Systems & Control Transactions

Original Research Article
Peer Reviewed Conference Proceeding
ESCAPE 36 - European Symposium on Computer Aided Process Engineering
Sheffield, UK. 21-24 June 2026

PSE PRESS

Machine Learning and Adaptive Sampling Powered Feasible Path Algorithm for Black-box Optimization

Zixuan Zhang[ab], Xiaowei Song[ab], Jiaming Li[ab], Yujiao Zeng[a], Yaling Nie[a], Min Zhu[a], Dongyun Lu[a], Yibo Zhang[a], Xin Xiao[ad]*, Jie Li[cd§]

[a] Chinese Academy of Science, Institute of Process Engineering, Beijing, China
[b] University of Chinese Academy of Science, School of Chemical Engineering, Beijing, China
[c] The University of Manchester, Department of Chemical Engineering, Manchester, United Kingdom
[d] China Institute for Innovation and Development Strategy, Work Committee on Carbon Neutral Intelligent Industry Innovation, Beijing, China
* Corresponding Author: xxiao@ipe.ac.cn.
§ Corresponding Author: jie.li-2@manchester.ac.uk.

ABSTRACT

Black-box optimization (BBO) deals with problems involving functions that are either unknown, imprecise, or costly to evaluate. Current BBO methods encounter multiple challenges, such as high computational demands from excessive function evaluations, difficulties in handling complex constraints, lack of theoretical convergence guarantees, and unstable performance due to significant variations in solution quality. This work presents a machine learning-powered feasible path (MLFP) framework for general BBO problems involving complex constraints. An adaptive sampling strategy is first proposed to explore optimal regions and pre-filter potentially infeasible points, thereby reducing the number of evaluations. Machine learning algorithms are utilized to build surrogates for black-box functions. The feasible path algorithm is integrated to accelerate theoretical convergence by updating only independent variables instead of all variables. Computational experiments demonstrate that MLFP can rapidly and robustly converge near the KKT point, even when training surrogates with small datasets. Compared to state-of-the-art BBO algorithms, MLFP stably delivers equivalent or superior solutions with fewer evaluations across benchmark examples.

Keywords: Optimization, Machine Learning, Black-box, Adaptive Sampling, Feasible Path Algorithm, Surrogate Model

INTRODUCTION

Black-box optimization (BBO) refers to a class of optimization problems where the algebraic forms of the objective function and/or constraints are unknown, imprecise, or prohibitively expensive to compute. This characteristic is ubiquitous in engineering and scientific fields, particularly in chemical engineering, where many complex systems cannot be fully captured by mechanistic models. For instance, multi-step catalytic reaction networks, molecular structure–activity relationships (SARs) linking polymer architecture to material performance, and high-fidelity simulations (e.g., density functional theory calculations, computational fluid dynamics for reactor design, and rigorous distillation modeling) all exhibit black-box properties [1-5]. These systems are foundational to chemical design and manufacturing, from catalyst development and material synthesis to plant-wide operation tuning, and optimizing them relies heavily on specialized BBO approaches. Advancing BBO methods is therefore critical to reducing development cycles, cutting costs, and unlocking innovative engineering solutions.

Despite its importance, BBO faces inherent challenges rooted in the absence of gradient information for black-box functions, which prevents the use of classical gradient-based optimization. Existing methods can be broadly categorized into three classes, each with distinct limitations. Stochastic or heuristic algorithms (e.g., PSO, GA, SA) offer global search capabilities but lack theoretical convergence guarantees and require extensive function evaluations, making them impractical for expensive

systems. Deterministic partition methods (e.g., DIRECT, pattern search) avoid randomness but suffer from the curse of dimensionality and high computational costs. For example, a modified DIRECT algorithm for LNG process optimization required 20, 000 evaluations [6]. Surrogate model-based methods, the most widely adopted class, reduce black-box calls by using cheap approximations (e.g., neural networks, Gaussian processes) but introduce new issues: implicit filtering is limited to bound-constrained problems, ReLU NN-based frameworks become computationally intractable for complex architectures [7], ALAMO performs poorly for high-dimensional inputs [8], trust-region filter (TRF) methods suffer from initial value sensitivity [9], and Bayesian optimization (BO) struggles with complex equality constraints and exhibits significant result stochasticity (variations up to 300% across runs) [10]. Additionally, many surrogate-based methods overlook the computational burden of sampling and training.

