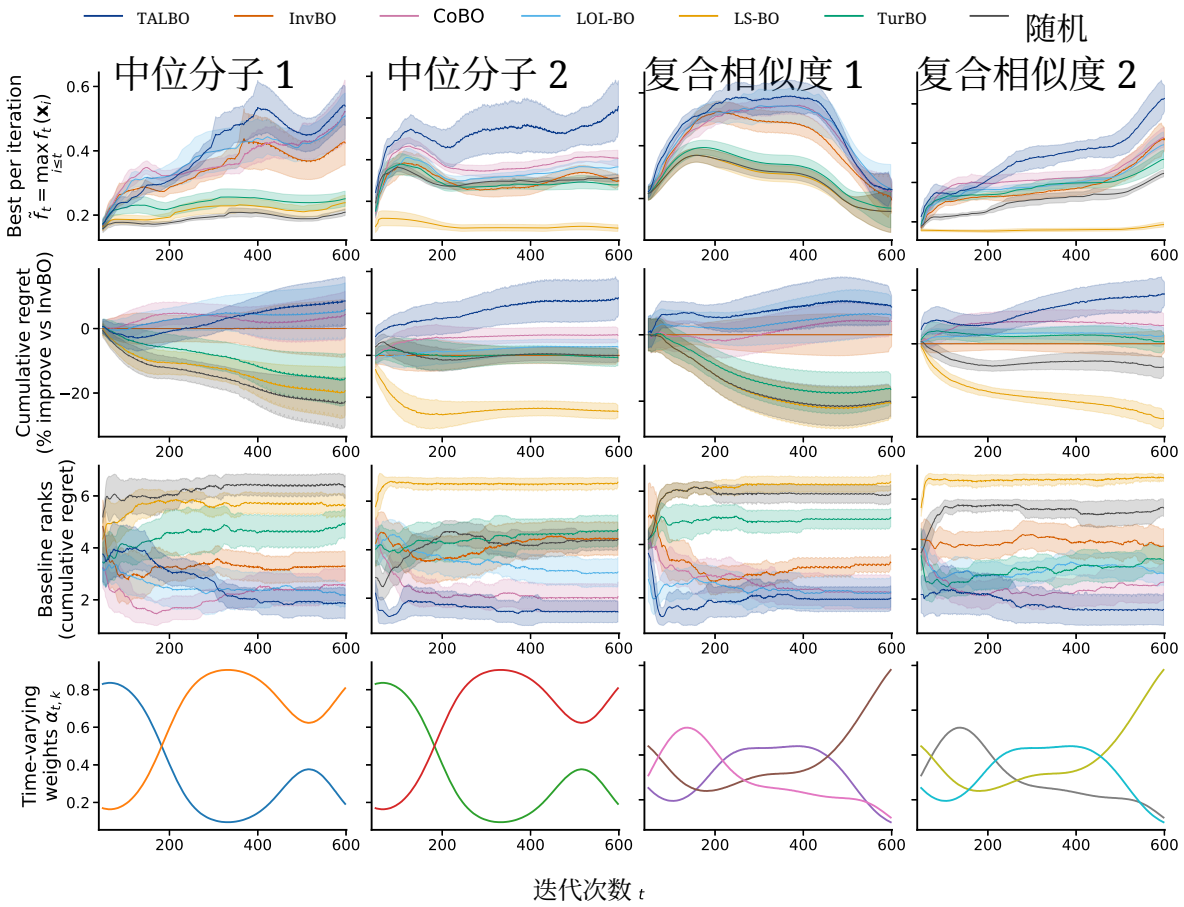

图 2：**各动态多目标任务的性能比较**。第一行：当前目标下的迄今最优值。第二行：累积遗憾，表示为相对于 INVBO 的百分比改进。第三行：基于累积遗憾的每次迭代基线排名（越低越好）。第四行：决定动态多目标权衡的时变目标权重，使用 $\ell_w = 0.2$ 表示 k_w（公式 22）。10 个随机种子上的均值 $\pm\frac{1}{2}$标准差。所有轨迹均使用移动平均进行平滑，以提高可读性。**在所有任务中，TALBO 始终优于其他基线方法，取得更高的迄今最优值、更低的遗憾以及更好的平均排名**。

对优先级漂移速度的敏感性。表2 报告了不同 GP 长度尺度$\ell_w \in \{0.1, 0.2, 0.3\}$下所有任务平均后的最终基线排名。这里，$\ell_w$ 是用于生成公式中的权重轨迹的时间核的（相对）长度尺度22。有效时间相关长度为 $\ell_w T$，其中 $T = 600$ 是优化时域；因此，$\ell_w = 0.1$，0.2、0.3 分别对应约60、120和180 次迭代的特征漂移尺度。较小的 ℓ_w 会引发更快的优先级变化，而较大的 ℓ_w 则会使目标函数变化更平缓、更缓慢。在所有测试条件下，TALBO 的平均排名最低，表明其在优先级快速和渐进转移时均持续表现出色。各任务的完整曲线见图S4，证实了这些趋势具有稳定性。

时不变目标函数。一个自然的担忧是，当目标函数固定时，显式建模时间可能会损害性能。由于通常无法事先知道是否会发生漂移，因此有必要验证该方法在静态设置下不会遭受系统性惩罚。图 S5 报告了时不变目标函数下三个代表性任务的已找到最优值轨迹。在这种情况下，时间感知的优势自然会减弱。TALBO 在部分任务上仍具竞争力，但并未始终优于最强的静态基线方法。差异取决于具体任务，可能反映了底层生成模型的架构差异（例如，DGBFGP 与标准变分自编码器之间的差异）。总体而言，这些结果表明，在不存在漂移时，性能不会出现明显退化。

7 讨论

我们提出了时变潜空间贝叶斯优化，在这种方法中，目标函数会随轮次变化，而搜索则在学习到的表示中进行。我们提出了 TALBO，将时间信息同时纳入代理模型和表示中，并证明了其在动态分子设计任务上的稳健性能；消融实验表明，两个时间组件的作用因任务而异。

局限与展望。 我们的基准测试通过在多属性优化中平滑改变标量化权重来引入漂移。这隔离出一种清晰的偏好漂移形式，但实际实验活动可能会出现更复杂的变化（例如，检测实验更新、约束变化、突发的机制变化，或其他不可重复性的来源）。重要的是，我们的主要跟踪指标 $f_t = \max_{i \le t} f_t(\boldsymbol{x}_i)$ 涉及根据当前目标函数对过去的分子重新评分。因此，f_t 衡量的是 跟踪：即随着偏好漂移，优化器能否维持一组仍具竞争力的候选项。然而，在 f_t 反映新测量值或条件的实验条件下，可能无法对过去的设计重新评分，此时瞬时值 $f_t(\boldsymbol{x}_t)$ 便成为主要的可观测量。值得鼓舞的是，图 S6 表明，按照瞬时视角 $f_t(\boldsymbol{x}_t)$，TALBO 仍具有竞争力。时变潜空间贝叶斯优化中，陈旧数据管理的原则性方法是值得探索的未来方向。动态贝叶斯优化已探索多种遗忘机制，例如折扣、滑动窗口、重置和基于相关性的剪枝，用于处理在较早目标函数切片下收集、且对当前目标函数已不再具有预测力的观测 [Bogunovic 等人，2016年，Bardou 等人， 2024]。将此类策略扩展到潜空间贝叶斯优化尤其值得期待，因为陈旧性不仅可能出现在代理模型数据集中（标签过时），也可能出现在表示层面：随着潜在几何通过重新训练或显式时间条件化而演变，过去潜在观测的语义可能发生变化。最后，我们的时间代理模型采用了简单的平稳时间核；表达能力更强的时间模型（例如非平稳核、变点核或状态空间表述）或许能更好地捕捉突发或多尺度漂移。

