

# AI驱动的光刻工艺参数闭环优化系统 - Demo

## 1. 项目核心思路

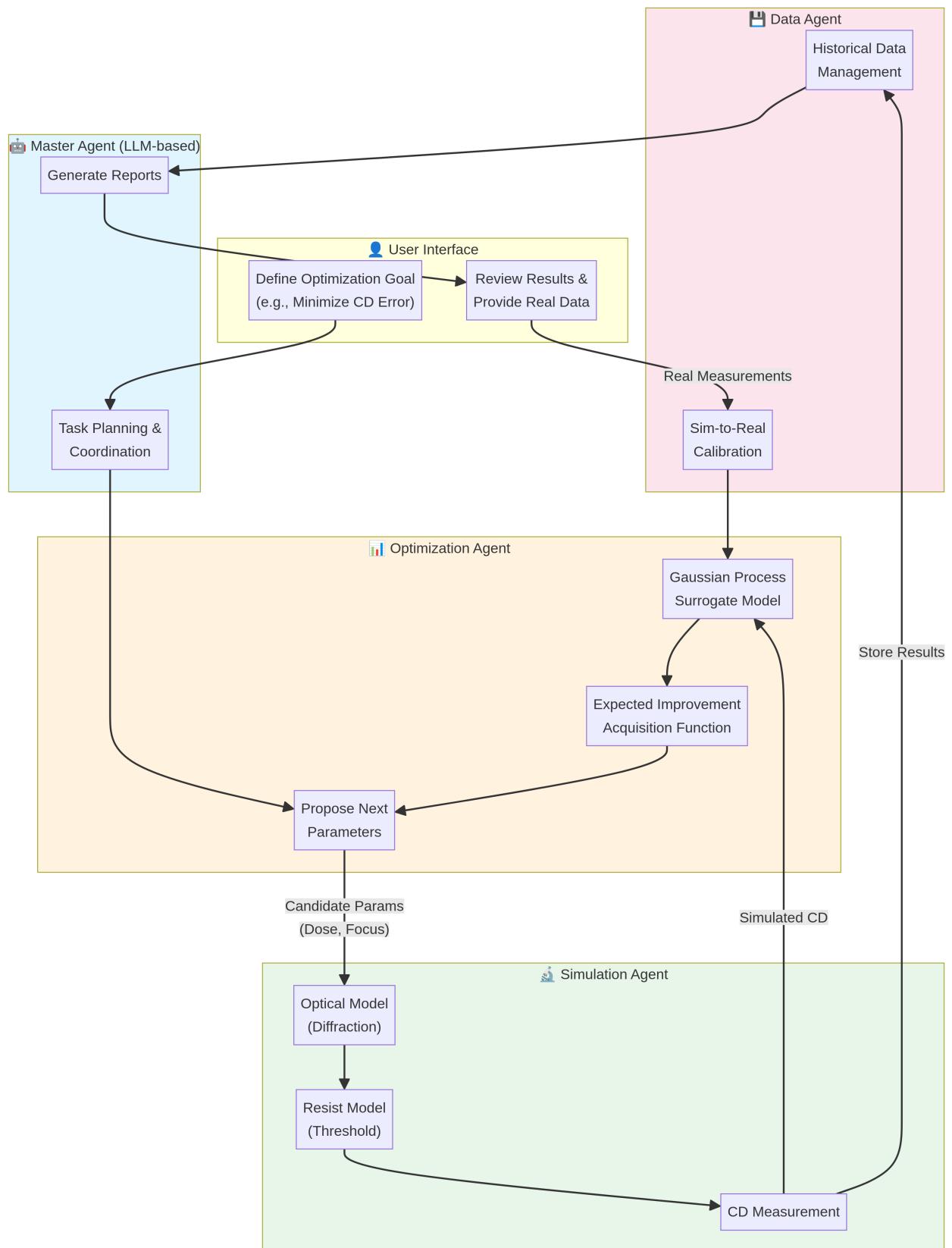
本项目旨在演示一个AI驱动的闭环系统，用于自动优化光刻工艺参数。其核心思路是替代传统依赖工程师经验的人工调参过程，通过“虚拟实验+智能决策”的方式，高效、系统地找到最优工艺配方。

**闭环流程:** 1. **定义目标:** 用户设定优化目标（例如，最小化关键尺寸CD偏差）。2. **智能提议:** AI优化智能体（基于贝叶斯优化）根据现有数据，提出一组最有希望改善结果的候选工艺参数。3. **虚拟实验:** 可微分光刻仿真器接收这组参数，模拟光刻过程，并预测出对应的CD结果。4. **评估与更新:** 将仿真结果与目标进行比较，计算偏差。AI智能体根据新的“（参数，结果）”数据点，更新其内部的代理模型。5. **循环迭代:** 重复步骤2-4，直到满足收敛条件（如CD偏差足够小或达到预设实验次数）。

