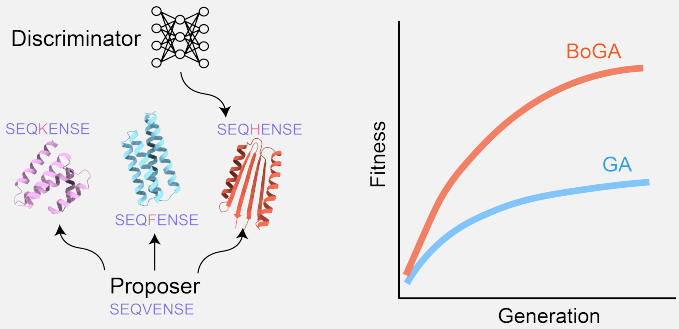
arXiv:2603.02753v1 [cs.LG] 3 Mar 2026

深度学习引导的进化优化，用于蛋白质设计

Erik Hartman[1], * Di Tang[1] Johan Malmstr¨om[1]

3月4日 , 2026



由于序列空间庞大且序列-功能关系复杂，设计具有理想特性的新型蛋白质仍是一项重大挑战。高效探索这一空间，以找出符合特定设计标准的序列，对于推动治疗药物和生物技术的发展至关重要。在此，我们提出 BoGA（贝叶斯优化遗传算法），这是一个将进化搜索与贝叶斯优化相结合、以高效遍历序列空间的框架。BoGA 将遗传算法作为随机提案生成器，整合到代理建模循环中，并根据先前的评估结果和代理模型预测对候选项进行优先级排序，从而实现数据高效的优化。我们通过对序列和结构设计任务进行基准测试，展示了 BoGA 的实用性；随后将其应用于设计针对肺炎溶血素的肽结合剂。肺炎溶血素是肺炎链球菌的一种关键毒力因子。BoGA 加速了高置信度结合剂的发现，展现出其面向不同目标实现高效蛋白质设计的潜力。该算法已在 BoPep 套件中实现，并以 MIT 许可证发布，可在GitHub获取。

[1]隆德大学医学院感染医学系，瑞典*通讯作者：Erik Hartman，
erik.hartman@med.lu.se

1 引言

设计具有特定特征的蛋白质，例如具备结合潜力、结构特性或催化活性，对于开发下一代治疗药物和推动生物技术发展具有变革性潜力 [1, 2]。然而，序列空间的天文数字般庞大，加之序列与功能之间复杂的非线性关系，使理性设计面临挑战。要识别符合精确设计标准的序列，就需要高效地在这一组合空间中进行导航；在此过程中，对每个候选项进行实验验证既耗时又成本高昂。为应对这一挑战，各种计算方法应运而生，用于高效探索序列空间并识别符合特定设计标准的候选项 [3, 4, 5, 6, 7, 8,9, 10, 11, 12, 13]。

一些方法将蛋白质设计表述为序列空间中的黑箱优化问题，并采用优化算法加以解决。例如，遗传算法（GAs）利用突变、重组和选择，迭代改进候选序列，模拟自然进化；事实证明，这些方法在包括蛋白质设计在内的广泛领域中都十分有效[14, 15, 16, 17, 6, 18, 19]。尽管遗传算法能够发现具有功能的蛋白质，但其效率受到寻找最优解所需评估次数的限制。

另一种常用于蛋白质设计的优化算法是贝叶斯优化（BO） [20]。在贝叶斯优化中，使用先前的评估结果训练代理模型，以近似目标函数，并引导搜索转向序列空间中有希望的区域。与遗传算法结合时，贝叶斯优化可以基于先前的评估结果和模型预测，以有原则的方式对遗传算法提出的候选项进行优先级排序 [21, 22, 23, 24, 25]。尽管文献中近期提出了这类算法，并将其称为代理辅助进化算法或机器学习引导的定向进化 [26], ，但它们尚未应用于蛋白质设计，其在这一领域的表现仍有待探索。

在此，我们介绍 **BoGA**（**贝**叶斯优化**遗**传算法（遗传**算**法），这是一种将进化搜索与贝叶斯优化结合于在线学习循环中的框架。基于我们此前开发的 **BoPep**（一种用于结合体发现的贝叶斯优化框架）[27], ，BoGA 将遗传算法作为随机提案生成器纳入代理建模循环，从而扩展了这一概念。我们首先在简单的序列和结构设计任务上对 BoGA 的性能进行基准测试，以展示其效用。随后，我们应用 BoGA 设计针对肺炎溶血素的肽结合剂；肺炎溶血素是一种对肺炎球菌毒力至关重要的成孔毒素 [28]。BoGA 在 BoPep 套件中实现，该套件以模块化方式支持嵌入模型、代理模型架构、概率推断模式和基于对接的评估函数。BoGA 以 MIT 许可证发布于GitHub。

2 主要内容

用于蛋白质序列设计的经典遗传算法（GA）通过迭代突变不断演化的序列种群，优化目标属性，例如抗菌效力或结合亲和力。每一代中，适应度最高的序列会被选中，随后重复这一过程，直至收敛或达到既定的停止准则。这种渐进式改进由随机突变算子驱动，它们为序列群体引入多样性并促进探索。然而，遗传算法通常需要多次评估目标函数才能找到高性能序列；当评估涉及结构预测或对接模拟时，计算成本可能很高。

贝叶斯优化（BO）引入一个基于已评估数据训练的代理模型，用于近似真实的目标函数。该代理模型能够捕捉序列空间中未观测区域的不确定性，并使采集函数（例如期望改进）能够优先选择新的候选项，在利用（对已知高性能序列附近进行采样）和探索（对不确定区域进行采样）之间取得平衡。

在此，我们在 BoGA 中结合这两种范式，并将其应用于序列设计问题，以寻找使预定义目标最大化或最小化的序列（**图 1**）。在这一方案中，遗传算法充当提案生成器，通过对表现最佳的序列进行随机突变，生成多样的候选序列。代理模型充当判别器，通过利用采集函数定义候选项的适应度，选择哪些提案应接受评估。只要目标函数的评估成本高于代理模型的训练和提案生成成本，并且代理模型具有足够的准确性，这种方法就能提高优化效率。BoGA 已经

采用最先进的方法进行了开发，包括使用蛋白质语言模型进行序列嵌入、使用最先进的概率深度学习模型作为代理模型，以及使用多个结构预测模型来预测该序列将折叠成的三维结构。

BoGA 的一项关键特性是模块化，因为它支持多种采集函数、代理架构、嵌入策略和选择策略。下面我们将介绍 BoGA 的通用设置，随后说明我们在实验中采用的具体方案。接着，我们展示 BoGA 在序列级和结构级优化任务上的表现，之后将其用于设计针对肺炎溶血素的肽结合剂。

2.1 BoGA框架描述

首先，令 $\mathcal{X}$ 表示所有可变长度为 L的肽序列构成的空间，其中每个序列 $\mathbf{x} \in \mathcal{A}^L$， $\mathcal{A}$ 表示标准的 20 字母氨基酸字母表。设计任务从一个或多个种子序列 $\mathcal{X}_0 = \{\mathbf{x}_1, \ldots, \mathbf{x}_{n_0}\} \subset \mathcal{X}$开始。在每一代$t$，BoGA 都会维护一个已评估序列的数据集

$$\mathcal{D}_t = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n_t}, \quad y_i = f(\mathbf{x}_i), \tag{1}$$

其中 $f : \mathcal{A}^L \to \mathbb{R}$ 是目标函数，它将序列 $\mathbf{x}$ 映射为一个标量分数，例如结合剂设计中的结合亲和力或对接能。