To address these limitations, this study proposes a novel machine learning-powered feasible path (MLFP) framework for general BBO problems involving complex equality and inequality constraints. The framework integrates three core components to balance sample efficiency, approximation accuracy, and convergence speed. First, an adaptive sampling strategy uses support vector machines (SVM) to filter infeasible points and focus on potential optimal regions, ensuring informative samples even with limited data. Second, machine learning models (e.g., multi-layer perceptrons with Swish activation) are employed as surrogates, leveraging their strong nonlinear approximation capabilities and twice-differentiable properties to enable gradient-based optimization. Third, the feasible path algorithm is integrated to update only independent variables instead of the entire variable set, avoiding complex nonlinear terms from full-space formulations and accelerating convergence by retaining computation graphs for efficient derivative calculation.

This work validates MLFP on 2 industrial processes (Williams-Otto process and extractive distillation process). The results demonstrate that MLFP outperforms state-of-the-art algorithms by stably achieving equivalent or better solutions with fewer evaluations.

PROBLEM STATEMENT

The general form of a BBO problem to be solved in this work is presented below, which is a nonlinear programming (NLP) problem.

$$\begin{aligned} \min_{\mathbf{x}\in\mathbb{R}^n} \quad & f(\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{h}(\mathbf{x}) = 0 \\ & \mathbf{g}(\mathbf{x}) \le 0 \\ & \mathbf{x}^{\boldsymbol{lb}} \le \mathbf{x} \le \mathbf{x}^{\boldsymbol{ub}} \end{aligned} \tag{1}$$

where the function $f: \mathbb{R}^n \to \mathbb{R}$ is the objective function, $\mathbf{h}: \mathbb{R}^n \to \mathbb{R}^{p_E}$ is the equality constraints and $\mathbf{g}: \mathbb{R}^n \to \mathbb{R}^{p_I}$ is the inequality constraints excluding the bounding constraints. p_E , and p_I represent the dimensions of equations and inequalities respectively; $\mathbf{x}$ represent a vector of variables; the superscript ub represents the upper bound, and lb represents the lower bound. In this setting, the black-box function may pertain to the objective function f as well as to the constraint functions $\mathbf{h}$ (equalities) and $\mathbf{g}$ (inequalities), depending on the problem structure.

Without loss of generality, we divide the equality constraints $\mathbf{h}(\mathbf{x}) = 0$ into $\mathbf{h_1}(\mathbf{x}) = 0$ and $\mathbf{h_2}(\mathbf{x}) = 0$ where $\mathbf{h_1}(\mathbf{x}) = 0$ containing all black-box constraints and $\mathbf{h_2}(\mathbf{x}) = 0$ without containing any black-box functions. If any inequality constraints $\mathbf{g_j}(\mathbf{x}) \le 0$ containing black-box functions, we can introduce auxiliary variables $\mathbf{v}$ and let $\mathbf{v} = \mathbf{g_j}(\mathbf{x})$. We then add $\mathbf{v} = \mathbf{g_j}(\mathbf{x})$ into $\mathbf{h_1}(\mathbf{x}) = 0$ and remove $\mathbf{g_j}(\mathbf{x}) \le 0$ from the inequality constraint sets $\mathbf{g}(\mathbf{x}) \le 0$. The auxiliary variable $\mathbf{v}$ is then added into variables $\mathbf{x}$. The original BBO problem (1) can be reformatted into problem (2) below,

$$\begin{aligned} \min_{\mathbf{x}\in\mathbb{R}^n} \quad & f(\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{h_1}(\mathbf{x}) = 0 \\ & \mathbf{h_2}(\mathbf{x}) = 0 \\ & \mathbf{g}(\mathbf{x}) \le 0 \\ & \mathbf{x}^{\boldsymbol{lb}} \le \mathbf{x} \le \mathbf{x}^{\boldsymbol{ub}} \end{aligned} \tag{2}$$