参考文献

Alan Nawzad Amin, Nate Gruver, Yilun Kuang, Lily Li, Hunter Elliott, Calvin McCarter, Aniruddh Raghu, Peyton Greenside, 和 Andrew Gordon Wilson。受序列演化生成模型启发的抗体贝叶斯优化。*arXiv* 预印本 *arXiv:2412.07763*，2024。Maximilian Balandat, Brian Karrer, Daniel Jiang, Samuel Daulton, Ben Letham, Andrew G Wilson, 和 Eytan Bakshy。BoTorch：高效蒙特卡洛贝叶斯优化框架。载于神经信息处理系统进展，2020。Mehmet Yi˘git Balık, Maksim Sinelnikov, Priscilla Ong, 和 Harri Lähdesmäki。用于深度生成模型中可扩展高斯过程先验的贝叶斯基函数近似。载于第四十二届国际机器学习会议，2025。Anthony Bardou, Patrick Thiran, 和 Giovanni Ranieri。逝者如斯：在动态贝叶斯优化中移除过时观测。神经信息处理系统进展，2024。Ilija Bogunovic, Jonathan Scarlett, 和 Volkan Cevher。时变高斯过程老虎机优化。载于人工智能与统计，2016。Onur Boyar, Yanheng Gu, Yuji Tanaka, Shunsuke Tonogai, Tomoya Itakura, 和 Ichiro Takeuchi。Crystal-LSBO：利用潜在空间贝叶斯优化自动设计从头设计晶体。神经计算，2025。Nathan Brown, Marco Fiscato, Marwin HS Segler, 和 Alain C Vaucher。GuacaMol：从头分子设计模型的基准测试。《化学信息与建模杂志》，2019。Paul Brunzema, Alexander Von Rohr, 和 Sebastian Trimpe。时变贝叶斯优化在控制器调参中的应用。载于 *2022*年 *IEEE* 第*61*届决策与控制会议，2022。Francesco Paolo Casale, Adrian Dalca, Luca Saglietti, Jennifer Listgarten, 和 Nicolo Fusi。高斯过程先验变分自编码器。神经信息处理系统进展，2018。Jaewon Chu, Jinyoung Park, Seunghun Lee, 和 Hyunwoo J Kim。基于反演的潜空间贝叶斯优化。神经信息处理系统进展，2024。David Eriksson, Michael Pearce, Jacob Gardner, Ryan D Turner, 和 Matthias Poloczek。通过局部贝叶斯优化实现可扩展的全局优化。神经信息处理系统进展，2019。Vincent Fortuin, Dmitry Baranchuk, Gunnar Rätsch, 和 Stephan Mandt。GP-VAE：深度概率时间序列插补。载于人工智能与统计国际会议，2020。Lester Frei, Beichen Gao, Jiami Han, Joseph M Taft, Edward B Irvine, Cédric R Weber, Rachita K Kumar, Benedikt N Eisinger, Andrey Ignatov, Zhouya Yang 等。通过深度突变学习筛选能够抵御 SARS-CoV-2 奥密克戎变异株演化的治疗性抗体。《自然·生物医学工程》，2025。

Roman Garnett。贝叶斯优化。剑桥大学出版社，2023。Rafael Gómez-Bombarelli, Jennifer N Wei, David Duvenaud, José Miguel Hernández-Lobato, Benjamín Sánchez- Lengeling, Dennis Sheberla, Jorge Aguilera-Iparraguirre, Timothy D Hirzel, Ryan P Adams, 和 Alán Aspuru-Guzik。利用数据驱动的分子连续表示实现自动化学设计。*ACS Central Science*，2018。Miguel González-Duque, Richard Michael, Simon Bartels, Yevgen Zainchkovskyy, Søren Hauberg, 和 Wouter Boomsma。离散序列高维贝叶斯优化的综述与基准测试。载于第三十八届神经信息处理系统大会数据集与基准测试专题，2024。Antoine Grosnit, Rasul Tutunov, Alexandre Max Maraval, Ryan-Rhys Griffiths, Alexander I Cowen-Rivers, Lin Yang, Lin Zhu, Wenlong Lyu, Zhitang Chen, Jun Wang 等。利用变分自编码器和深度度量学习进行高维贝叶斯优化。*arXiv* 预印本 *arXiv:2106.03609*，2021。James Hensman, Alexander Matthews, 和 Zoubin Ghahramani。可扩展的变分高斯过程分类。载于人工智能与统计，2015。Diederik P Kingma 和 Max Welling。自编码变分贝叶斯。*arXiv* 预印本 *arXiv:1312.6114*，2013。Seunghun Lee, Jaewon Chu, Sihyeon Kim, Juyeon Ko, 和 Hyunwoo J Kim。通过学习相关潜在空间推进贝叶斯优化。神经信息处理系统进展，2023。Seunghun Lee, Jinyoung Park, Jaewon Chu, Minseo Yoon, 和 Hyunwoo J. Kim。通过自回归归一化流实现潜空间贝叶斯优化。载于第十三届国际学习表征会议，2025。Roman Marchant 和 Fabio Ramos。面向智能环境监测的贝叶斯优化。载于*2012*年*IEEE/RSJ*智能机器人与系统国际会议，2012。Roman Marchant, Fabio Ramos, Scott Sanner 等。用于时空监测的序贯贝叶斯优化。载于*UAI*，2014。Natalie Maus, Haydn Jones, Juston Moore, Matt J Kusner, John Bradshaw, 和 Jacob Gardner。结构化输入上的局部潜在空间贝叶斯优化。神经信息处理系统进展，2022。Henry Moss, Sebastian W. Ober, 和 Tom Diethe。潜空间 COWBOYS 回归：重新思考变分自编码器在结构化空间贝叶斯优化中的应用。载于第*42*届国际机器学习会议论文集，2025。Kevin P Murphy。概率机器学习：高级主题。麻省理工学院出版社，2023。Anita Rácz, Levente M Mihalovits, Maximilian Beckers, Nikolas Fechner, Nikolaus Stiefl, Finton Sirockin, William McCoull, Emma Evertsson, Malin Lemurell, Gergely Makara 等。药物化学优化的变迁格局。《自然综述：药物发现》，2025。Siddharth Ramchandran, Gleb Tikhonov, Kalle Kujanpää, Miika Koskinen, 和 Harri Lähdesmäki。纵向变分自编码器。载于人工智能与统计国际会议，2021。Siddharth Ramchandran, Manuel Haussmann, 和 Harri Lähdesmäki。利用高斯过程先验变分自编码器进行高维贝叶斯优化。载于第十三届国际学习表征会议，2025。C. Rasmussen 和 C. Williams。《机器学习中的高斯过程》。麻省理工学院出版社，2006。Kihyuk Sohn, Honglak Lee, 和 Xinchen Yan。利用深度条件生成模型学习结构化输出表示。神经信息处理系统进展，2015。Arno Solin 和 Simo Särkkä。用于降秩高斯过程回归的希尔伯特空间方法。《统计与计算》，2020。Iiris Sundin, Alexey Voronov, Haoping Xiao, Kostas Papadopoulos, Esben Jannik Bjerrum, Markus Heinonen, Atanas Patronov, Samuel Kaski, 和 Ola Engkvist。人在回路辅助的从头分子设计。《化学信息学杂志》，2022。Michalis Titsias。稀疏高斯过程中诱导变量的变分学习。载于人工智能与统计，2009。Marcelo DT Torres, Yimeng Zeng, Fangping Wan, Natalie Maus, Jacob Gardner, 和 Cesar de la Fuente-Nunez。用于抗生素优化的生成式人工智能方法。*bioRxiv*，2024。Austin Tripp, Erik Daxberger, 和 José Miguel Hernández-Lobato。通过加权再训练，在深度生成模型的潜在空间中实现样本高效优化。神经信息处理系统进展，2020。Austin Tripp, Sergio Bacallado, Sukriti Singh, 和 José Miguel Hernández-Lobato。用于可扩展分子机器学习的Tanimoto随机特征。神经信息处理系统进展，2023。

Mattia Vallerio、Jan Hufkens、Jan Van Impe 和 Filip Logist。用于不确定性下非线性动态过程多目标优化的交互式决策支持系统。《专家系统与应用》，2015。Andrew Gordon Wilson、Zhiting Hu、Ruslan Salakhutdinov 和 Eric P Xing。深度核学习。载于人工智能与统计，2016。Wenjie Xu、Masaki Adachi、Colin N Jones 和 Michael A Osborne。与人类专家协作的有原则贝叶斯优化。载于神经信息处理系统进展，2024。

附录的组织结构如下：

- 附录A 报告正文中提及的补充实验结果，包括：

  – 与预训练数据集基线方法的性能比较（图S1）； – 动态变化的优先级下各组件的数据集最优值（图S2）； – 双重时间建模的消融实验（图S3）； – 不同 GP 长度尺度用于生成漂移权重计划的补充结果（图S4）； – 时不变目标函数的结果（图S5）。– 瞬时值发现指标的结果（图S6）。