在每一代中，都会从当前种群 $\mathcal{X}_t$ 或完整评估历史 $\mathcal{D}_t$ 中选出一个精英子集 S_k ，作为突变的亲代。选择函数 $\mathcal{S}(\mathcal{X}_t,\ \mathcal{D}_t,\ k)$用于确定 S_k；例如，根据 $f(\mathbf{x})$对候选项进行排序并选择排名最靠前的 k ，可得到

$$S_k = \mathcal{S}(\mathcal{X}_t, \mathcal{D}_t, k) = \{\mathbf{x}_i : f(\mathbf{x}_i) \le f(\mathbf{x}_j) \text{ for at most } k-1 \text{ other candidates}\}, \tag{2}$$

不过，也可以采用其他策略，例如按比例选择排名靠前的候选项、指数抽样或基于阈值的筛选。变异算子 $\mathcal{M}$通过删除、插入和替换等突变算子，以随机方式生成多样化的候选序列池，

$$\mathcal{X}' = \{\mathbf{x}'_i \sim \mathcal{M}(S_k) : i = 1, \ldots, k_{\text{propose}}\}. \tag{3}$$

每个提议的序列$\mathbf{x}' \in \mathcal{X}'$ 都会使用序列编码器转换为连续嵌入$\mathbf{z}' = \phi(\mathbf{x}')$。这些嵌入构成代理模型 $f_{\boldsymbol{\theta}}$的输入空间。该代理模型会为每个嵌入后的候选项预测适应度 $f_{\boldsymbol{\theta}}(\mathbf{z}')$。

为引导探索，代理模型的输出会经由采集函数 $\alpha(\cdot)$转换，该函数优先考虑预期能够带来改进的候选项，这些候选项可以具有较高的预测性能，也可以具有较高的模型不确定性。从候选池中选出采集值最大的 m_{select} 个候选项进行显式评估，

$$\mathbf{x}_{\text{next}} = \underset{\mathbf{x}' \in \mathcal{X}'}{\arg\max}\, \alpha\left(\hat{f}_{\boldsymbol{\theta}}(\phi(\mathbf{x}'))\right), \qquad |\mathbf{x}_{\text{next}}| = m_{\text{select}}, \tag{4}$$

其真实得分 $f(\mathbf{x}_{\text{next}})$ 则通过评估获得，例如进行结构预测并计算生物物理特性。这些新的评估结果扩充了用于训练代理模型的数据集，随后通过最小化预测损失来更新模型参数

$$\boldsymbol{\theta} \leftarrow \underset{\boldsymbol{\theta}}{\arg\min}\, \mathcal{L}\left(\hat{f}_{\boldsymbol{\theta}}(\phi(\mathbf{x})), f(\mathbf{x})\right). \tag{5}$$

这一循环构成了 BoGA优化周期的核心（算法1）。

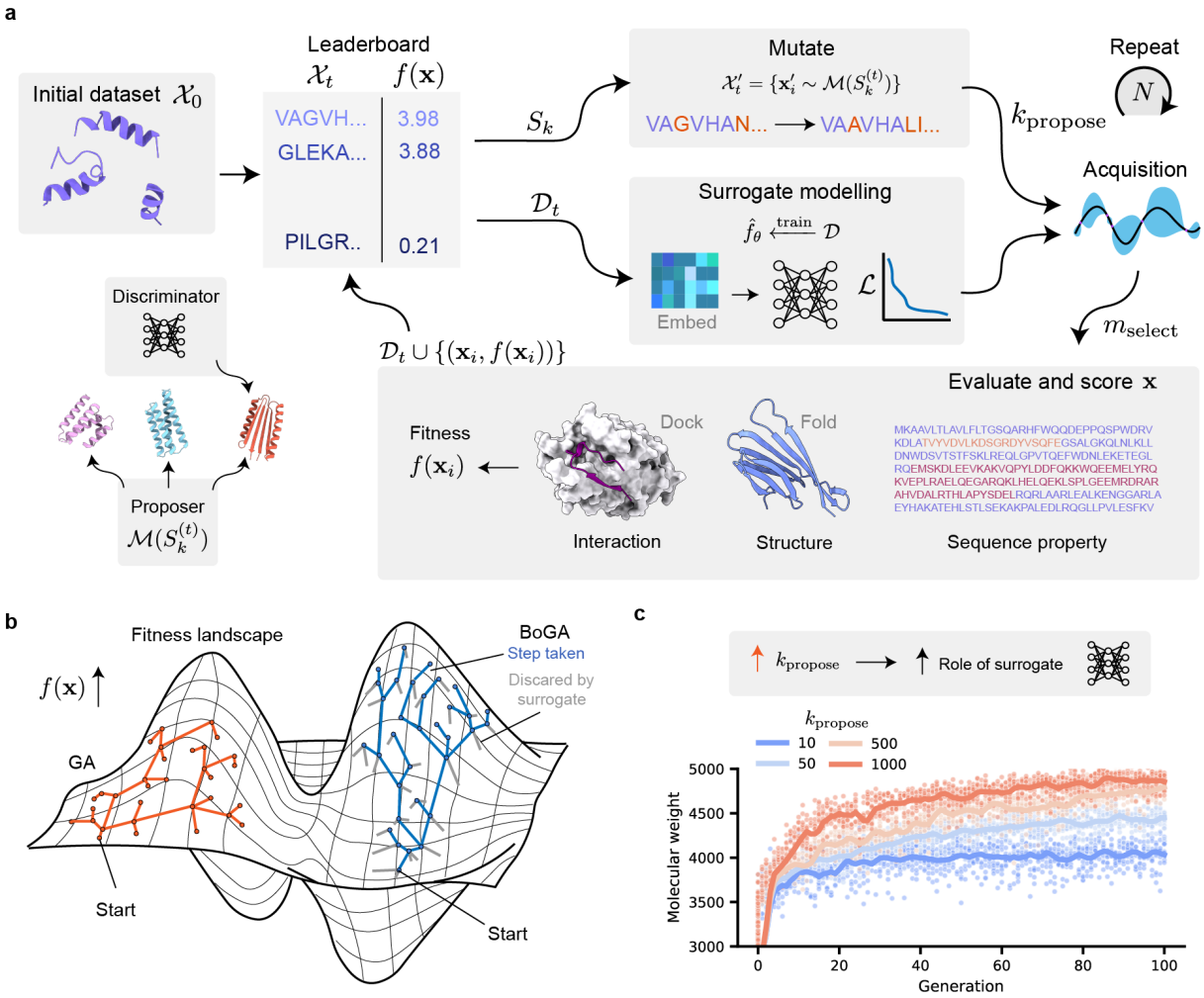


图 1：**BoGA 将进化优化与贝叶斯选择相结合，用于序列设计。a** 优化循环示意图。从初始已评估数据集 $\mathcal{D}_0$中，从排行榜中选出精英集 S_k，排行榜按每条序列的适应度 f（$\mathbf{x}$）对其进行排序。提议器（遗传突变：替换/插入/删除）生成一个包含 k_{propose} 个候选项的候选池 $\mathcal{X}'$。代理模型 f_θ对序列进行嵌入并预测评分 ˆ；采集函数 α 选出适合显式评估的排名前 m_{select} 个候选项。适应度函数由用户确定，可利用源自氨基酸序列、折叠结构和/或由折叠复合物定义的链间相互作用的属性。新评分的序列会加入数据集 $\mathcal{D}_t$，并重复该循环。**b** 序列空间中的概念性搜索。标准遗传算法（橙色）会评估许多提议，包括会降低适应度的提议；而 BoGA（蓝色）使用代理模型剔除低价值提议（灰色），并将评估集中在更有可能提高适应度的提议上。**c**以分子量最大化为目标时的优化轨迹示例，展示了随着 k_{propose} 增加，性能得到提升。

算法1： BoGA：贝叶斯优化遗传算法

输入： 初始序列 $\mathcal{X}_0$、目标函数 f、嵌入模型 ϕ、世代数 T，
精英规模 k、提议规模 k_{propose}、选择规模 m_{select}