where $\mathbf{h_1}: \mathbb{R}^n \to \mathbb{R}^{p_{E1}}$ and $\mathbf{h_2}: \mathbb{R}^n \to \mathbb{R}^{p_{E2}}$ are the equality constraints and $\mathbf{g}: \mathbb{R}^n \to \mathbb{R}^{p_I}$ is the inequality constraints excluding the bounding constraints. $p_{E1}, p_{E2}$and p_I represent the dimensions of equations and inequalities respectively. In this setting, the black-box function may pertain to the objective function f as well as to the constraint functions $\mathbf{h_1}$ (equalities). We assume that all functions (e.g., $f, \mathbf{h_2}$ and $\mathbf{g}$) that are not the black-box functions, are twice continuously differentiable. The black-box functions $\mathbf{h_1}$ or f are unknown and their derivatives are not available; however, we assume that they can be evaluated at a given point and approximated by a twice continuously differentiable function.

METHOD

As $f(\mathbf{x})$ and/or $\mathbf{h_1}(\mathbf{x}) = 0$ in problem (2) contain black-box functions that are unknown or expensive for evaluation, we need to develop surrogate models for these black-box functions. To do this, we can divide all variables in (2) into independent variables (denoted as $\mathbf{x}_I$) and dependent variables (denoted as $\mathbf{x}_D$) where $\mathbf{x}_D$ has the same dimension as $\mathbf{h_1}$. Then, problem (2) can be transformed into problem (3).

Input: Surrogate model $\boldsymbol{s}(\cdot)$
Input: $\boldsymbol{\lambda}_{qp}^0 \leftarrow 0$, $\boldsymbol{\mu}_{qp}^0 \leftarrow 0$, $\boldsymbol{\rho}^0 \leftarrow 0$, $\boldsymbol{\nu}^0 \leftarrow 0$
Input: $\eta \leftarrow 0.1$, $tol \leftarrow 10^{-6}$, line search multiple $\beta \leftarrow 0.618$, line search maximum iteration $n_{max} \leftarrow 10$
Input: Feasible initial point $\boldsymbol{x}^0$
Output: Solution $\boldsymbol{x}^*$

$k \leftarrow 0$, $acc_{inf} \leftarrow \infty$, $acc_{step} \leftarrow \infty$, $acc_{opt} \leftarrow \infty$
while $acc_{inf} > tol$ or ($acc_{step} > tol$ and $acc_{opt} > tol$) **do**
  $k \leftarrow k + 1$
  $\boldsymbol{y}^k \leftarrow \boldsymbol{s}(\boldsymbol{x}^k)$
  Calculate $\nabla_x \boldsymbol{y}^k$ and $\nabla_x^2 \boldsymbol{y}^k$ based on Section 3
  Calculate $\nabla_x f$, $\nabla_x \boldsymbol{g}^k$, $\nabla_x \boldsymbol{h}^k$ and $\boldsymbol{H}^k$
  if $\boldsymbol{H}^k$ is positive definite **then**
    $\boldsymbol{B}^k \leftarrow \boldsymbol{H}^k$
  else
    $\boldsymbol{B}^k \leftarrow \boldsymbol{H}^k + \boldsymbol{E}^k$
  end if
  Solve (QP) subproblem, then obtain $\boldsymbol{d}^k$, $\boldsymbol{\lambda}_{qp}^k$ and $\boldsymbol{\mu}_{qp}^k$
  $\boldsymbol{\rho}^k \leftarrow \max(|\boldsymbol{\lambda}_{qp}^k|, \frac{|\boldsymbol{\lambda}_{qp}^k| + \boldsymbol{\rho}^{k-1}}{2})$
  $\boldsymbol{\nu}^k \leftarrow \max(|\boldsymbol{\mu}_{qp}^k|, \frac{|\boldsymbol{\mu}_{qp}^k| + \boldsymbol{\nu}^{k-1}}{2})$
  Calculate $\phi_1(\boldsymbol{x}^k, \boldsymbol{\lambda}_{qp}^k, \boldsymbol{\mu}_{qp}^k)$ and $D(\phi_1(\boldsymbol{x}^k, \boldsymbol{\lambda}_{qp}^k, \boldsymbol{\mu}_{qp}^k); \boldsymbol{d}^k)$
  $n \leftarrow 0$, $\alpha \leftarrow 1$
  while $n \leq n_{max}$ **do**
    $n \leftarrow n + 1$
    $\boldsymbol{x}^{k+1} \leftarrow \boldsymbol{x}^k + \alpha \boldsymbol{d}^k$
    if $\phi_1(\boldsymbol{x}^k + \alpha \boldsymbol{d}^k, \boldsymbol{\lambda}_{qp}^k, \boldsymbol{\mu}_{qp}^k) - \phi_1(\boldsymbol{x}^k, \boldsymbol{\lambda}_{qp}^k, \boldsymbol{\mu}_{qp}^k) < \alpha\eta D(\phi_1(\boldsymbol{x}^k, \boldsymbol{\lambda}_{qp}^k, \boldsymbol{\mu}_{qp}^k); \boldsymbol{d}^k)$ **then**
      break
    else
      $\alpha \leftarrow \beta\alpha$
    end if
  end while
  Update acc_{inf}, acc_{step}, acc_{opt} based on appropriate criteria
end while
return $\boldsymbol{x}^k$