- 附录B 介绍了 TALBO 各组件及密切相关机制的实现细节，包括：

  – DGBFGP 的实现细节（附录 B.1）； – CoBO 风格潜在对齐正则化项（附录 B.2）； – InvBO 风格潜变量反演（附录 B.3）； – 采用深度核的 SVGP 代理目标（附录 B.4）； – 潜在空间中的信赖域贝叶斯优化（TuRBO）（附录 B.5）； – 完整的 TALBO 循环（附录 B.6）； – 变分自编码器和 GP 先验变分自编码器的背景介绍（附录 B.7）。

- 附录C 介绍了我们实验设置的实现细节，包括：

  – 全局超参数（附录C.1，表S1）； – DGBFGP 配置和协变量（附录C.2，表S2–S3 和图S7）； – InfoTransformerVAE 配置（附录C.3）； – 计算资源（附录C.4）。

A 补充结果

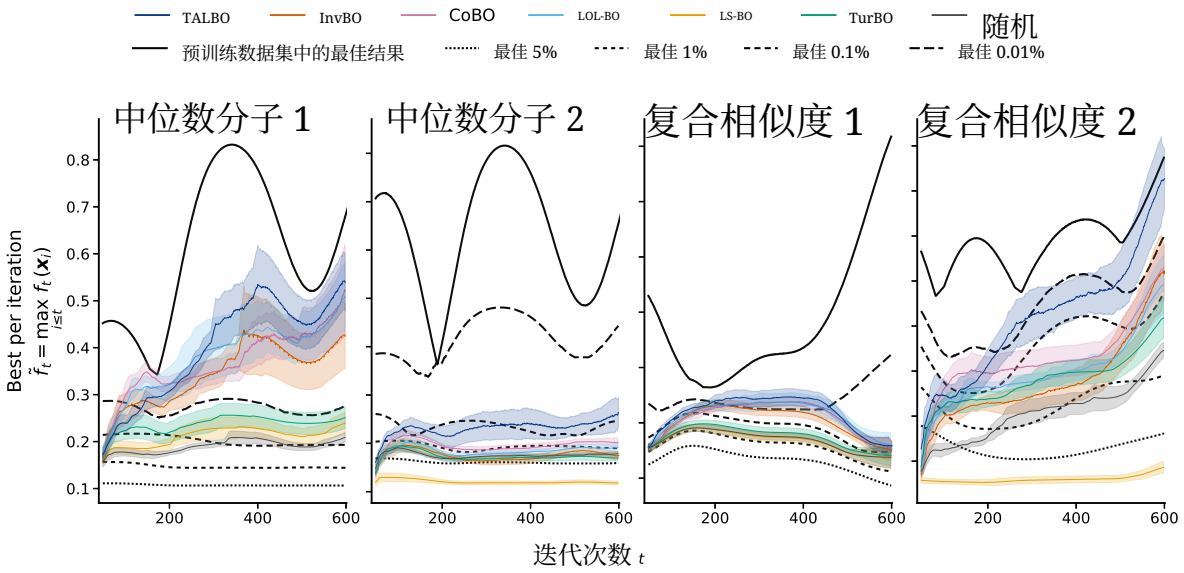

图 S1：**与预训练数据集基线的性能比较**。每个动态多目标任务的当前目标下的迄今最优值，并附加从预训练数据集得出的统计数据。彩色曲线对应优化基线方法（10 个随机种子的均值 $\pm\frac{1}{2}$ 标准差）。黑色实线表示在每个时间步上预训练数据集内可达到的最优值，即在数据集中的所有分子上评估 $\max_j f_t(x_j)$。黑色虚线表示数据集的最高百分位子集（最佳 5%、1%、0.1% 和 0.01%），提供无需自适应优化即可达到的性能参考水平。所有轨迹均采用居中移动平均进行平滑处理，以提高可读性。**该比较突显了仅凭数千次查询超越强劲的预训练集参考水平有多么困难，也展示了动态贝叶斯优化方法仍能在多大程度上超越数据集内最优指标**。

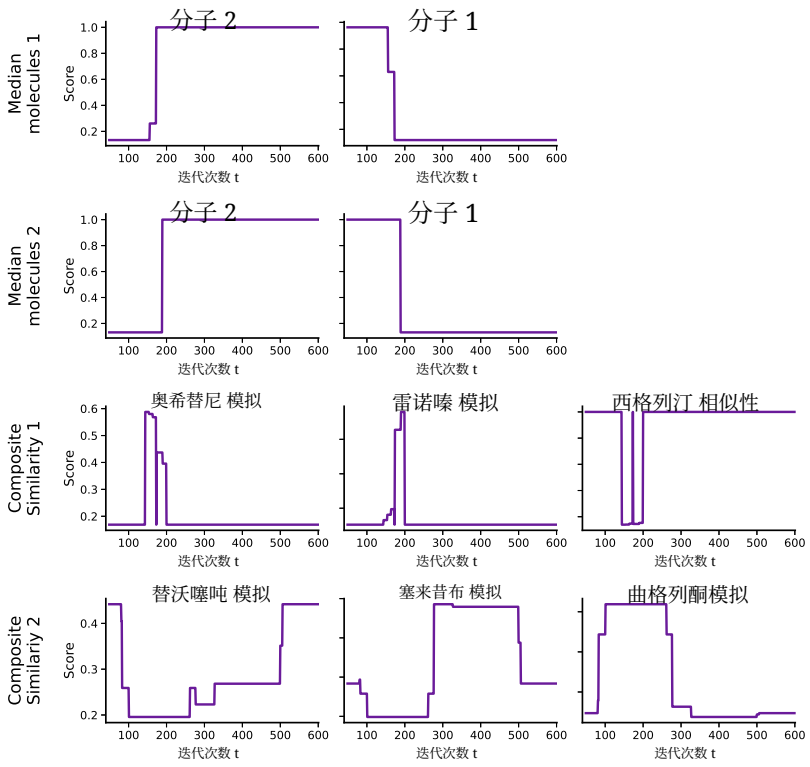

图 S2：**动态变化的优先级下各组件数据集最优值**。对于每项任务，我们使用与基准测试相同的随时间变化的加权方案，在完整的 GuacaMol 预训练语料库（127 万个分子）上评估每个目标分量，并绘制每次迭代中可达到的最佳组件值。**由此得到的轨迹呈现出明显的机制变化和组件间转变，证实权重方案会使底层目标景观随时间发生有意义的变化**。

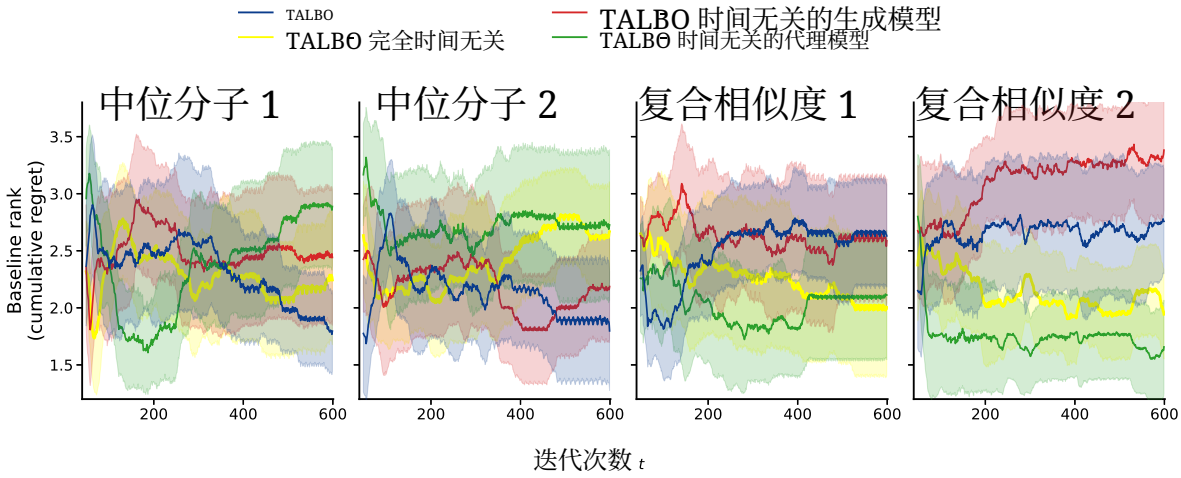

图 S3：**双重时间建模机制的消融实验**。对于每个任务和每次迭代，基线方法在每个随机种子内根据累积遗憾进行排序，然后在 10 个随机种子上取平均。曲线显示均值 $\pm\frac{1}{2}$ 在随机种子上的标准差。该设置类似于图 2：$\ell_w = 0.2$。**显式地对时间进行建模可带来整体性能提升，不过生成模型与代理模型中时间建模的相对贡献因任务而异**。