输出： 优化种群 $\mathcal{X}_T$

初始化数据集 $\mathcal{D}_0 \leftarrow \{(\mathbf{x}\ f(\mathbf{x})) : \mathbf{x} \in \mathcal{X}_0\}$;
初始化种群 $\mathcal{X}_0$;
对于 $t = 1$ **到** T **执行**
  // 训练代理模型
  $\theta_t \leftarrow \arg\min_{\theta} \mathcal{L}(\ \hat{f}_{\theta}(\phi(\mathbf{x}))\ f(\mathbf{x}))$ 对于 $(\mathbf{x}\ f(\mathbf{x})) \in \mathcal{D}_{t-1}$;
  // 选择精英序列（例如，按排名）
  $S_k \leftarrow \mathcal{S}(\mathcal{X}_{t-1}, \mathcal{D}_{t-1}, k)$;
  // 通过突变生成候选方案
  $\mathcal{X}' \leftarrow \emptyset$;
  for $i = 1$ **到** $k_{propose}$ **do**
    $\mathbf{x}' \sim \mathcal{M}(S_k)$; // 应用变异算子
    $\mathcal{X}' \leftarrow \mathcal{X}' \cup \{\mathbf{x}'\}$;
  // 使用采集函数对候选方案进行评分
  for $\mathbf{x}' \in \mathcal{X}'$ **do**
    $\mathbf{z}' \leftarrow \phi(\mathbf{x}')$; // 嵌入序列
    $a(\mathbf{x}') \leftarrow \alpha(\ \hat{f}_{\theta_t}(\mathbf{z}'))$; // 计算采集值
  // 选择排名靠前的候选项进行评估
  $\mathcal{X}_{\text{eval}} \leftarrow \arg\text{top}^{m_{\text{select}}}{}_{\mathbf{x}' \in \mathcal{X}'}\ a(\mathbf{x}')$;
  // 评估所选候选项
  for $\mathbf{x} \in \mathcal{X}_{eval}$ **do**
    $y \leftarrow f(\mathbf{x})$; // 昂贵评估
    $\mathcal{D}_t \leftarrow \mathcal{D}_{t-1} \cup \{(\mathbf{x}\ y)\}$;
    $\mathcal{X}_t \leftarrow \mathcal{X}_{t-1} \cup \{\mathbf{x}\}$;
返回 $\mathcal{X}_T$;

2.2 用于序列与结构优化的BoGA

BoGA包含若干可针对不同性能进行调节的参数。其中一个参数是突变函数生成的提议数量 k_{propose}，与之对照的是被评估候选项的数量 m_{select}。当二者相同时，该方法即为标准遗传算法。然而，随着 k_{propose} 变得大于 m_{select}，代理模型在剔除候选项中逐渐承担更大的作用。我们首先考察了 k_{propose} 如何影响较简单序列特异性属性的优化性能，例如 β-折叠比例和归一化疏水矩（uHrel）。此处，在氨基酸水平上预测的 β-折叠比例被定义为序列中E、M、A和L残基所占的比例。使用ESM-2将序列嵌入连续的潜在空间，使用主成分分析将嵌入维度降至100维，并使用深度证据回归双向门控循环单元作为代理模型。 m_{select} 固定为10，而 k_{propose} 在 $[10, 50, 100, 500, 1000]$中变化。uHrel是代理模型更难预测的复杂属性，从而导致验证$R^2 \approx 0.8$，而β-折叠倾向则可预测，验证结果为 $R^2 \approx 1$。在这两种情况下，性能均随 k_{propose}提高，不过折叠比例优化比uHrel优化受益更多（**图2**）。

无需预训练的优化技术能够灵活定义优化目标，从而可以改变优化目标，引导生成过程朝任意

期望指标迈进。为展示这一点，我们将该算法用于设计具有特定二级结构的蛋白质。我们使用 AlphaFold 2 [29] 进行结构预测，并将通过 DSSP 计算出的二级结构乘以预测的 TM-score 作为优化目标，同时优化结构和模型置信度。我们比较了将 k_{propose} 设为 10（即标准遗传算法）和设为 500（由代理模型选出 2% 的序列进行评估）这两种情况。在这些情况下，提高 k_{propose} 改善了优化性能，使最终 10 代样本的平均适应度在片层目标中从 0.178 ± 0.748 提升至 0.253 ± 0.0750，在螺旋目标中从 0.367 ± 0.578 提升至 0.507 ± 0.105（**图 2c**）。这表明 BoGA 可用于优化任意目标，而代理模型则可通过优先筛选待评估的候选项来提高优化效率。

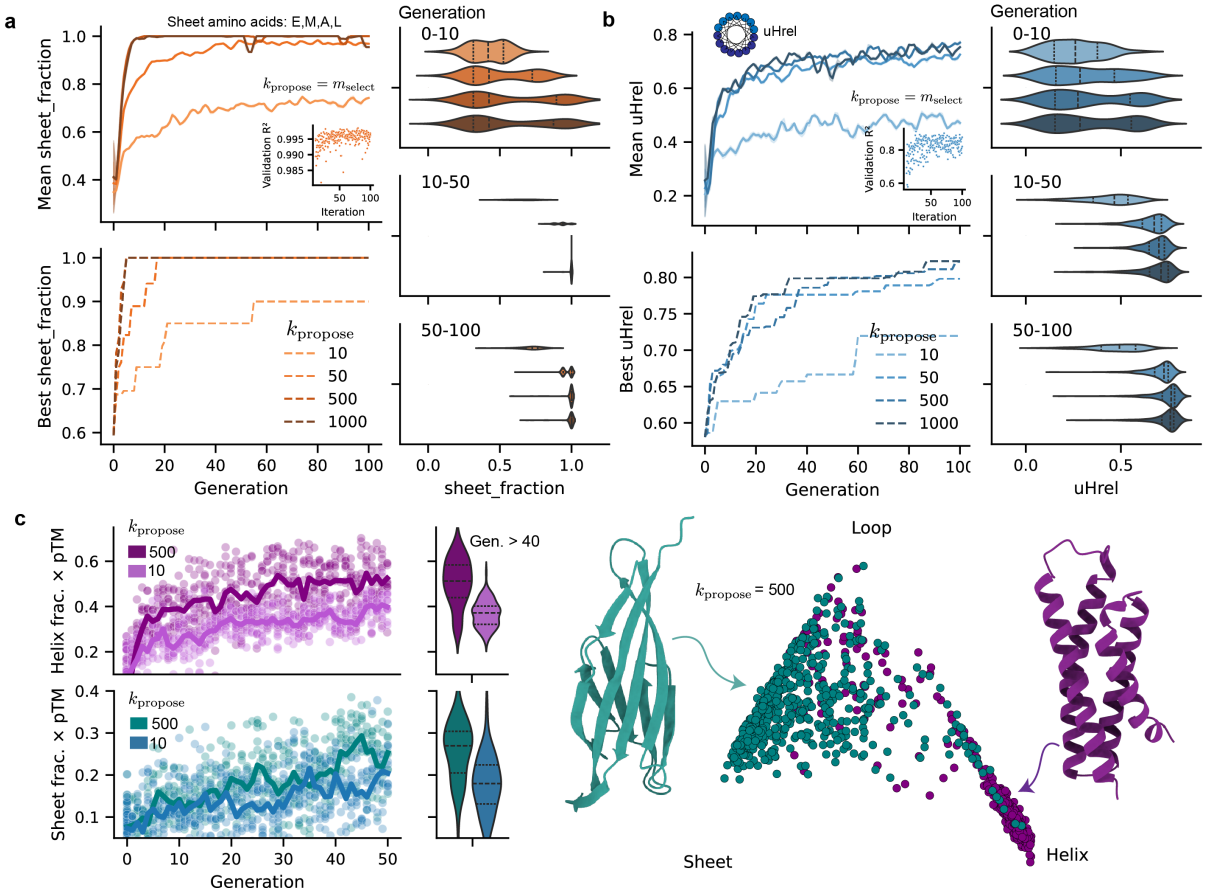


图 2：**BoGA 提高了序列级和结构级目标的优化效率。a** 对 β-折叠比例（E、M、A和L残基所占比例）进行优化。左图展示了不同 k_{propose} 值下的优化轨迹，右图展示了优化过程中不同阶段的适应度值分布。插图展示了优化过程中代理模型 R^2 的变化。**b**与**a** 布局相似，但目标函数为归一化疏水矩（uHrel）。**c** 使用 AlphaFold 2 预测的二级结构进行结构引导优化，并根据预测TM分数（pTM）加权。左图展示优化轨迹，小提琴图展示最后 10 代的适应度值分布。右图为在 $k_{\text{propose}} = 500$ 条件下运行所得样本的二级结构三元图。