Algorithm 1: Pseudocode for feasible path algorithm.

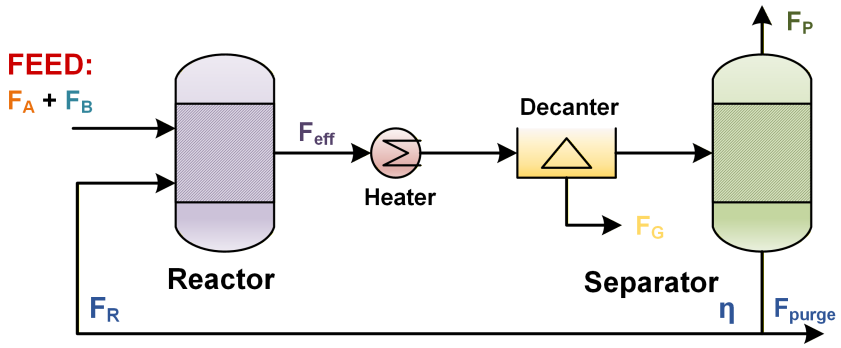

Figure 1. Flow diagram for WO process.

$$\begin{aligned} \min_{\boldsymbol{x} \in \mathbb{R}^n} \quad & f(\boldsymbol{x}_I, \boldsymbol{x}_D) \\ \text{s.t.} \quad & \boldsymbol{h}_1(\boldsymbol{x}_I, \boldsymbol{x}_D) = 0 \\ & \boldsymbol{h}_2(\boldsymbol{x}_I, \boldsymbol{x}_D) = 0 \\ & \boldsymbol{g}(\boldsymbol{x}_I, \boldsymbol{x}_D) \leq 0 \\ & \boldsymbol{x}_I^{\boldsymbol{lb}} \leq \boldsymbol{x}_I \leq \boldsymbol{x}_I^{\boldsymbol{ub}} \\ & \boldsymbol{x}_D^{\boldsymbol{lb}} \leq \boldsymbol{x}_D \leq \boldsymbol{x}_D^{\boldsymbol{ub}} \end{aligned} \tag{3}$$

Adaptive sampling strategy

Adaptive sampling is pivotal for surrogate-assisted BBO, as it generates high-quality data, builds accurate models, and reduces function evaluations. Most existing strategies are iteratively integrated with optimization, selecting the next sampling point based on current results, which is effective for coupled sampling-optimization algorithms but unsuitable for ones that construct offline surrogate before optimization. To address this, we propose a new adaptive sampling strategy to build a global dataset for pre-optimization surrogate training, focusing on capturing details near optimal regions and variable bounds.

In this work we proposed a adaptive sampling strategy that focus on potential optimal regions and onstraint