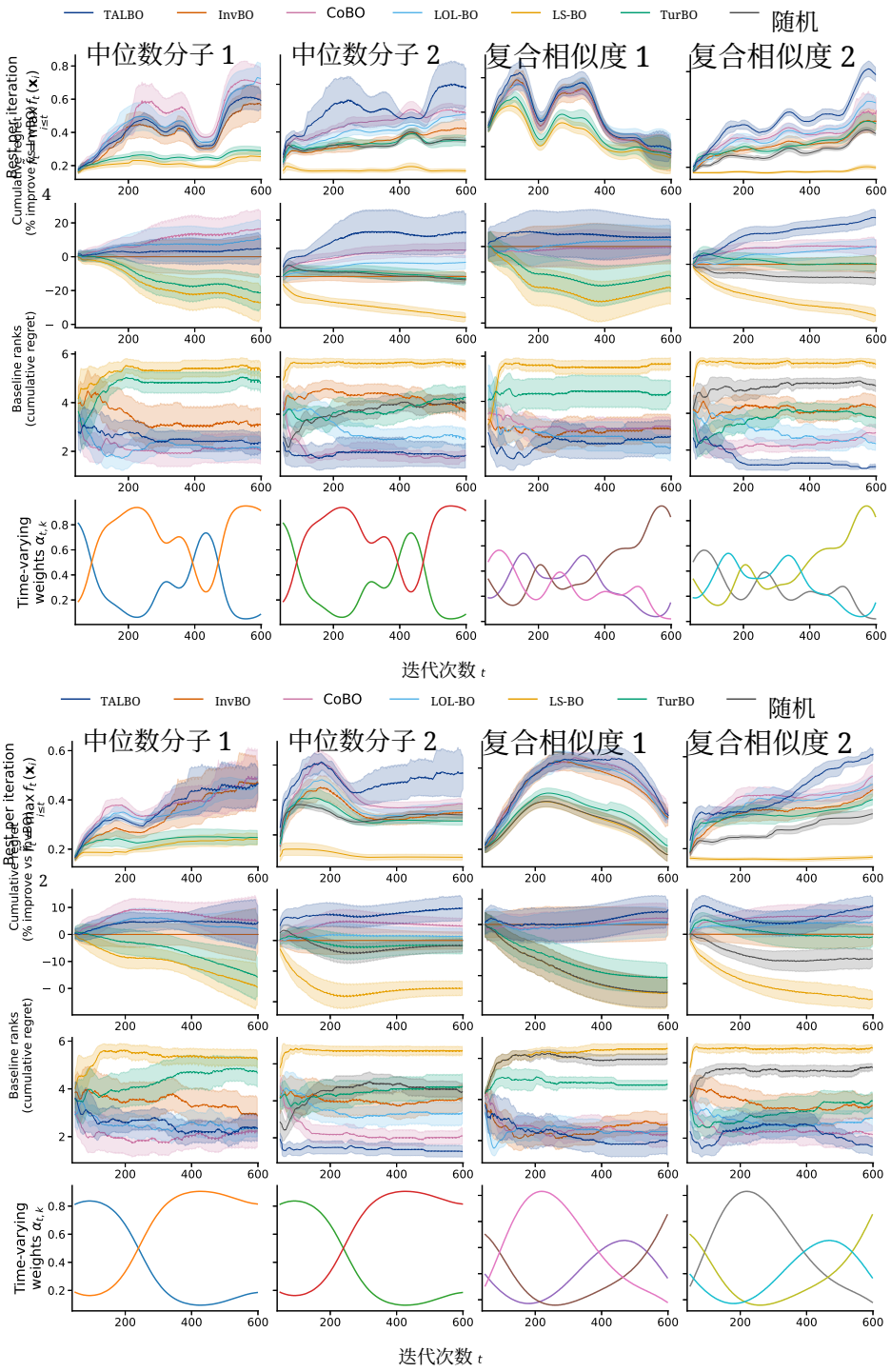

图 S4：**不同 GP 长度尺度下，各动态多目标任务的性能比较** $\ell_w = 0.1$（**上图**）**和** $\ell_w = 0.3$（**下图**）。第一行：当前目标下的迄今最优值。第二行：累积遗憾，以相对于 InvBO 的百分比改进表示。第三行：根据累积遗憾计算的每次迭代基线排名（越低越好）。第四行：决定动态多目标权衡的时变目标权重。10 个随机种子上的均值 $\pm\frac{1}{2}$标准差。所有轨迹均采用居中移动平均进行平滑，以提高可读性。

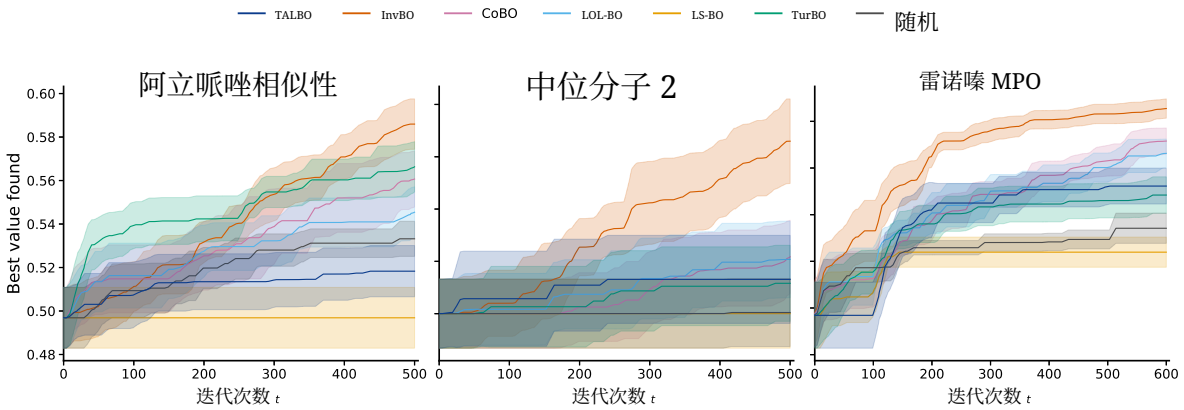

图 S5：**时不变目标**。当目标函数随时间保持固定时，三个代表性任务的最佳已发现值随预言机调用次数的变化。10 个随机种子上的均值 $\pm\frac{1}{2}$ 标准差。在没有时间漂移的情况下，方法之间的差异因任务而异；正如预期，时间感知并未提供系统性优势。为便于阅读，所有轨迹均使用居中移动平均进行平滑。**TALBO 在静态设置中保持竞争力，且未表现出系统性退化**。