2.3 针对细菌毒力因子生成结合剂

为了评估 BoGA 能否生成具有生物学相关性的结合剂，我们设计了针对肺炎溶血素（PLY）的结合剂。肺炎溶血素是一种成孔毒力因子，在肺炎链球菌的发病机制中起关键作用 [28]。PLY 通过结构域 4 与富含胆固醇的宿主细胞膜结合，并随后寡聚化形成大型 β桶状孔道，从而导致细胞裂解。因此，中和 PLY 是一种颇具前景的治疗策略，有望同时降低细菌毒力和宿主驱动的免疫病理学。此前，我们鉴定出一种抗体，它通过结合位于结构域 4 中的表位，能够有效中和 PLY[30]。在此，我们利用 BoGA 设计肽结合剂，靶向同一表位（**图 3a**）。

优化目标采用了此前开发的标量组合指标，该指标综合了结构预测置信度指标和结合有利性的生物物理测量值 [27]。我们采用期望改进采集函数，在利用与探索之间取得平衡。与之前类似，我们结合使用 ESM-2 和主成分分析来嵌入肽序列，并使用深度证据回归 [31] BiGRU 作为代理模型。初始化后仅优化了一次超参数，因为先前的实验表明，在后续世代中优化超参数会损害性能，这可能是由于数据集的同质性所致。我们使用 Boltz-2 预测 [32] 肽-蛋白质复合物的构象。参数 k_{propose} 设为 500，m_{pool} 设为 10。优化进行了 100 个世代，初始序列数为 100。作为比较，我们还使用$k_{\text{propose}} = 10$进行了优化，以量化代理模型对结合剂设计的影响。

使用 $k_{\text{propose}} = 500$ 加速了高评分结合剂的发现（**图 3b**），并使种群适应度提升得更快，表明其能够更高效地探索序列空间中有潜力的区域。随着优化的进行，适应度分布呈现双峰，这可能说明模型既选出了高置信度、高评分的结合剂，也选出了置信度较低但在预期改进标准下得分较高的结合剂。此外，更大的候选池产生了更多高置信度结合剂，其置信度通过预测界面 pTM（ipTM）和肽预测对齐误差（PAE）衡量（**图 3c**）。

为进一步优化 BoGA 产生的顶级候选项，我们采用了优化后流程，包括使用 ProteinMPNN 进行序列恢复 [33]以及使用 FastRelax 进行结构松弛（**图 3d**） [5, 33, 34, 35]。对排名前 100 的 BoGA 序列进行了三轮序列恢复和松弛，生成了 400 个优化后的候选项。随后，将这些候选项与全长 PLY结构重新对接并重新评分。优化后的设计在 pLDDT 与 ipTM 空间中聚集于高置信度区域，其中许多设计具有较优的 Boltz-2 界面距离评分（**图 3e**）。经过质量、溶解性和结构一致性筛选后，最终获得了 41 个高置信度结合剂（**图 3f**）。排名靠前的结合剂与 PLY 形成复合物的代表性预测结构见**图 3f**。使用 AlphaFold 3 [36] 独立进行的结构预测进一步支持了顶级结合剂的结合姿势。两种建模方法均表明，其结合模式一致、界面置信度高且结合自由能低（ΔG）（**图 3g**）。