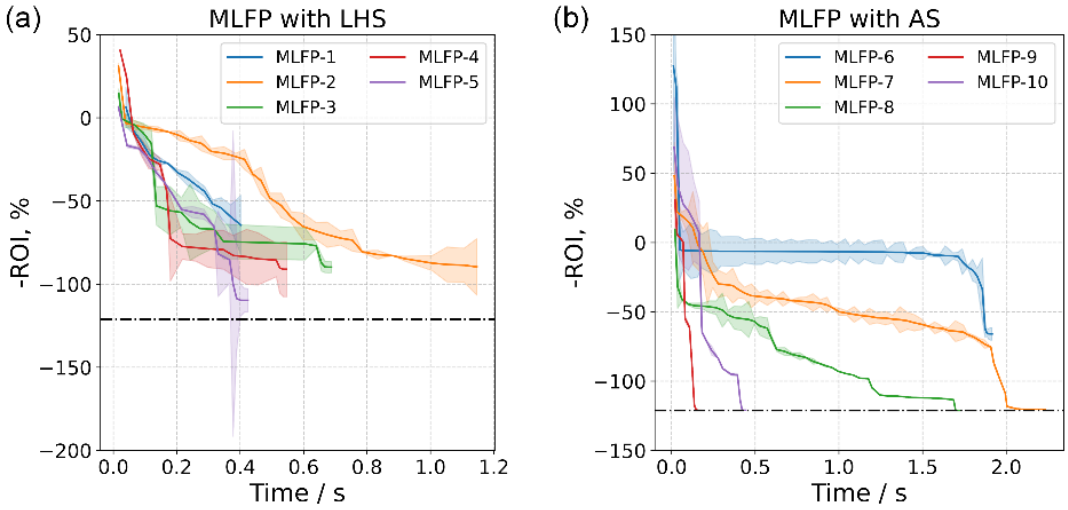

Figure 2. Convergence curves for WO process with error band of MLFP 1-10.

boundaries. The strategy proceeds in three steps:

1. Slightly expand the bounds of each independent variable (e.g., ±5% of original) and perform initial Latin hypercube sampling (LHS) to collect a small sample set (e.g., 10% of total required). Evaluate black-box functions at these samples to obtain dependent variable values, understand function behaviors, and identify promising regions.

2. Label samples as feasible (satisfying constraints like $g(x_I, x_D), h_2(x_I, x_D)$ and bound constraints for variables) or infeasible (1 or 0), then train a support vector machine (SVM) classifier. To mitigate inaccuracy from limited initial samples, extract support vectors from the trained SVM and retrain the classifier by treating all support vectors as feasible, which includes near-boundary infeasible points to ensure precise boundary characterization.

3. Establish a sub-region around the best feasible point, conduct LHS within this area (e.g., 10% of total samples per iteration), and use the trained SVM to filter out infeasible points (avoiding wasted black-box evaluations). Update the best feasible point and repeat until the required sample size is reached.

Feasible path optimization algorithm

After the surrogate models for $\mathbf{h}_1(\mathbf{x}) = 0$ and/or the objective function are developed, then $\mathbf{h}_1(\mathbf{x}) = 0$ and/or the objective function in problem (3) are replaced by the surrogates denoted as $\boldsymbol{x}_D = \mathbf{s}(\mathbf{x}_I)$ and/or $fs(\mathbf{x}_I)$ and problem (4) below is obtained. We also assume that $\mathbf{s}(\mathbf{x}_I)$ and/or $fs(\mathbf{x}_I), \mathbf{h}_2$ and $\mathbf{g}$ are all twice continuously differentiable. The remaining constraints $\mathbf{h}_2$ and $\mathbf{g}$, and the objective function f can be calculated without calling the black-box.

$$\begin{aligned} \underset{\mathbf{x}\in\mathbb{R}^n}{min} \quad & f(\mathbf{x}_I, \mathbf{x}_D) \text{ or } fs(\mathbf{x}_I) \\ \text{s.t.} \quad & \mathbf{x}_D = \mathbf{s}(\mathbf{x}_I) \\ & \mathbf{h}_2(\mathbf{x}_I, \mathbf{x}_D) = 0 \\ & \mathbf{g}(\mathbf{x}_I, \mathbf{x}_D) \le 0 \\ & \mathbf{x}_I^{lb} \le \mathbf{x}_I \le \mathbf{x}_I^{ub} \\ & \mathbf{x}_D^{lb} \le \mathbf{x}_D \le \mathbf{x}_D^{ub} \end{aligned} \tag{4}$$

To solve problem (4) effectively, we employ the feasible path optimization algorithms developed by Ma et al. [11]. Different from full-space formulation, which directly employs NLP algorithms to solve problem (4), feasible path optimization algorithms divide the entire problem (4) to two subproblems including a small optimization problem in the outer level and a process simulation problem in the inner level. While the small optimization problem in the outer level can be solved using the existing gradient-based optimization algorithms such as improved SQP, the process simulation problem in the inner level could be solved using solution algorithms for nonlinear equations such as a Newton-based algorithm or PTC approach [11]. In this work, the inner level only requires the use of a surrogate model to predict $\mathbf{x}_D$, which is considerably faster than rigorous simulation or black-box evaluation. Moreover, when predicting $\mathbf{x}_D$, the computation graph will be retained to compute the derivative of $\mathbf{x}_D$ with respect to $\mathbf{x}_I$, instead of calculating a large Jacobi matrix.