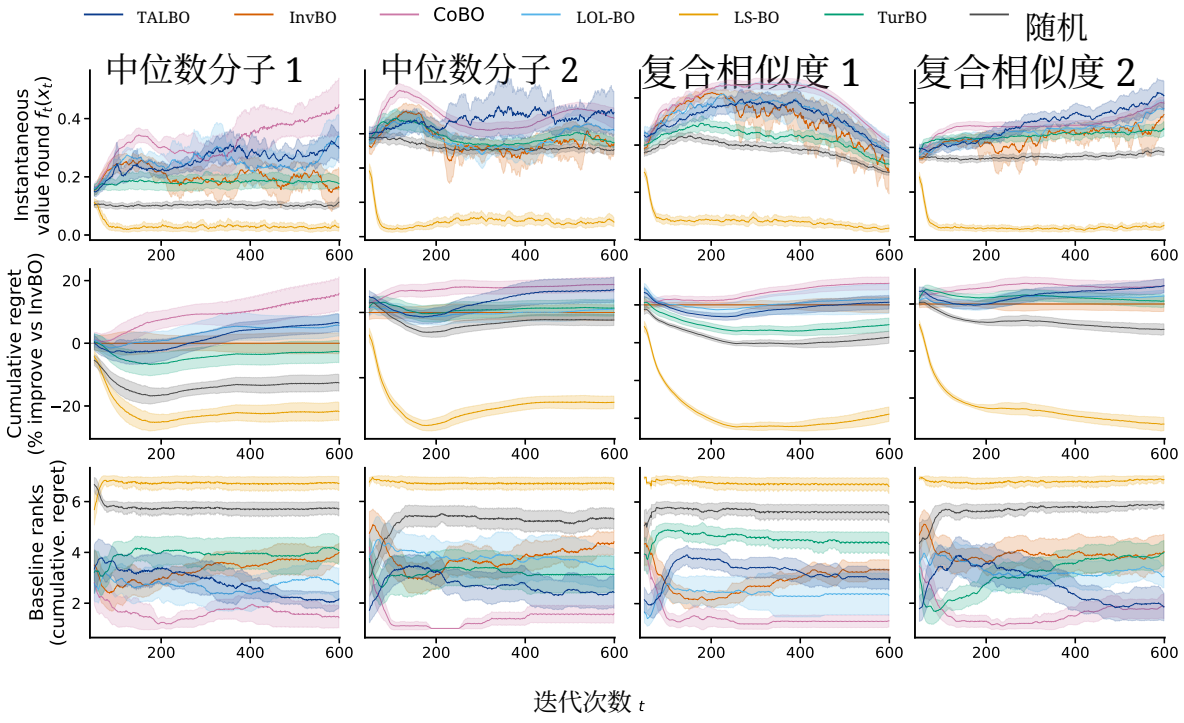

图 S6：**动态多目标任务的性能比较（瞬时评估）**。第一行：每次迭代时评估的瞬时目标值 $f_t(\mathbf{x}_t)$。第二行：累积遗憾，以相对于 INVBO 的百分比改进表示。第三行：根据累积遗憾计算的每次迭代基线排名（越低越好）。10 个随机种子的均值 $\pm\frac{1}{2}$ 标准差。所有轨迹均使用移动平均进行平滑处理，以提升可读性。**TALBO 在各项任务中始终实现有竞争力的瞬时性能和出色的基于遗憾的排名**。

B 基线与组件细节

B.1 深度生成基函数高斯过程（DGBFGP）

本节介绍第 4.1 节中提出的高斯过程先验潜变量构造的实现细节，包括用于实现可扩展高斯过程先验的希尔伯特空间近似，以及实践中采用的解码器似然。我们的介绍遵循 [Balık 等人的工作 2025]。

连续协变量的核函数近似。 对于单变量连续协变量 $c \in \mathbb{R}$，我们对组件 r 使用平方指数（SE）核（为清晰起见，省略了 $c^{(r)}$ 中的索引 r）

$$k_{\mathrm{se}}^{(r)}(c, c' \mid \sigma_r, \ell_r) = \sigma_r^2 \exp\left(-\frac{(c-c')^2}{2\ell_r^2}\right). \tag{S1}$$

根据维纳–辛钦定理，平稳核可通过谱密度 $s(\omega)$ 表示，

$$\begin{aligned} s(\omega) &= \int_{-\infty}^{\infty} k(u)\, e^{-i\omega u}\, du, \\ k(u) &= \frac{1}{2\pi}\int_{-\infty}^{\infty} s(\omega)\, e^{i\omega u}\, d\omega, \end{aligned} \tag{S2}$$

对于 SE 核，其谱密度为

$$s_{\mathrm{se}}(\omega \mid \sigma_r, \ell_r) = \sigma_r^2 \ell_r \sqrt{2\pi}\, \exp\left(-\tfrac{1}{2}\ell_r^2 \omega^2\right). \tag{S3}$$

在有界域 Ω 上采用希尔伯特空间（特征函数）近似 $= [-J, J]$，并施加狄利克雷边界条件，

$$k_{\mathrm{se}}^{(r)}(c, c') \approx \tilde{k}^{(r)}(c, c') = \sum_{m=1}^{M} s_{\mathrm{se}}(\sqrt{\lambda_m} \mid \sigma_r, \ell_r)\, \phi_m^{(r)}(c)\, \phi_m^{(r)}(c'), \tag{S4}$$

其中，拉普拉斯算子特征对为

$$\begin{aligned} \phi_m(c) &= \frac{1}{\sqrt{J}} \sin\left(\frac{\pi m (c+J)}{2J}\right), \\ \lambda_m &= \left(\frac{\pi m}{2J}\right)^2. \end{aligned} \tag{S5}$$

注意，为简洁起见，我们省略了 $\phi_m(c)$ 和 λ_m 中的索引 r，因为它们仅通过第 r 个协变量分量的域 $\Omega = [-J, J]$ 的大小依赖于 r。定义 $\boldsymbol{\phi}^{(r)}(c) = [\phi_1(c) \ldots, \phi_M(c)]^\top$，潜在维度 l 的高斯过程样本可写成如下线性形式：

$$h^{(r,l)}(c) \approx \boldsymbol{a}^{(r,l)\top} \boldsymbol{\phi}^{(r)}(c), \qquad \boldsymbol{a}^{(r,l)} \sim \mathcal{N}\big(\mathbf{0}, \boldsymbol{S}^{(r)}(\sigma_r, \ell_r)\big), \tag{S6}$$

其协方差为对角形式

$$\boldsymbol{S}^{(r)}(\sigma_r, \ell_r) = \mathrm{diag}\big(s_{\mathrm{se}}(\sqrt{\lambda_1} \mid \sigma_r, \ell_r), \ldots, s_{\mathrm{se}}(\sqrt{\lambda_M} \mid \sigma_r, \ell_r)\big). \tag{S7}$$

在我们的实现中，使用希尔伯特特征，其中 $J = 2.55$ 和 $M = 128$。

解码器似然（类别标记）。 对于结构化离散设计（例如 SELFIES 字符串），解码器 $p_\theta(\boldsymbol{x} \mid \boldsymbol{z})$ 按标记自回归分解。给定标记序列 $\boldsymbol{x} = (x_1, \ldots, x_L)$ 和前缀 $x_{<\ell}$，解码器定义了每一步的类别概率

$$\pi_\ell(\boldsymbol{z}, x_{<\ell}) := p(x_\ell \mid \boldsymbol{z}, x_{<\ell}) = \mathrm{softmax}\big(\mathrm{logits}_\ell(\boldsymbol{z}, x_{<\ell})\big), \qquad \ell = 1, \ldots, L. \tag{S8}$$

公式中的重构项20采用经长度归一化的解码器对数似然来实现

$$\log p_\theta(\boldsymbol{x} \mid \boldsymbol{z}) = \frac{1}{L}\sum_{\ell=1}^{L} \log \pi_\ell(\boldsymbol{z}, x_{<\ell})_{x_\ell}. \tag{S9}$$

B.2 CoBO 风格的潜在空间对齐正则化项

在贝叶斯优化期间微调生成式表征时（算法 1），仅优化重构并不能保证潜在空间中的距离能够反映黑箱目标函数的变化。这种“几何结构—目标函数”不匹配是潜空间贝叶斯优化中一种已知的失效模式，而 CoBO [Lee 等人， 2023] 提出了简单的正则化项，以促进目标函数关于潜在编码的局部平滑性，并避免潜在尺度退化。TALBO 采用了相同的正则化项和加权方案，下面将对此作简要说明。

基于目标函数的重要性权重。 为了将正则化重点放在有前景的设计上，我们为每个样本分配一个软重要性权重 $w_i \in (0,1]$，该权重由其目标值导出。遵循 Lee 等人 [2023], 的做法，令 y_q 为当前小批量中目标值的 q 分位数（且 $\sigma_w > 0$ 为平滑参数）。定义

$$w_i := \lambda(y_i) = \Pr(Y > y_q), \qquad Y \sim \mathcal{N}(y_i, \sigma_w^2), \tag{S10}$$

等价地， $w_i = 1 - \Phi\big((y_q - y_i)/\sigma_w\big)$ 其中 Φ 为标准高斯累积分布函数。

加权 Lipschitz 正则项。 CoBO 通过鼓励潜在空间中的复合目标函数表现为 Lipschitz 函数来实现平滑性。具体而言，对于一个小批量 $\{(\boldsymbol{z}_i,\ y_i,\ w_i)\}_{i=1}^N$，定义成对斜率

$$s_{ij} := \frac{|y_i - y_j|}{\|\boldsymbol{z}_i - \boldsymbol{z}_j\|_2}, \qquad L := \operatorname{median}\big(\{s_{ij}\}_{i,j}\big), \tag{S11}$$

并且只惩罚超过典型（中位数）值的斜率：

$$\mathcal{L}_{\text{Lip}}^{(w)} := \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} \sqrt{w_i w_j}\ \max\big(0,\ s_{ij} - L\big). \tag{S12}$$

几何平均加权 $\sqrt{w_i w_j}$ 强调高值区域的平滑性，这正是 CoBO 提出的方法。

潜在空间尺度正则项。 需要注意的是，仅使用 $\mathcal{L}^{(w)}$ Lip 会产生退化解（例如，增大所有成对距离 $\|\boldsymbol{z}_i - \boldsymbol{z}_j\|_2$ 会使 s_{ij}缩小）。因此，CoBO 会约束潜在空间的平均 尺度。令

$$\bar{d}_{\boldsymbol{z}} := \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} \|\boldsymbol{z}_i - \boldsymbol{z}_j\|_2, \qquad c_d := \mathbb{E}\big[\|U - V\|_2\big] = 2\,\frac{\Gamma\big(\frac{d+1}{2}\big)}{\Gamma\big(\frac{d}{2}\big)}, \tag{S13}$$

其中 $U, V \sim \mathcal{N}(\mathbf{0}, I_d)$，且 d 为潜在维度。然后我们定义

$$\mathcal{L}_Z := \big|\bar{d}_z - c_d\big|. \tag{S14}$$

这使 潜在几何锚定在标准正态先验所隐含的尺度上，从而避免平凡的重新缩放 s.