本应用凸显了 BoGA 在针对具有生物学相关性的靶标高效设计肽结合剂方面的实用价值，并展示了代理建模如何加速高置信度候选项的识别。

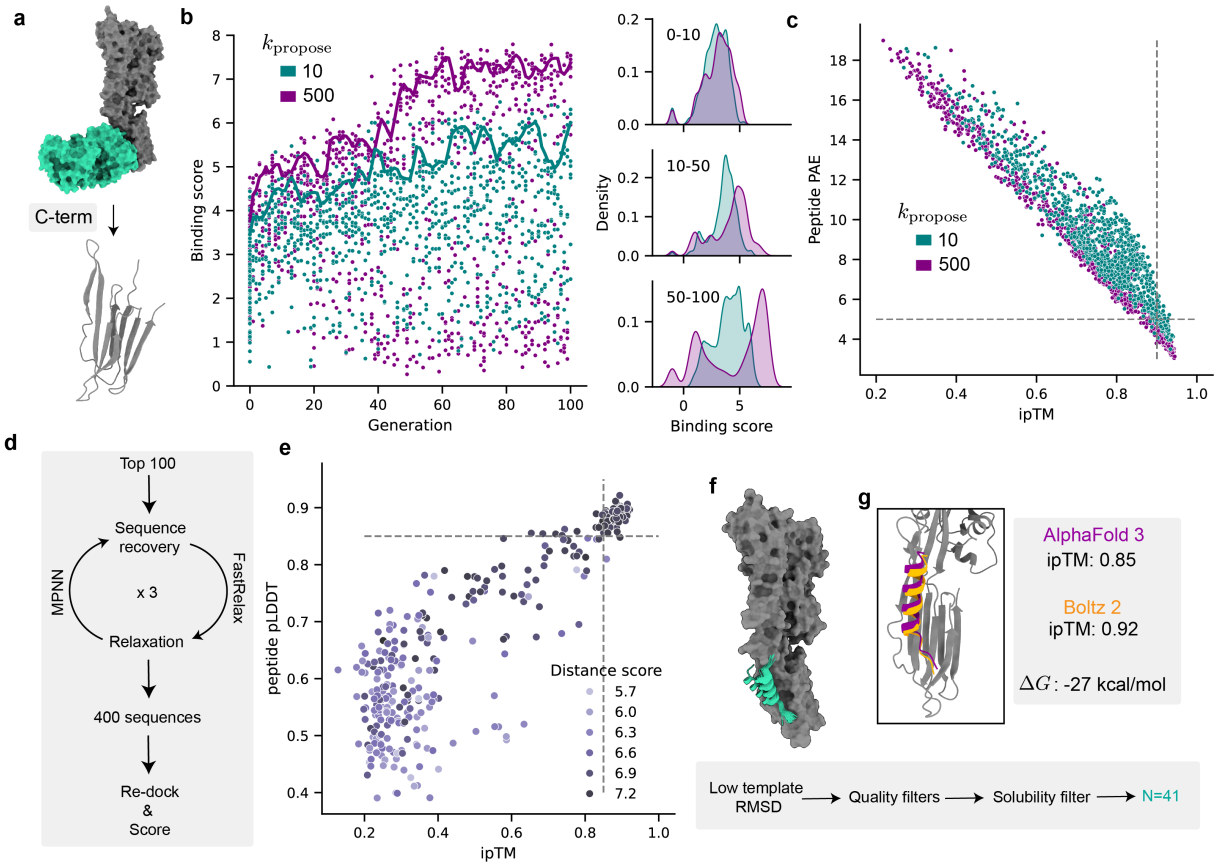


图 3：**BoGA 可高效设计靶向肺炎溶血素的肽结合剂。a** 中和单克隆抗体与 PLY 复合物示意图。PLY 以灰色显示，抗体以绿色显示。底部的模型表示结构域 4。**b** 在 100 个世代中，$k_{\text{propose}} = 10$ （青绿色）和 $k_{\text{propose}} = 500$ （紫色）的结合剂评分优化轨迹。各点表示单个经评估的候选项；实线表示滚动计算的最高四分位数均值。右图：早期（0-10）、中期（10-50）和后期（50-100）世代的结合评分分布核密度估计，显示较大的候选池可加快发现高评分结合剂。**c** 经评估序列的预测界面 pTM（ipTM）与肽预测对齐误差（PAE）之间的关系。较大的 k_{propose} 会增加对高置信度、低 PAE 结合剂的采样。虚线表示 ipTM=0.9 和 PAE=5 Å 的截断值。**d**优化后精修方法概述。使用 ProteinMPNN 和 FastRelax 对 BoGA 的前 100 条序列进行三轮序列恢复，得到共 400 个候选项，随后将其重新对接至全长 PLY 并进行评分。**e**精修候选项的 pLDDT 与 ipTM。各点根据 Boltz-2 界面距离评分着色。高置信度结合剂聚集在 i pTM 和 pLDDT 均较高的区域（右上象限，虚线所示）。**f** 表现最佳的结合剂（青色）与 PLY（灰色）结合的预测复合物示例（Boltz-2）。根据质量、溶解性和结构一致性筛选候选项后，最终获得 41 个高置信度结合剂。**g** 使用 AlphaFold 3 和 Boltz 2 对一款表现最佳的结合剂进行正交结构预测。两个模型均支持一致的结合姿势，且界面置信度较高（AlphaFold 3 的 ipTM= 0.85；Boltz 2 的 ipTM = 0.92）。PyRosetta 估算的结合自由能（ΔG） [34]表明相互作用具有显著有利性。

3 讨论

BoGA 将进化优化与贝叶斯决策相结合，形成了一种能够高效探索序列空间的算法。与评估所有提出的候选项的经典遗传算法不同，BoGA 使用代理模型对候选项进行优先级排序，以供评估，从而将计算资源集中于最有希望的序列。对于序列、结构和结合剂级目标，将生成 与评估解耦，可降低计算成本，并提升对高分候选项的选择效果。

与 BindCraft 等基于幻觉的方法或 RFDiffusion 等基于扩散的方法等其他流行的当代方法相比，[7, 9], BoGA 具有多项优势。首先，BoGA 无需在大型数据集上训练复杂模型。其次，其优化目标灵活，可适配任何可微或不可微指标，无需重新训练。第三，BoGA 能够灵活地融入结构预测和对接方面的进展，因为这些模型仅用于评估，而非生成。最后，遗传算法能够通过突变更自由地探索序列空间，因此 BoGA 受生成模型归纳偏置的限制较少。

这些优势是以无法使用预训练模型在单次前向传播中生成候选项为代价的；与迭代优化相比，这种方式在计算上更高效。不过，在典型流程中，结构预测通常用于筛选候选项，因此主要瓶颈在于评估，而非生成。在此类情境下，BoGA 等算法带来的额外计算开销往往是合理的。

代理模型的益处取决于模型的质量。如果代理模型的预测能力较差或校准不当，采集分数就会失去参考价值，导致其行为类似于经典遗传算法；若模型引入系统性偏差，效果甚至会更差。相反，随着代理模型准确度和不确定性校准能力的提高，BoGA 会越来越受采集函数驱动，并能在所提议的候选项中实现近乎确定性的优化。

总的来说，BoGA 被设计为一种通用且可扩展的蛋白质设计策略。BoPep 框架支持灵活替换序列嵌入、代理模型架构、采集函数和突变算子，而无需改变核心优化循环 [27]。总体而言，BoGA 展示了将进化式提议与贝叶斯决策相结合，如何增强用于蛋白质设计的进化算法。通过利用代理模型对候选项进行优先级排序，BoGA 能够加速发现高性能序列，同时适应多种不同的设计目标。

4 方法

广义而言，BoGA 是一种由代理模型引导的氨基酸序列设计遗传算法。通过突变迭代生成候选序列，对其进行评分，并用其训练代理模型，以指导进一步探索。

令 $\mathcal{X}$ 表示所有可变长度 L 的肽序列构成的空间，其中每个序列 $\mathbf{x} \in \mathcal{A}^L$，$\mathcal{A}$ 表示由 20 个标准字母组成的氨基酸字母表。设计任务从一个或多个种子序列 $\mathcal{X}_0 = \{\mathbf{x}_1, \ldots, \mathbf{x}_{n_0}\} \subset \mathcal{X}$ 开始。在每一代 t，BoGA 都会维护一个已评估序列的数据集

$$\mathcal{D}_t = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n_t}, \quad y_i = f(\mathbf{x}_i), \tag{6}$$

其中，$f : \mathcal{A}^L \to \mathbb{R}$ 是将序列 $\mathbf{x}$ 映射为标量评分的目标函数，例如在结合剂设计中映射为结合亲和力或对接能。

每个序列 $\mathbf{x}_i$ 都会被嵌入为数值表示

$$\mathbf{z}_i = \phi(\mathbf{x}_i), \tag{7}$$

其中 $\phi : \mathcal{A}^L \to \mathbb{R}^D$ 表示一种嵌入模型，例如 ESM-2，它将序列映射到 D维潜在空间。嵌入$\mathbf{z}_i \in \mathbb{R}^D$ 为代理学习提供了连续向量空间。当 D较大时，为降低计算复杂度并缓解过拟合，可在代理模型训练前对嵌入应用主成分分析（PCA）或变分自编码器（VAE）等降维技术。

ˆ 代理模型 $f_{\boldsymbol{\theta}}$以神经网络权重 $\boldsymbol{\theta}$为参数，通过最小化预测分数与真实分数之间的损失来近似 f:

$$\boldsymbol{\theta}_{t+1} = \arg\min_{\boldsymbol{\theta}} \mathcal{L}\big(\hat{f}_{\boldsymbol{\theta}}(\mathbf{z}_i), y_i\big), \quad (\mathbf{x}_i, y_i) \in \mathcal{D}_t. \tag{8}$$

通过选择函数 $\mathcal{S}(\mathcal{X}_t,\ \mathcal{D}_t,\ k)$选出精英子集；该函数作用于当前种群 $\mathcal{X}_t$ 或完整的评估历史 $\mathcal{D}_t$:

$$S_k^{(t)} = \mathcal{S}(\mathcal{X}_t, \mathcal{D}_t, k). \tag{9}$$

例如，按适应度对序列排序并选取前 k 个，可得到 $S_k^{(t)} = \{\mathbf{x}_1, \ldots\ldots, \mathbf{x}_k\}$，其中 $f(\mathbf{x}_1) < f(\mathbf{x}_2) < \cdots < f(\mathbf{x}_k)$。随后选取该精英子集进行突变。随机突变算子 $\mathcal{M}$ 对 $S_k^{(t)}$ 进行扰动，生成一个大小为 k_{propose}的候选池，

$$\mathcal{X}'_t = \{\mathbf{x}'_i \sim \mathcal{M}(S_k^{(t)}) : i = 1, \ldots, k_{\text{propose}}\}, \tag{10}$$

其中，$\mathcal{M}$ 以学习得到的概率引入氨基酸替换、插入或删除。

对于每个提出的序列 $\mathbf{x}' \in \mathcal{X}'_t$，都会计算其嵌入 $\mathbf{z}' = \phi(\mathbf{x}')$，代理模型随后预测其适应度 $f_{\boldsymbol{\theta}}(\mathbf{z}')$。接着，使用采集函数 $\alpha(\cdot)$对候选项进行评分，并选出m_{select} 个最有希望的候选项进行评估:

$$\mathbf{x}_{\text{next}} = \arg\max_{\mathbf{x}' \in \mathcal{X}'_t} \alpha\big(\hat{f}_{\boldsymbol{\theta}}(\phi(\mathbf{x}'))\big). \tag{11}$$