Detailed algorithm parameters and pseudocode are presented in the **Algorithm 1**.

In this work, the ℓ_1 merit function, defined as Eq.(5), is used to evaluate convergence criteria.

$$\phi_1(\mathbf{x}_I^k; \boldsymbol{\lambda}_{qp}^k, \boldsymbol{\mu}_{qp}^k) = f(\mathbf{x}^k) + (\boldsymbol{\rho}^k)^{\mathrm{T}} |\mathbf{h}(\mathbf{x}_I^k)| + (\boldsymbol{\nu}^k)^{\mathrm{T}} \mathbf{g}(\mathbf{x}_I^k)^- \tag{5}$$

where, $\mathbf{g}(\mathbf{x}^k)^- := \max(0, -\mathbf{g}(\mathbf{x}^k))$, and the penalty

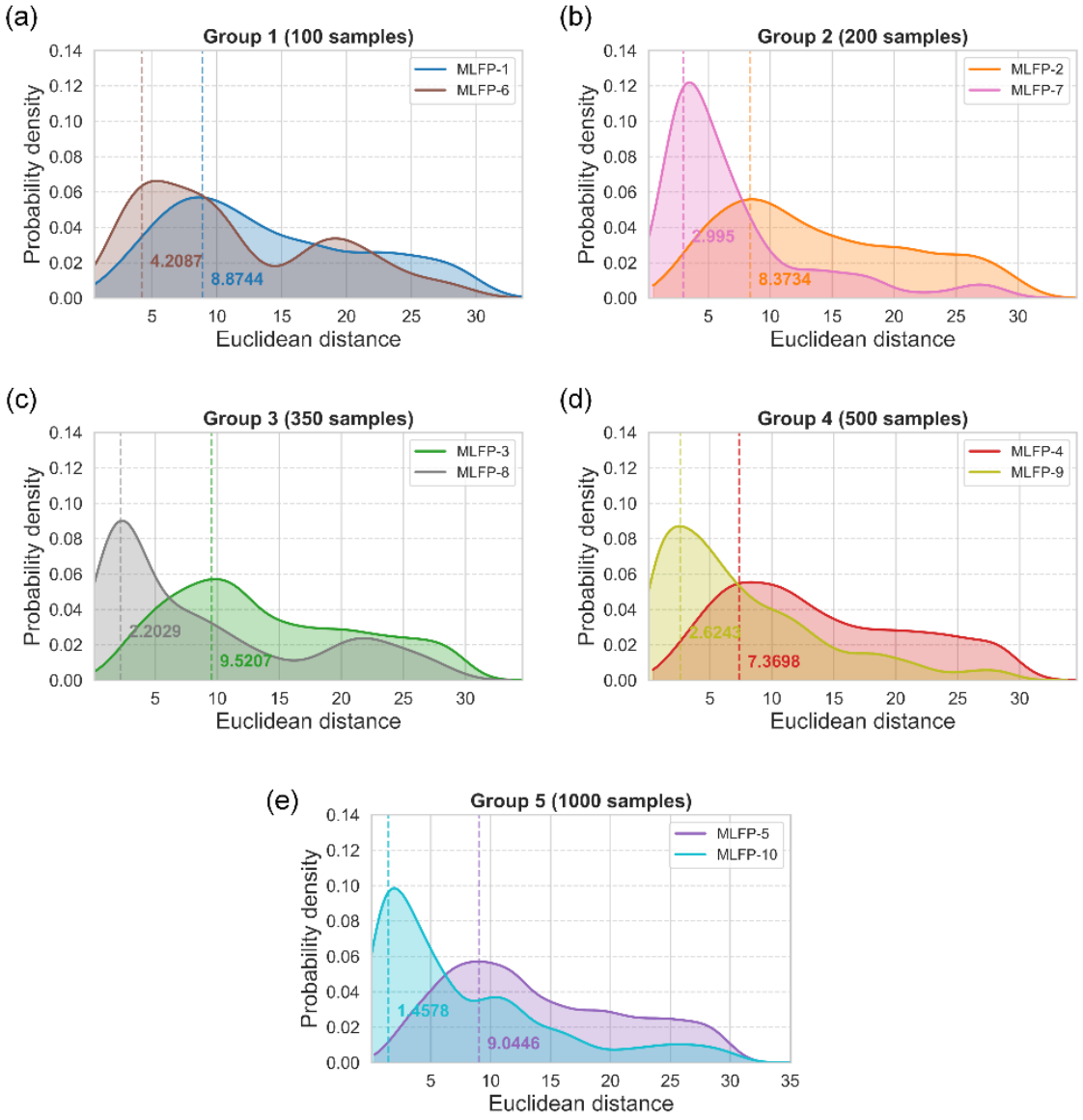

Figure 3. Probability density function of Euclidean distance from sampling points to rigorous model-based solution of WO process.

parameters are defined as follows [12]:

$$\boldsymbol{\rho}^k = max\left(|\boldsymbol{\lambda}_{qp}^k|, \frac{\boldsymbol{\rho}^{k-1}+|\boldsymbol{\lambda}_{qp}^k|}{2}\right) \tag{6}$$

$$\boldsymbol{\nu}^k = max\left(|\boldsymbol{\mu}_{qp}^k|, \frac{\boldsymbol{\nu}^{k-1}+|\boldsymbol{\mu}_{qp}^k|}{2}\right) \tag{7}$$

Given the ℓ_1 merit function being not differentiable everywhere, the Eq.(8) describes the directional derivative of $\phi_1(\mathbf{x}^k;\boldsymbol{\lambda}_{qp}^k,\boldsymbol{\mu}_{qp}^k)$ along the direction $\mathbf{d}^k$ generated by the SQP subproblem.