表征更新期间的完整目标函数。 我们在贝叶斯优化步骤中通过将这些对齐项加入 DGBFGP 目标函数来更新生成模型；（与代理模型联合训练时）还会加入代理损失 $\mathcal{L}_{\text{SVGP}}$ （附录 B.4）：

$$\mathcal{L}_{\text{TALBO}} = -\mathcal{L}_{\text{DGBFGP}} + \mathcal{L}_{\text{Lip}}^{(w)} + \mathcal{L}_Z + \mathcal{L}_{\text{SVGP}}. \tag{S15}$$

B.3 InvBO 风格的潜空间反演

潜空间贝叶斯优化中一种反复出现的失效模式，是用于训练代理模型的潜在编码与预言机实际评估的离散设计不对齐。具体而言，如果与已评估设计 $\boldsymbol{x}$ 关联的潜在编码 $\boldsymbol{z}$ 在当前解码器下无法解码回 $\boldsymbol{x}$ （即 $\hat{\boldsymbol{x}} = \text{Dec}_\theta(\boldsymbol{z}) \neq \boldsymbol{x}$），那么训练样本对 $(\boldsymbol{z}, y)$（其中 $y = f_t(\boldsymbol{x})$）便不再对应代理模型目标 $f_t(\text{Dec}_\theta(\boldsymbol{z}))$。表示或解码器更新后，这种不匹配可能会加剧。遵循 InvBO [Chu 等人 2024], 的即插即用反演模块，我们可以选择通过显式反演解码器来优化潜在编码，以找回能够重构已评估设计的编码，而无需额外的预言机调用。

给定一个设计 $\boldsymbol{x} \in \mathcal{X}$，我们求解

$$\boldsymbol{z}^{\text{inv}}(\boldsymbol{x}) \in \arg\min_{\boldsymbol{z} \in \mathbb{R}^d}\ \mathcal{L}_{\text{inv}}(\boldsymbol{z}; \boldsymbol{x}), \tag{S16}$$

$$\mathcal{L}_{\text{inv}}(\boldsymbol{z}; \boldsymbol{x}) := -\log p_\theta(\boldsymbol{x} \mid \boldsymbol{z}), \tag{S17}$$

其中 $-\log p_\theta(\boldsymbol{x} \mid \boldsymbol{z})$ 用于衡量解码器重构 $\boldsymbol{x}$ 自 $\boldsymbol{z}$的效果。在我们的 SELFIES 设置中， $\mathcal{L}_{\text{inv}}(\boldsymbol{z}; \boldsymbol{x})$ 通过解码器 [Maus 等人提出的（长度归一化）词元负对数似然来实现， 2022]。
我们从当前的编码器/嵌入输出进行初始化（例如， $\boldsymbol{z}^{(0)} \sim q_\phi(\boldsymbol{z} \mid \boldsymbol{x})$，或 $\boldsymbol{z}^{(0)} = \mathbb{E}_q[\boldsymbol{z}(\mathbf{c}(\boldsymbol{x}, t); \boldsymbol{A})]$），并仅更新 $\boldsymbol{z}$，同时保持解码器参数固定：

$$\boldsymbol{z}^{(k+1)} \leftarrow \boldsymbol{z}^{(k)} - \eta \nabla_{\boldsymbol{z}} \mathcal{L}_{\text{inv}}(\boldsymbol{z}^{(k)}; \boldsymbol{x}), \qquad k = 0, \ldots, T_{\text{inv}} - 1. \tag{S18}$$

与 InvBO 中一样，一旦解码分子 $\hat{\boldsymbol{x}} = \Gamma(\boldsymbol{z}^{(k+1)})$ 在输入空间距离 $d_{\mathcal{X}}(\cdot, \cdot)$ 下与 $\boldsymbol{x}$ 足够接近（例如，SELFIES 字符串上的归一化莱文斯坦距离），我们便可提前停止，

$$d_{\mathcal{X}}(\boldsymbol{x}, \hat{\boldsymbol{x}}) \leq \epsilon. \tag{S19}$$

精炼潜变量 $\boldsymbol{z}^{\text{inv}}(\boldsymbol{x})$ 随后可以替换新添加点所存储的潜在编码和/或当前的 top-k 集合，从而在解码器更新下为代理学习维持一个对齐的 数据集。

B.4 带深度核函数的 SVGP 代理模型目标函数

在迭代 t时，代理模型对潜在目标进行建模

$$g(\boldsymbol{z}, t) \;:=\; f_t(\Gamma(\boldsymbol{z})),$$

使用数据集 $\mathcal{D}_t = \{(\boldsymbol{z}_i, t_i, y_i)\}_{i=1}^{n_t}$。我们对 g 施加高斯过程先验，

$$g \sim \mathcal{GP}\big(0,\, k((\boldsymbol{z}, t), (\boldsymbol{z}', t'))\big), \qquad k((\boldsymbol{z}, t), (\boldsymbol{z}', t')) = k_{\boldsymbol{z}}(\boldsymbol{z}, \boldsymbol{z}')\, k_t(t, t'),$$

并使用高斯似然

$$p(y_i \mid g(\boldsymbol{z}_i, t_i)) = \mathcal{N}\big(y_i \mid g(\boldsymbol{z}_i, t_i), \sigma_y^2\big).$$

为了扩展到超出 $n_t \gg 10^3$，我们使用稀疏变分高斯过程（SVGP） [Hensman 等人， 2015]以及诱导输入$\mathbf{U} = \{u_m\}_{m=1}^{M}$ 位于联合输入空间（例如 $u_m = (\boldsymbol{z}_m,\ t_m)$），以及变分分布

$$q(\mathbf{g_U}) = \mathcal{N}(\mathbf{m}, \mathbf{S}),$$

其中$\mathbf{g_U} := \big(g(u_1)\text{.....},\ g(u_M)\big)$。令 $p(\mathbf{g_U}) = \mathcal{N}(\mathbf{0},\ \mathbf{K_{UU}})$表示诱导输入处的高斯过程先验。SVGP证据下界为

$$\mathcal{L}_{\text{SVGP}} = \sum_{i=1}^{n_t} \mathbb{E}_{q(g(\boldsymbol{z}_i, t_i))}\big[\log p(y_i \mid g(\boldsymbol{z}_i, t_i))\big] - \mathrm{KL}\big(q(\mathbf{g_U}) \,\|\, p(\mathbf{g_U})\big), \tag{S20}$$

并相对于变分参数（$\mathbf{m}$，$\mathbf{S}$）和核超参数进行优化（通常对 i采用小批量）。

深度核学习。当使用深度核 [Wilson 等人， 2016], 我们将 $\boldsymbol{z}$ 替换为学习得到的特征映射 $\phi_\psi(\boldsymbol{z})$并定义

$$k_{\boldsymbol{z}}(\boldsymbol{z}, \boldsymbol{z}') \;:=\; k_{\text{RBF}}\big(\phi_\psi(\boldsymbol{z}), \phi_\psi(\boldsymbol{z}')\big),$$

从而使代理模型i nput 变为 $(\phi_\psi(\boldsymbol{z})\ t)$。随后，我们将 ψ 与公式 S20中的 SVGP 目标函数联合优化 .