**核心价值:** \* **效率:** 在高维参数空间中进行智能搜索，避免盲目试错，显著减少实验次数。\* **数据驱动:** 所有决策均基于仿真数据，过程可追溯，结果可复现。\* **自动化:** 实现从参数搜索到评估的闭环运行，解放人力。

## 2. 技术架构与创新点

本Demo聚焦于展示上述闭环的核心技术路径，其架构由两大关键模块组成：



## 模块一：可微分光刻仿真器 ( `src/litho_sim.py` )

这是闭环系统的“虚拟实验平台”。本项目借鉴**TorchLitho** [1] 和 **LithoBench** [2] 的思想，实现了一个简化但核心功能完备的可微分仿真器。

- **技术实现:** 使用PyTorch构建。它模拟一个简化的光学模型（例如，高斯模糊模拟衍射）和光刻胶模型（Sigmoid函数模拟显影阈值）。
- **创新点 (可微分):** 整个仿真过程是完全可微分的。这意味着我们可以计算出**输出CD对输入参数 (如曝光剂量Dose、离焦Focus) 的梯度**。虽然本Demo主要使用贝叶斯优化（一种黑盒优化方法），但可微分为未来采用更高效的基于梯度的优化算法（如Adam）提供了可能。

## 模块二：贝叶斯优化智能体 ( `src/bo_agent.py` )

这是闭环系统的“智能决策大脑”。它负责在不确定性中做出最明智的参数选择。

- **技术实现:** 基于**BoTorch** [3] 库实现。它内部维护一个高斯过程 (GP) 代理模型，用于拟合“工艺参数 -> CD偏差”这个未知的黑盒函数。
- **决策逻辑:** 采用**预期改善(Expected Improvement, EI)**作为采集函数。EI能够很好地平衡“在当前已知最优解附近进行探索 (Exploitation)”和“在不确定性高的区域进行探索 (Exploration)”，从而避免陷入局部最优。

## 3. 如何运行Demo

### 3.1. 环境搭建

```
# 建议使用conda创建独立环境
conda create -n litho_demo python=3.8
conda activate litho_demo

# 安装核心依赖
pip install torch botorch gpytorch numpy matplotlib
```

### 3.2. 运行闭环优化

```
python src/main.py
```

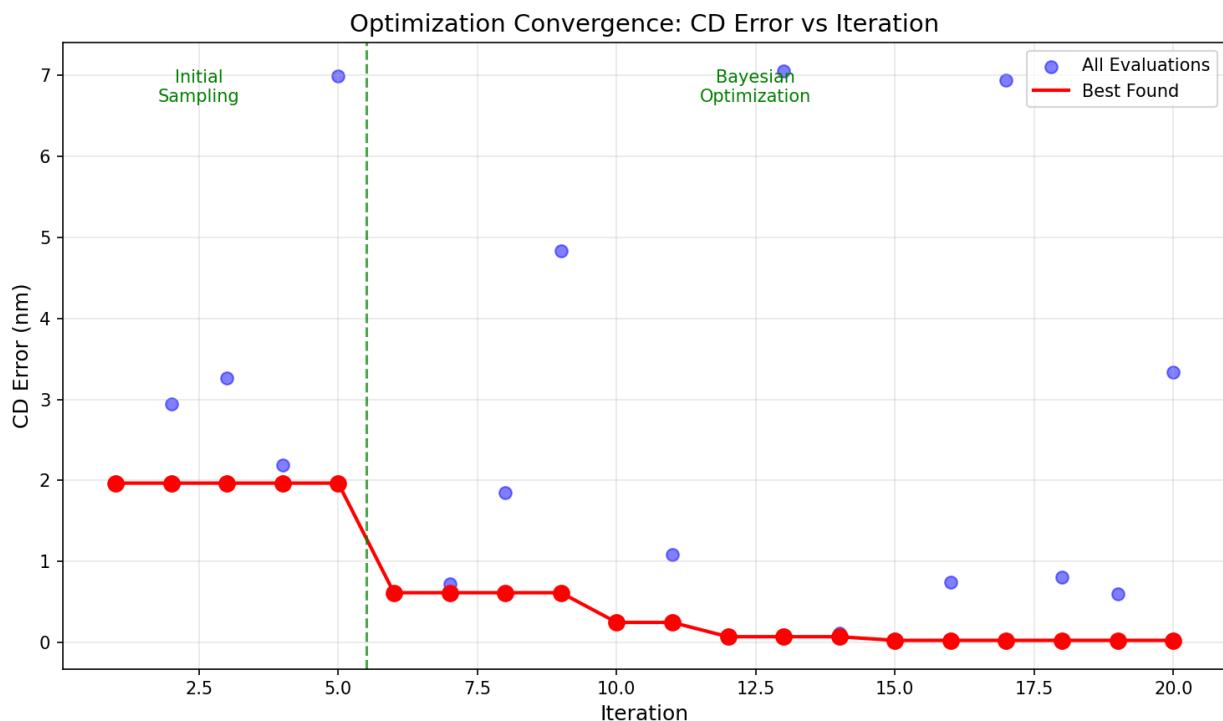
脚本将执行以下操作：1. 初始化光刻仿真器和一个“真实”的目标CD值。2. 初始化贝叶斯优化智能体，定义参数 (Dose, Focus) 的搜索范围。3. 运行15轮闭环优化迭代。4. 在每一轮中，打印出AI提议的参数、仿真得到的CD以及当前的最佳结果。5. 优化结束后，在 `results/` 目录下生