$$D(\phi_1(\mathbf{x}_I^k;\boldsymbol{\lambda}_{qp}^k,\boldsymbol{\mu}_{qp}^k);\mathbf{d}^k) = \nabla f(\mathbf{x}_I^k)^{\mathrm{T}}\mathbf{d}^k - (\boldsymbol{\rho}^k)^{\mathrm{T}}|\mathbf{h}(\mathbf{x}_I^k)| - (\boldsymbol{\nu}^k)^{\mathrm{T}}\mathbf{g}(\mathbf{x}_I^k)^{-} \tag{8}$$

The Armijo condition guarantees a sufficient decrease of the merit function:

$$\phi_1(\mathbf{x}_I^k + \alpha\mathbf{d}^k;\boldsymbol{\lambda}_{qp}^k,\boldsymbol{\mu}_{qp}^k) - \phi_1(\mathbf{x}_I^k;\boldsymbol{\lambda}_{qp}^k,\boldsymbol{\mu}_{qp}^k) < \alpha\eta D(\phi_1(\mathbf{x}_I^k;\boldsymbol{\lambda}_{qp}^k,\boldsymbol{\mu}_{qp}^k);\mathbf{d}^k) \tag{9}$$

Where $\eta \in (0, 0.5)$ is a hyper-parameter. In this work, we choose $\eta = 0.1$ according to Ma et al. [12] and set line search multiple $\beta = 0.618$, maximum number of line search iterations $n_{max} = 10$. However, it is possible that no step length α can satisfy the Armijo condition, regardless of how small it is. Under such circumstances, we accept the last α when the maximum number of line searches has been reached. According to the backtracking method,