B.5 信赖域贝叶斯优化配置

TurBO-M [Eriksson 等人， 2019] 是高维贝叶斯优化中广泛采用的一种策略，其通过维护M 个独立的局部搜索，每个搜索都限制在各自的超矩形信赖域内。研究表明，TurBO-1 在大多数设置中优于其他变体，因此随后被 Maus 等人 [2022]用于潜空间贝叶斯优化。尽管潜空间贝叶斯优化会将原始输入空间 $\mathcal{X}$ 映射到压缩潜在空间 $\mathcal{Z}$，但由此形成的搜索空间仍然是高维的，这给传统贝叶斯优化方法带来了挑战。TurBO 通过以当前最优解为中心开展优化，并围绕其建立信赖域来提高搜索效率，如下所述：
一个以 $\boldsymbol{\kappa} \in \mathbb{R}^d$ 为中心、长度为 $L > 0$ 且权重为 $\boldsymbol{\omega} \in \mathbb{R}^d$ 的信赖域是

$$\mathcal{T}(\boldsymbol{\kappa}; L, \boldsymbol{\omega}) = \big\{\boldsymbol{z} \in \mathbb{R}^d : \boldsymbol{\kappa} - \tfrac{1}{2}\boldsymbol{\omega} L \leq \boldsymbol{z} \leq \boldsymbol{\kappa} + \tfrac{1}{2}\boldsymbol{\omega} L\big\}. \tag{S21}$$

新批次的状态更新 Y_{next} （截至目前的最优值为 $y^\star$）：

$$\text{Success if} \quad \max(Y_{\text{next}}) > y^\star + 10^{-3}|y^\star|, \quad \text{failure otherwise},$$
$$y^\star \leftarrow \max\Big(y^\star,\ \max(Y_{\text{next}})\Big). \tag{S22}$$

令 n_{succ} 和 n_{fail} 分别表示成功计数器和失败计数器。

成功时：

$$n_{\text{succ}} \leftarrow n_{\text{succ}} + 1, \qquad n_{\text{fail}} \leftarrow 0.$$

失败时：

$$n_{\text{succ}} \leftarrow 0, \qquad n_{\text{fail}} \leftarrow n_{\text{fail}} + 1.$$

如果 n_{succ} 达到容差 N_{succ}，则扩大 $L \leftarrow \min(2L, L_{\max})$；如果 n_{fail} 达到容差 N_{fail}，则缩小 $L \leftarrow \frac{L}{2}$。

如果 $L < L_{\min}$，则触发重启。

B.6 综合 TALBO 循环

算法2 TALBO，采用 CoBO 式对齐和 InvBO 式反演的完整版本

要求： 失败容差 N_{fail}；改进阈值 δ；初始评估集 $\mathcal{D}_0 = \{(\boldsymbol{x}_i, t_i, y_i)\}_{i=1}^{n_0}$

.1：预训练 DGBFGP，以获得 $q_0(\boldsymbol{A}, \sigma, \ell)$ 和解码器参数 θ_0。2： $\mathcal{D}_0^{\text{rich}} \leftarrow \emptyset$。**3：对于每个 $(\boldsymbol{x}_i, t_i, y_i) \in \mathcal{D}_0$ 执行 4：** $\boldsymbol{z}_i \leftarrow$ **嵌入并对齐$(\boldsymbol{x}_i, t_i;\ q_0, \theta_0)$ 5：** $\mathcal{D}_0^{\text{rich}} \leftarrow \mathcal{D}_0^{\text{rich}} \cup \{(\boldsymbol{x}_i, \boldsymbol{z}_i, t_i, y_i)\}$**6：结束循环 7：** $y^\star \leftarrow$ **最大值$\{y : (\boldsymbol{x}, \boldsymbol{z}, t, y) \in \mathcal{D}_0^{\text{rich}}\}$；** $n_{\text{fail}} \leftarrow 0$。**8：从 $t = 1$ 到 T，或直到满足判据为止，执行 9：在 $\mathcal{D}_{t-1}^{(z)}$ 上拟合时空代理模型 $g_t := \{(\boldsymbol{z}, t, y) : (\boldsymbol{x}, \boldsymbol{z}, t, y) \in \mathcal{D}_{t-1}^{\text{rich}}\}$（公式 S20）10：选择 $\hat{\boldsymbol{z}}_t \in \arg\max_{\boldsymbol{z} \in \mathcal{T}_t} \alpha\big(\boldsymbol{z};\ p(g_t(\cdot, t) \mid \mathcal{D}_{t-1}^{(z)})\big)$，其中 $\mathcal{T}_t$ 是以潜在空间中的当前最优解为中心的当前 TuRBO 信赖域（附录 B.5）。11：解码并评估 $\boldsymbol{x}_t \sim p_{\theta_{t-1}}(\cdot \mid \hat{\boldsymbol{z}}_t)$。12：** $y_t = f_t(\boldsymbol{x}_t) + \varepsilon_t$。**13：** $\boldsymbol{z}_t \leftarrow$ **嵌入并对齐$(\boldsymbol{x}_t, t;\ q_{t-1}, \theta_{t-1})$ 14：** $\mathcal{D}_t^{\text{rich}} \leftarrow \mathcal{D}_{t-1}^{\text{rich}} \cup \{(\boldsymbol{x}_t, \boldsymbol{z}_t, t, y_t)\}$**15：如果 $y_t > y^\star + \delta\,|y^\star|$，则 16：** $y^\star \leftarrow y_t$；$n_{\text{fail}} \leftarrow 0$。**17：否则 18：** $n_{\text{fail}} \leftarrow n_{\text{fail}} + 1$。**19：结束条件判断 20：如果 $n_{\text{fail}} \geq N_{\text{fail}}$，则 21：通过优化公式 S15，将 DGBFGP 后验分布更新为 $q_t(\boldsymbol{A}, \sigma, \ell)$，并将解码器更新为 θ_t。22：** $\mathcal{D}_t^{\text{rich}}{}_{,\text{new}} \leftarrow \emptyset$。**23：对于每个 $(\boldsymbol{x}_i, \boldsymbol{z}_i, t_i, y_i) \in \mathcal{D}_t^{\text{rich}}$执行 24：** $\boldsymbol{z}_i \leftarrow$ **嵌入并对齐$(\boldsymbol{x}_i, t_i;\ q_t, \theta_t)$**25 $\mathcal{D}_{t\ \text{new}}^{\text{rich}} \leftarrow \mathcal{D}_{t\ \text{new}}^{\text{rich}} \cup \{\ \boldsymbol{x}_i\ \ \boldsymbol{z}_i\ \ t_i\ \ y_i\ \}$**：,,(,,,)：结束循环**

27： $\mathcal{D}_t^{\text{rich}} \leftarrow \mathcal{D}_t^{\text{rich}}{}_{,\text{new}}$；$n_{\text{fail}} \leftarrow 0$。**28：否则 29：** $q_t \leftarrow q_{t-1}$；$\theta_t \leftarrow \theta_{t-1}$。**30：结束条件判断 31：结束循环 32：过程 嵌入并对齐$(\boldsymbol{x}, t;\ q, \theta)$：33：** $\mathbf{c} \leftarrow \mathbf{c}(\boldsymbol{x}, t)$ **// 协变量可能包括预先计算的描述符或元数据 34：** $\boldsymbol{z} \leftarrow \mathbb{E}_q[\boldsymbol{z}(\mathbf{c}; \boldsymbol{A})]$**35：// 反演步骤：优化 $\boldsymbol{z}$，使其解码后还原为 $\boldsymbol{x}$（第 B.3 节）36：** $\boldsymbol{z}^{(0)} \leftarrow \boldsymbol{z}$**37：从 $k = 1$ 到 T_{inv} 执行 38：** $\boldsymbol{z}^{(k)} \leftarrow \boldsymbol{z}^{(k-1)} - \eta \nabla_{\boldsymbol{z}} \mathcal{L}_{\text{inv}}(\boldsymbol{z}^{(k-1)}; \boldsymbol{x})$ **39：如果 $d_{\mathcal{X}}(\boldsymbol{x}, \Gamma(\boldsymbol{z}^{(k)})) \leq \epsilon$，则 40：中断 41：结束条件判断 42：结束循环**