成两张图： \* convergence\_plot.png : 展示CD偏差随迭代次数的收敛过程。 \* parameter\_search.png : 展示AI在二维参数空间中的探索路径。

## 4. Demo运行结果

### 优化收敛曲线

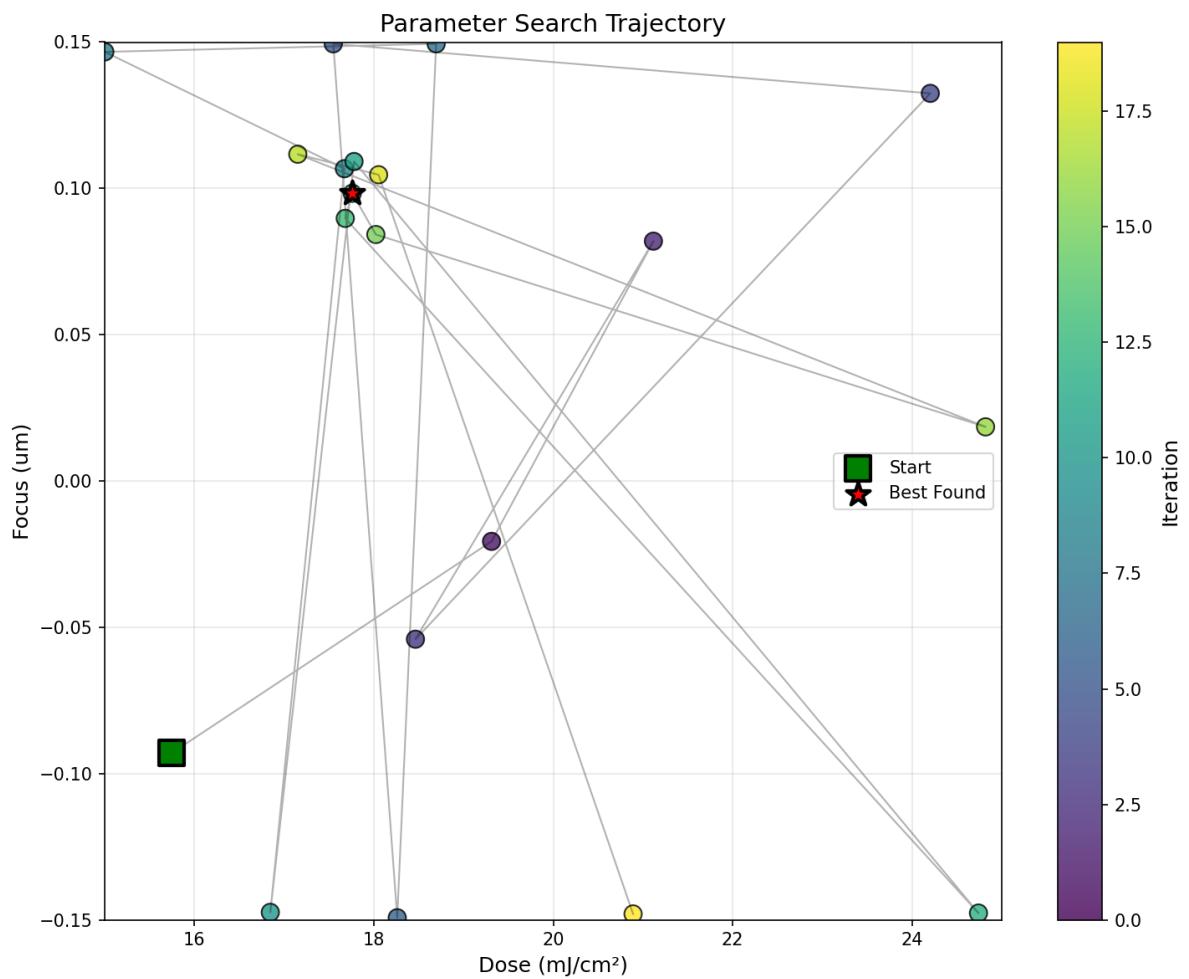
下图展示了CD偏差随迭代次数的变化。可以看到，在初始采样阶段（前5次）建立代理模型后，贝叶斯优化阶段快速收敛到最优解。