随后通过物理评估获得真实目标值，例如对接或协同折叠，并迭代更新种群和数据集:

$$\mathcal{X}_{t+1} = \mathcal{X}_t \cup \mathbf{x}_{\text{next}}, \qquad \mathcal{D}_{t+1} = \mathcal{D}_t \cup \{(\mathbf{x}_i, f(\mathbf{x}_i)) : \mathbf{x}_i \in \mathbf{x}_{\text{next}}\}. \tag{12}$$

此优化构成了 BoGA 算法的核心。该比率

$$\frac{m_{\text{select}}}{k_{\text{propose}}}, \tag{13}$$

决定代理模型的影响程度：随着该比率降低，代理模型在引导搜索转向序列空间中高性能区域时发挥的作用会更大。

BoGA 在 BoPep 框架中实现 [27], 该框架为嵌入模型、代理模型架构、概率推断模式和基于对接的评估函数提供模块化支持。

4.1 计算复杂度

BoGA 的计算成本由三个主要部分组成：目标函数评估、代理模型训练和序列嵌入。对于不采用代理建模的标准遗传算法（GA），每一代的主要成本是使用目标函数评估 m_{select}

候选项，其每代复杂度为 $\mathcal{O}(m_{\text{select}} \cdot C_f)$，其中 C_f 表示单次评估的成本。经过 T 代后，总成本按$\mathcal{O}(T \cdot m_{\text{select}} \cdot C_f)$的规模增长。

BoGA 会因代理建模产生额外开销，但旨在减少达到收敛所需的昂贵评估次数。在每一代中，代理模型都基于不断扩大的数据集进行训练，成本为 C_{train}； k_{propose} 个提议的候选项以 C_{embed} 的成本进行嵌入；代理模型再以 $C_{\text{surrogate}}$ 的成本评估这些嵌入。因此，BoGA 每代的成本为 $\mathcal{O}(C_{\text{train}} + C_{\text{embed}} + C_{\text{surrogate}} + m_{\text{select}} \cdot C_f)$。

当 $C_f \gg C_{\text{train}} + C_{\text{embed}} + C_{\text{surrogate}}$ 时，BoGA 的主要优势便会显现；也就是说，评估目标函数的成本远远高于代理模型训练、嵌入和推理的总成本。在这种情况下，代理模型可以以相对较低的计算成本筛除低质量候选项，使算法能够将昂贵的评估集中在序列空间中有前景的区域。这种权衡在蛋白质设计应用中尤其有利，因为结构预测的成本比代理模型操作高出几个数量级。

4.2 评估 k_{propose} 的影响

为探究候选池大小对优化效率的影响，我们在固定每代高成本评估次数（m_{select}）的情况下改变了 k_{propose}。对于每个目标，运行均以相同数量的序列初始化，并执行固定数量的世代。通过设置 $k_{\text{propose}} = m_{\text{select}}$ 实现遗传算法基线，同时保持其他所有组件不变。

序列优化针对两个目标进行了简单的序列级优化：最大化预测的 β-折叠比例（E、M、A和L残基的比例），以及最大化归一化疏水矩（uHrel）。使用 ESM-2 对序列进行嵌入，并通过主成分分析将嵌入降至100维。深度证据回归双向门控循环单元用作代理模型。对于这两个目标，m_{select} 固定为10，而 k_{propose} 则在[10, 50, 100, 500, 1000]范围内变化。每种设置均运行100个世代。

结构优化对于结构引导优化，我们在固定 m_{select}的情况下采用相同的 k_{propose} 扫描，但目标评估需要进行结构预测。使用 AlphaFold 2 折叠候选序列，并使用 DSSP 指定二级结构。结构目标定义为目标二级结构比例乘以 AlphaFold 2 预测TM分数，从而同时奖励结构含量和预测置信度。

4.3 肺炎溶血素的应用

使用 BoGA 针对肺炎溶血素（PLY）进行结合剂设计，并基于复合物结构进行评估。优化以 100 条肽序列为初始序列，运行 100 代。肽的最短长度设为 8，最长长度设为 25。使用 ESM-2 对序列进行嵌入，并通过主成分分析将其降至 100 维。代理模型采用深度证据回归双向门控循环单元（BiGRU），使用所有累积的评估结果进行训练，并结合期望改进采集函数，从候选池中选择候选项。初始超参数优化进行了 50 次试验和 5 折交叉验证，随后在优化的其余过程中固定模型参数。

对每条选定的肽，使用 Boltz-2 预测复合物，采用 3 个模型和 5 次循环。根据先前开发的标量目标函数对候选项进行评分，该函数结合模型置信度和界面有利性指标（界面 ΔG、ipTM、距离评分、肽pAE 和 Rosetta评分），这些指标均从预测的复合物中得出。我们比较了代理主导设置（$k_{\text{propose}} = 500$、 $m_{\text{select}} = 10$）与类遗传算法基线（$k_{\text{propose}} = 10$、 $m_{\text{select}} = 10$），以量化代理筛选在结合体发现中的效益。补充信息中提供了一个用于肺炎溶血素设计任务的配置示例。

为优化排名靠前的候选项，我们对表现最佳的 100 条 BoGA 序列进行了三轮 Pro- tein MPNN [33] 序列恢复，随后使用 Rosetta FastRelax [34, 35], 得到 400 个优化变体。我们将优化后的候选项重新对接到全长 PLY结构，并使用相同的评估流程重新评分；随后筛选出界面置信度较高且整体结构质量良好的最终结合剂。我们使用 AlphaFold 3 对一部分排名靠前的候选项进行了复合物预测评估。

5 致谢

我们感谢 Jonas Wallin 就符号表示和算法呈现提供了宝贵建议。我们也感谢 Malcolm Siljehag Alencar 就该算法进行了有益的讨论并提供反馈。

6 资助

EH 获瑞典皇家科学院资助（ME2024-0049）。J.M. 是 I-SPY网络的成员，该网络由 Leducq心血管研究基金会资助。本项目获得欧洲研究理事会（2024-ADG 101200871）、Mats Paulssons基金会（2025-0042）和 Alfred ¨Osterlunds Foundation 的支持。本文所表达的观点和意见仅代表作者本人，并不一定反映欧盟或欧洲研究理事会的立场。欧盟和资助机构均不对此承担责任。

7 代码可用性

BoGA 可在 BoPep 套件中获取，地址：GitHub。

参考文献

1. Fosgerau K 和 Hoffmann T. 肽类疗法：现状与未来方向。《今日药物发现》2015年1月；20:122–8. DOI: 10.1016/j.drudis.2014.10.003。可见于：http://dx.doi.org/10.1016/j.drudis.2014.10.0032. Muttenthaler M、King GF、Adams DJ 和 Alewood PF. 肽类药物发现趋势。《自然综述：药物发现》2021年2月；20:309–25. DOI: 10.1038/s41573-020-00135-8。可见于：http://dx.doi.org/10.1038/s41573-020-00135-83. Bhat S、Palepu K、Hong L、Mao J、Ye T、Iyer R、Zhao L、Chen T、Vincoff S、Watson R、Wang TZ、Srijay D、Kavirayuni VS、Kholina K、Goel S、Vure P、Deshpande AJ、Soderling SH、DeLisa MP 和 Chatterjee P. 通过对比语言建模，针对构象多样的靶标从头设计肽结合剂。《科学进展》2025年1月；11. DOI: 10.1126/sciadv.adr8638。可见于：http://dx.doi.org/10.1126/sciadv.adr8638