43: **返回** $\boldsymbol{z}^{(k)}$

B.7 变分自编码器与 GP 先验变分自编码器的背景

动机。 潜在空间贝叶斯优化通常依赖预训练生成模型，将结构化设计映射 $\boldsymbol{x} \in \mathcal{X}$ 到连续潜在空间 $\boldsymbol{z} \in \mathbb{R}^d$，并将候选潜变量解码回设计。最常见的基础模型是变分自编码器 [Kingma 和 Welling, 2013], 其潜变量先验有助于形成良好的嵌入。我们的方法在理念上与之密切相关，但在先验方面有所不同：我们不采用简单的独立先验（例如 $\mathcal{N}(\mathbf{0}, \mathbf{I})$），而是为潜在表示引入协变量依赖的高斯过程先验，从而获得时间连贯的潜在几何。下面简要回顾变分自编码器、条件变分自编码器和高斯过程先验变分自编码器，作为背景介绍。

变分自　　编码器。变分自编码器是观测 $\boldsymbol{x} \in \mathcal{X}$ 的潜变量生成模型，其形式为　　m

$$p_\omega(\boldsymbol{x}, \boldsymbol{z}) = p_\theta(\boldsymbol{x} \mid \boldsymbol{z})\, p(\boldsymbol{z}), \qquad \omega = \{\theta\}, \tag{S23}$$

其中 $\boldsymbol{z} \in \mathbb{R}^d$ 是未观测的潜在编码，且 $p(\boldsymbol{z})$ 通常是简单先验，例如 $\mathcal{N}(\mathbf{0}, \mathbf{I})$。后验分布 $p_\omega(\boldsymbol{z} \mid \boldsymbol{x})$通常难以处理，因此变分自编码器借助编码器采用摊销变分推断 $q_\phi(\boldsymbol{z} \mid \boldsymbol{x})$ 来近似它 [Kingma 和 Welling, 2013, Murphy, 2023]。学习通过最大化证据下界（ELBO）进行:

$$\log p_\omega(\boldsymbol{x}) \geq \mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x} \mid \boldsymbol{z})] - \mathrm{KL}(q_\phi(\boldsymbol{z} \mid \boldsymbol{x}) \,\|\, p(\boldsymbol{z}))\,, \tag{S24}$$

关于编码器参数与解码器参数（ϕ, θ）。

条件变分自编码器。 条件变分自编码器（cVAE）通过同时对编码器和解码器施加条件，纳入辅助协变量$\mathbf{c} \in \mathbb{R}^R$，

$$p_\omega(\boldsymbol{x}, \boldsymbol{z} \mid \mathbf{c}) = p_\theta(\boldsymbol{x} \mid \boldsymbol{z}, \mathbf{c})\, p(\boldsymbol{z} \mid \mathbf{c}), \qquad q_\phi(\boldsymbol{z} \mid \boldsymbol{x}, \mathbf{c}), \tag{S25}$$

从而得到形式相同的证据下界，其中 $\mathbf{c}$ 被视为观测输入 [Sohn 等人, 2015]。在我们的设定中，协变量记为 $\mathbf{c}(\boldsymbol{x}, t)$ 也可以包括时间。生成模型的另一种参数化方式假设模型仅依赖先验中的协变量 $\mathbf{c}$，而不依赖解码器中的协变量，因此得到 $p_\omega(\boldsymbol{x}, \boldsymbol{z} \mid \mathbf{c}) = p_\theta(\boldsymbol{x} \mid \boldsymbol{z})\, p(\boldsymbol{z} \mid \mathbf{c})$。

高斯过程先验变分自编码器。 标准变分自编码器（以及条件变分自编码器）在潜空间贝叶斯优化中的一个局限是：潜在空间的学习主要是为了在无监督目标下重构 $\boldsymbol{x}$，而由此形成的几何结构可能与下游优化目标并不十分匹配。高斯过程先验变分自编码器用高斯过程先验取代简单的潜变量先验，通过共享协变量将不同样本的潜在编码关联起来。具体而言，令 $\mathbf{c}_n$ 表示 $\boldsymbol{x}_n$ 的协变量，高斯过程先验变分自编码器为每个潜在维度 $j = 1, \ldots, d$设定潜函数 $h^{(j)}(\cdot)$,

$$h^{(j)}(\cdot) \sim \mathcal{GP}(0, k(\cdot, \cdot)), \qquad \boldsymbol{z}_n^{(j)} := h^{(j)}(\mathbf{c}_n), \tag{S26}$$

因此，集合 $\{\boldsymbol{z}_n\}_{n=1}^N$ 在先验上不再相互独立，而是随 $\mathbf{c}$平滑变化。这会形成协变量感知的潜在几何（例如，当 $\mathbf{c}$包含时间时，就会体现为纵向结构）；人们也探索过通过在高斯过程先验中纳入辅助变量，来提升潜空间中的贝叶斯优化 [Ramchandran 等人, 2025]。我们的 TALBO 模型以 DGBFGP 作为生成模型，可视为这一思路的一种可扩展、随时间变化的变体：它采用带有全局变分参数的基函数近似，使基于高斯过程先验的潜在表示能够实际应用于大规模场景。

C 实验设置与实现细节

C.1 全局超参数

优化设置	
优化时域 T	600
初始数据集大小 N_{init}	100
批次大小 B	10
随机种子数量	10
时变目标生成	
时间核（方程 22）	平方指数 k_{w}
相对长度尺度 ℓ_w	0.2（除非另有说明）
有效漂移尺度	$\ell_w T$
温度 β (方程23)	1.0
噪声 σ^2	0.0
贝叶斯优化配置	
采集函数	q-批次汤普森采样
$\boldsymbol{z}$ 上的代理核	深度核径向基函数

表 S1：实验中使用的主要优化、目标函数生成及建模设置。

C.2 DGBFGP 配置

	DGBFGP
潜在维度	$d = 256$
最大令牌长度	$L_{\max} = 128$
数据集	GuacaMol
基 / 特征	希尔伯特空间特征（基_func=hs），基大小 $M = 128$
GP KL 缩放因子	gpB=0.001（gp_kl_scale_B）
超参数	批次大小 1024，学习率 5×10^{-4}
解码器	InfoTransformerVAE 解码器
丢弃	$p_{\text{drop}} = 0.2$

表 S2：实验中使用的 DGBFGP 配置和训练超参数。

索引	协变量名称
1	分子量
2	氢键供体数
3	氢键受体数
4	可旋转键数
5	药物相似性的定量估计（QED）
6	合成可及性评分（SAS）
7	长环数量
8	时间协变量 t（时间版本可选）

表 S3：TALBO 用于多属性优化任务的协变量。

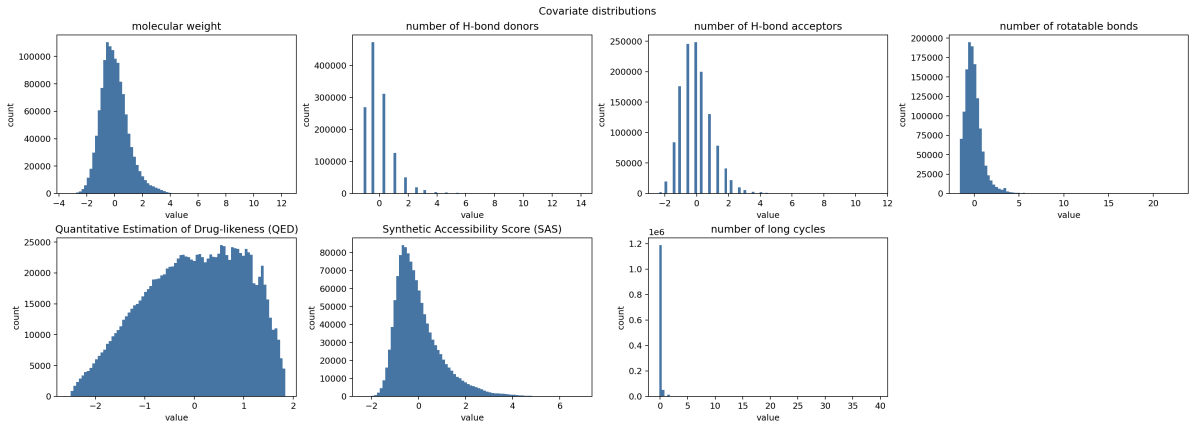

图 S7：预训练期间使用的分子协变量直方图（表 S3）。

C.3 InfoTransformerVAE 配置

	InfoTransformerVAE
潜在维度	$d = 256$（瓶颈_大小=2，d_模型=128）
最大解码长度	$L_{\max} = 128$
KL 权重	$\beta_{\mathrm{KL}} = 0.1$(kl_因子)

表 S4：实验中使用的 InfoTransformerVAE 架构和训练超参数

C.4 计算资源

所有实验均在 Nvidia Tesla V100 GPU 上运行。