**关键发现:** - 初始最佳CD偏差: ~1.97 nm - 最终CD偏差: **0.08 nm** (仅6次BO迭代后) - 总实验次数: 20次

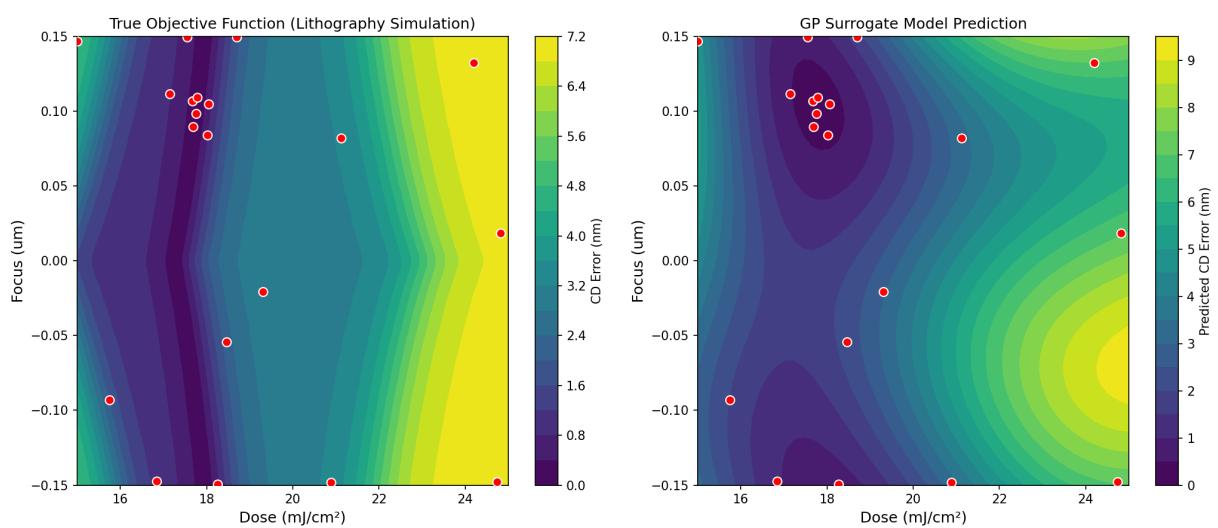
### 参数搜索轨迹

下图展示了AI在二维参数空间 (Dose × Focus) 中的探索路径。颜色从深到浅表示迭代顺序。



## GP代理模型可视化

下图对比了真实目标函数（左）和GP代理模型的预测（右）。红点为采样点。



## 5. 项目结构

```
ai_litho_demo/
├── README.md                      # 项目说明文档
├── src/
│   ├── litho_sim.py                # 可微分光刻仿真器
│   ├── bo_agent.py                 # 贝叶斯优化智能体
│   └── main.py                     # 闭环优化主程序
└── data/
    └── (预留用于真实数据集)
└── results/
    ├── optimization_history.json   # 优化历史数据
    ├── convergence_plot.png       # 收敛曲线图
    ├── parameter_search.png       # 搜索轨迹图
    └── surrogate_model.png        # 代理模型图
└── docs/
    └── architecture.png           # 系统架构图
```

## 6. 参考文献

- [1] Chen, G., et al. (2024). Open-Source Differentiable Lithography Imaging Framework. SPIE Advanced Lithography + Patterning.
- [2] Zheng, S., et al. (2023). LithoBench: Benchmarking AI Computational Lithography. NeurIPS.
- [3] Balandat, M., et al. (2020). BoTorch: A Framework for Efficient Monte-Carlo Bayesian Optimization. NeurIPS.