4. Leyva YC、Torres MDT、Oliva CA、Fuente-Nunez C de la 和 Brizuela CA. 采用配钥匙机方法进行定制化结构肽设计。《自然·机器智能》2025年10月；7:1685–97. DOI：10 . 1038 / s42256 - 025 - 01119 - 2. 可见于：http://dx.doi.org/10.1038/s42256-025-01119-2 5. Bennett NR、Coventry B、Goreshnik I、Huang B、Allen A、Vafeados D、Peng YP、Dauparas J、Baek M、Stewart L、DiMaio F、De Munck S、Savvides SN 和 Baker D. 利用深度学习改进从头蛋白质结合剂设计。《自然·通讯》2023年5月；14. DOI：10.1038/s41467-023-38328-5. 可见于：http://dx.doi.org/10.1038/s41467- 023-38328-5 6. Jendrusch MA、Yang ALJ、Cacace E、Bobonis J、Voogdt CGP、Kaspar S、Sch weimer K、Perez-Borrajero C、Lapouge K、Scheurich J、Remans K、Hennig J、Typas A、Korbel JO 和 Sadiq SK. AlphaDesign：一种基于 AlphaFold 的从头蛋白质设计框架。《分子系统生物学》2025年6月；21:1166–89. DOI：10.1038/s44320-025-00119-z. 可见于：http://dx.doi.org/10.1038/s44320-025-00119-z 7. Pacesa M、Nickel L、Schellhaas C、Schmidt J、Pyatova E、Kissling L、Barendse P、Choud-hury J、Kapoor S、Alcaraz-Serna A、Cho Y、Ghamary KH、Vinu´e L、Yachnin BJ、Wollacott AM、Buckley S、Westphal AH、Lindhoud S、Georgeon S、Goverde CA、Hatzopoulos GN、G¨onczy P、Muller YD、Schwank G、Swarts DC、Vecchio AJ、Schneider BL、Ovchinnikov S 和 Correia BE. 使用 BindCraft 单次设计功能性蛋白质结合剂。《自然》2025年8月；646:483–92. DOI：10 . 1038 / s41586 - 025 - 09429 - 6. 可见于：http : // dx.doi.org/10.1038/s41586-025-09429-6 8. Stark H、Faltings F、Choi M、Xie Y、Hur E、O’Donnell T、Bushuiev A、U¸car T、Passaro S、Mao W、Reveiz M、Bushuiev R、Pluskal T、Sivic J、Kreis K、Vahdat A、Ray S、Goldstein JT、Savinov A、Hambalek JA、Gupta A、Taquiri-Diaz DA、Zhang Y、Hatstat AK、Arada A、Kim NH、Tackie-Yarboi E、Boselli D、Schnaider L、Liu CC、Li GW、Hnisz D、Sabatini DM、DeGrado WF、Wohlwend J、Corso G、Barzilay R 和 Jaakkola T. BoltzGen：迈向通用结合剂设计。2025年11月。 DOI：10.1101/2025.11.20.689494. 可见于：http://dx.doi.org/10.1101/2025.11.20.689494 9. Watson JL、Juergens D、Bennett NR、Trippe BL、Yim J、Eisenach HE、Ahern W、Borst AJ、Ragotte RJ、Milles LF、Wicky BIM、Hanikel N、Pellock SJ、Courbet A、Sheffler W、Wang J、Venkatesh P、Sappington I、Torres SV、Lauko A、De Bortoli V、Mathieu E、Ovchinnikov S、Barzilay R、Jaakkola TS、DiMaio F、Baek M 和 Baker D. 利用 RFdiffusion 从头设计蛋白质结构与功能。《自然》2023年7月；620:1089–100. DOI：10.1038/ s41586 - 023 - 06415 - 8. 可见于：http : / /dx. doi . org / 10 . 1038 / s41586 - 023 - 06415-8 10. Hartman E、Wallin J、Malmstr¨om J 和 Olsson J. 通过费曼-卡茨引导实现可控蛋白质设计。2025年。 DOI：10.48550/arXiv.2511.09216. 可见于：https: //arxiv.org/abs/2511.09216 11. Koh HY、Zheng Y、Yang M、Arora R、Webb GI、Pan S、Li L 和 Church GM. 人工智能驱动的蛋白质设计。《自然综述：生物工程》2025年9月；3:1034–56. DOI：10 . 1038 / s44222 - 025 - 00349 - 8. 可见于：http : / /dx. doi . org / 10 . 1038 / s44222 - 025 - 00349-8 12. Ebrahimi SB 和 Samanta D. 通过结构与化学设计改造蛋白质类疗法。《自然·通讯》2023年4月；14. DOI：10.1038/s41467-023- 38039-x. 可见于：http://dx.doi.org/10.1038/s41467-023-38039-x

13. Ingraham JB, Baranov M, Costello Z, Barber KW, Wang W, Ismail A, Frappier V, Lord DM, Ng-Thow-Hing C, Van Vlack ER, Tie S, Xue V, Cowles SC, Leung A, Rodrigues JV, Morales-Perez CL, Ayoub AM, Green R, Puentes K, Oplinger F, Panwar NV, Obermeyer F, Root AR, Beam AL, Poelwijk FJ 和 Grigoryan G。利用可编程生成模型探索蛋白质空间。《自然》2023年11月；623:1070–8。 DOI: 10.1038/s41586- 023-06728-8。可从以下网址获取：http://dx.doi.org/10.1038/s41586-023-06728-8 14. Browning Jr. JSL, Tauritz DR 和 Beckmann J。模拟分子进化的进化算法：一项新领域提案。《生物信息学简报》2024年7月；25。 DOI: 10.1093/bib/bbae360。可从以下网址获取：http://dx.doi.org/10.1093/bib/bbae360 15. Li Q, Vlachos EN 和 Bryant P。基于蛋白质序列信息设计线性和环状肽结合剂。《通讯化学》2025年7月；8。 DOI: 10.1038/s42004-025-01601-3。可从以下网址获取：http://dx.doi.org/10.1038/s42004-025-01601-3 16. Chaves Carvalho F, Mariano D, Bastos L, Abreu AP de, Lemos RP, Ara´ujo SC, Santos LM dos 和 Melo-Minardi RC de。利用 AlphaFold 和 Rosetta，通过进化与结构相结合的方法进行人工智能引导的肽抑制剂设计。《科学报告》2025年11月；15。DOI: 10.1038/s41598-025-28061-y。可从以下网址获取：http://dx.doi.org/10.1038/s41598-025-28061-y 17. Schneider G, Schr¨odl W, Wallukat G, M¨uller J, Nissen E, R¨onspeck W, Wrede P 和 Kunze R。通过人工神经网络和基于计算机的进化搜索设计肽。《美国国家科学院院刊》1998年10月；95:12179–84。 DOI: 10.1073/pnas.95.21.12179。可从以下网址获取：http://dx.doi.org/10.1073/pnas.95. 21.12179 18. Hong L 和 Kortemme T。通过多目标优化整合蛋白质序列设计。《公共科学图书馆计算生物学》2024年7月；20。编者：Onufriev A:e1011953。 DOI: 10.1371/journal.pcbi.1011953。可从以下网址获取：http://dx.doi.org/ 10.1371/journal.pcbi.1011953 19. Sinai S, Wang R, Whatley A, Slocum S, Locane E 和 Kelsic ED。AdaLead：一种简单而稳健的序列设计自适应贪心搜索算法。2020年。 DOI: 10.48550/ arXIV.2010.02141。可从以下网址获取：https://arxiv.org/abs/2010.02141 20. Zong C。批量贝叶斯优化下的蛋白质序列设计。2023年。 DOI: 10 . 48550/ARXIV.2303.10429。可从以下网址获取：https://arxiv.org/abs/2303.10429 21. Hie BL 和 Yang KK。用于蛋白质工程的自适应机器学习。《结构生物学当前观点》2022年2月；72:145–52。 DOI: 10.1016/j.sbi.2021.11.002。可从以下网址获取：http://dx.doi.org/10.1016/j.sbi.2021.11.002 22. Liang J, Lou Y, Yu M, Bi Y 和 Yu K。高成本优化的代理辅助进化算法综述。《膜计算杂志》2024年8月；7:108– 27。 DOI: 10.1007/s41965-024-00165-w。可从以下网址获取：http://dx.doi.org/10.1007/s41965-024-00165-w 23. Wang X, Jin Y, Schmitt S 和 Olhofer M。一种用于代理辅助进化多目标优化的自适应贝叶斯方法。《信息科学》2020年5月；519:317

31. DOI: 10.1016/j.ins.2020.01.048。获取地址：http://dx.doi.org/10.1016/j.ins.2020.01.04824. Hao H、Zhang X 和 Zhou A。进化优化和贝叶斯优化中的模型不确定性：比较分析。2024年。 DOI: 10.48550/ARXIV.2403.14413。获取地址：https://arxiv.org/abs/2403.14413

25. Freschlin CR、Fahlberg SA 和 Romero PA。利用机器学习探索蛋白质工程的适应度景观。《生物技术当前观点》2022年6月；75:102713。DOI：10.1016/j.copbio.2022.102713。获取地址：http://dx.doi.org/10.1016/j. copbio.2022.102713 26. Tran VQ、Nemeth M、Bartie LJ、Chandrasekaran SS、Fanton A、Moon HC、Hie BL、Kon-ermann S 和 Hsu PD。由蛋白质语言模型和上位性相互作用引导的快速定向进化。《科学》2026年2月。 DOI: 10.1126/science.aea1820。获取地址：http://dx.doi.org/10.1126/science.aea1820 27. Hartman E、Samsudin F、Siljehag Alencar M、Tang D、Bond PJ、Schmidtchen A 和 Malmstrom J。利用 BoPep 在肽序列空间中寻找肽结合剂。2025年1月。 DOI: 10.1101/2025.01.20.633551。获取地址：http://dx.doi. org/10.1101/2025.01.20.633551 28. Pereira JM、Xu S、Leong JM 和 Sousa S。肺炎球菌感染期间肺炎溶血素的阴阳。《免疫学前沿》2022年4月；13。 DOI：10.3389/fimmu. 2022.878244。获取地址：http://dx.doi.org/10.3389/fimmu.2022.878244 ˇ 29. Evans R、O'Neill M、Pritzel A、Antropova N、Senior A、Green T、Z´ıdek A、Bates R、Black-well S、Yim J、Ronneberger O、Bodenstein S、Zielinski M、Bridgland A、Potapenko A、Cowie A、Tunyasuvunakool K、Jain R、Clancy E、Kohli P、Jumper J 和 Hassabis D。利用 AlphaFold-Multimer 预测蛋白质复合物。2021年10月。 DOI：10.1101/2021.10.04.463034。获取地址：http://dx.doi.org/10.1101/2021.10.04.463034 30. Tang D、Kuˇcinskait˙e-Kodz˙e I、Str¨obaek J、Gueto Tettay CA、Simanaviˇcius M、Pleˇckaityt˙e M、Hultgren L、H¨akansson AP、Malmstr¨om L、Ekstr¨om S、Happonen L 和 Malmstr¨om J。肺炎溶血素表位景观的结构剖析揭示了一个横跨不同物种的胆固醇依赖性溶细胞素中和位点。2026年3月。 DOI：10.64898/2026.02. 27.708574。获取地址：http://dx.doi.org/10.64898/2026.02.27.708574 31. Amini A、Schwarting W、Soleimany A 和 Rus D。深度证据回归。2019年。 DOI：10.48550/ARXIV.1910.02600。获取地址：https://arxiv.org/abs/1910.02600 32. Passaro S、Corso G、Wohlwend J、Reveiz M、Thaler S、Somnath VR、Getz N、Portnoi T、Roy J、Stark H、Kwabi-Addo D、Beaini D、Jaakkola T 和 Barzilay R。Boltz-2：迈向准确高效的结合亲和力预测。2025年6月。 DOI: 10.1101/2025.06. 14.659707。获取地址：http://dx.doi.org/10.1101/2025.06.14.659707 33. Dauparas J、Anishchenko I、Bennett N、Bai H、Ragotte RJ、Milles LF、Wicky BIM、Courbet A、Haas RJ de、Bethel N、Leung PJY、Huddy TF、Pellock S、Tischer D、Chan F、Koep-nick B、Nguyen H、Kang A、Sankaran B、Bera AK、King NP 和 Baker D。使用 ProteinMPNN 进行稳健的深度学习蛋白质序列设计。《科学》2022年10月；378。 DOI：10.1126/science.add2187。获取地址：http://dx.doi.org/10.1126/science. add2187 34. Chaudhury S、Lyskov S 和 Gray JJ。PyRosetta：用于借助 Rosetta 实现分子建模算法的脚本式接口。《生物信息学》2010年1月；26:689–91。 DOI：10 . 1093 /bioinformatics / btq007。获取地址：http :// dx.doi .org/ 10.1093 / bioinformatics/btq007 35. Conway P、Tyka MD、DiMaio F、Konerding DE 和 Baker D。优化主链键几何可改善蛋白质能量景观建模。《蛋白质科学》2013年11月；23:47–55。 DOI：10.1002/pro.2389。获取地址：http://dx.doi.org/10.1002/pro.2389

36. Abramson J,Adler J, Dunger J,Evans R, Green T,Pritzel A, Ronneberger O,WillmoreL,Ballard AJ,BambrickJ,Bodenstein SW, Evans DA,Hung CC,O'NeillM, Reiman D,ˇTunyasuvunakoolK,Wu Z, Zemgulyt˙eA, ArvanitiE,BeattieC,BertolliO, BridglandA,Cherepanov A,CongreveM, Cowen-Rivers AI, Cowie A,Figurnov M, Fuchs FB, GladmanH, Jain R, Khan YA, Low CMR, Perlin K, PotapenkoA, Savy P, Singh S, SteculaA,ˇThillaisundaram A, Tong C, Yakneen S, Zhong ED, Zielinski M,Z´ıdek A, Bapst V, Kohli P, Jaderberg M, Hassabis D, and JumperJM. 基于 AlphaFold 3 对生物分子相互作用进行准确的结构预测。《自然》2024年5月；630：493–500。 DOI：10.1038/s41586-024-07487-w。参见：http://dx.doi.org/10.1038/s41586-024-07487-w

8 补充信息

以下是肺炎溶血素结合剂设计任务中所使用的 BoGA 配置示例：

```
调度计划 = [{ 'acquisition': 'expected_
improvement', 'generations': 100, 'm_select': 10, '
k_pool': 5000, 'acquisition_kwargs': { 'top_f
raction': 50, 'selection_method': 'uniform', } },]
```

```
boga = BoGA( 目标_结构_路径=目标_结构_路径， 初始_
序列=起始_序列， 最小_序列_长度=8， 最大_序列_长度=25,
n_初始=100， 突变_率=0.05， 代理_模型_参数={ '模型_类
型': '深度_证据', '网络_类型': 'bigru', 'n_试验': 50, 'n_划
分': 5, '超参数优化_间隔':200, # 永远不会发生 }, # 评分配
置 scoring_参数={ '待_纳入_的评分': ['界面_dG', 'ipt
m',
```

’在_结合_位点’, ’距离_评分’, ’肽_pae’, ’rosetta_评分’] ,’结合_位点_残基_索引’: 结合_位点_残基， ’所需_n_接触_残基’: 5, ’n_作业’: 12 }, docker_参数={ ’模型’: [’boltz’] ,’输出_目录’: ... ’模型数量_’: 3, ’循环_次数’: 5, ’扩散_样本’: 3, ’循环_步骤’: 5, ’gpu_编号’: ... }, 目标_函数=bopep_目标_v1, 目标_函数_参数={}, 嵌入_方法=’esm’, 嵌入_批次_大小=64， pca_主成分_数量=100， n_验证=0.2， 日志_目录 = ...) boga.run(调度计划=调度计划)