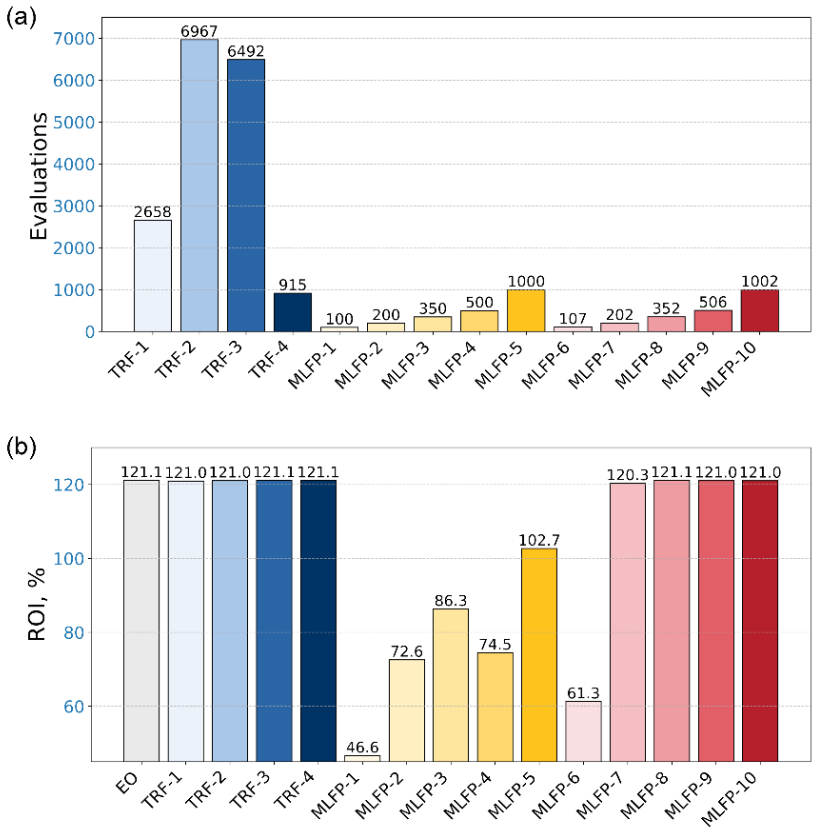

Figure 4. Comparasion for WO process from MLFPs and TRF1-4 on (a) the number of function evaluations of the black-box function and (b) objective function value.

after 10 reduction steps in the line search, the step size becomes approximately 8×10^{-3}. At this point, the step size is sufficiently small compared to the initial step size, yet it exerts little impact on convergence. Meanwhile, it remains larger than the tolerance, which ensures the iteration does not terminate prematurely before approaching the optimal value.

For practical engineering problems, due to numerical stability and other factors, the KKT conditions as criteria may be too strict, making the algorithm difficult to achieve convergence. Therefore, based on previous research, we check whether the following criteria are met after every line search:

$$acc_{inf} = \left\|\mathbf{h}\left(\mathbf{x}_I^k + \alpha \mathbf{d}^k\right)\right\|_1 + \left\|\mathbf{g}\left(\mathbf{x}_I^k + \alpha \mathbf{d}^k\right)^-\right\|_1 < tol \quad (10)$$

$$acc_{step} = \left|f\left(\mathbf{x}_I^k + \alpha \mathbf{d}^k\right) - f\left(\mathbf{x}_I^k\right)\right| < tol \quad (11)$$

$$acc_{opt} = \left\|\mathbf{d}^k\right\|_2 < tol \quad (12)$$

where, tol represents the convergence tolerance. The solution is considered optimal if Eqs.(10), (11) or Eqs.(10), (12)are satisfied. The satisfaction of Eq.(10) indicates that the optimal solution is feasible. The satisfaction of either Eq.(11) or Eq.(12) suggests that there is little potential for further decrease in the objective function.

CASE STUDY

In This work, MLPs are used to construct surrogate models with the Swish activation function. All examples are solved on a desktop with Intel(R) Core(TM) i9-14900K (3.20 GHz) processor and 128 GB of RAM running Windows 11 64-bit operating system.

Williams-Otto process

The Williams Otto (WO) process is a classic benchmark that has been widely used to evaluate optimization algorithms. The schematic diagram of the WO process is shown in **Figure 1**. The goal is to maximize the return on investment (ROI) by adjusting the reactor volume V, reaction temperature T, as well as the feed flow rates ($\mathrm{F_A}$and F_B) and material flow cycle ratio η of components A and B. The inputs of the surrogate model are V, T, as well as F_A, F_B and η, which are also independent variables in the optimization problem. The output includes the objective function value (ROI) and the product material flow rate F_P, which are dependent variables subject to boundary constraints. In this optimization problem, there are 27

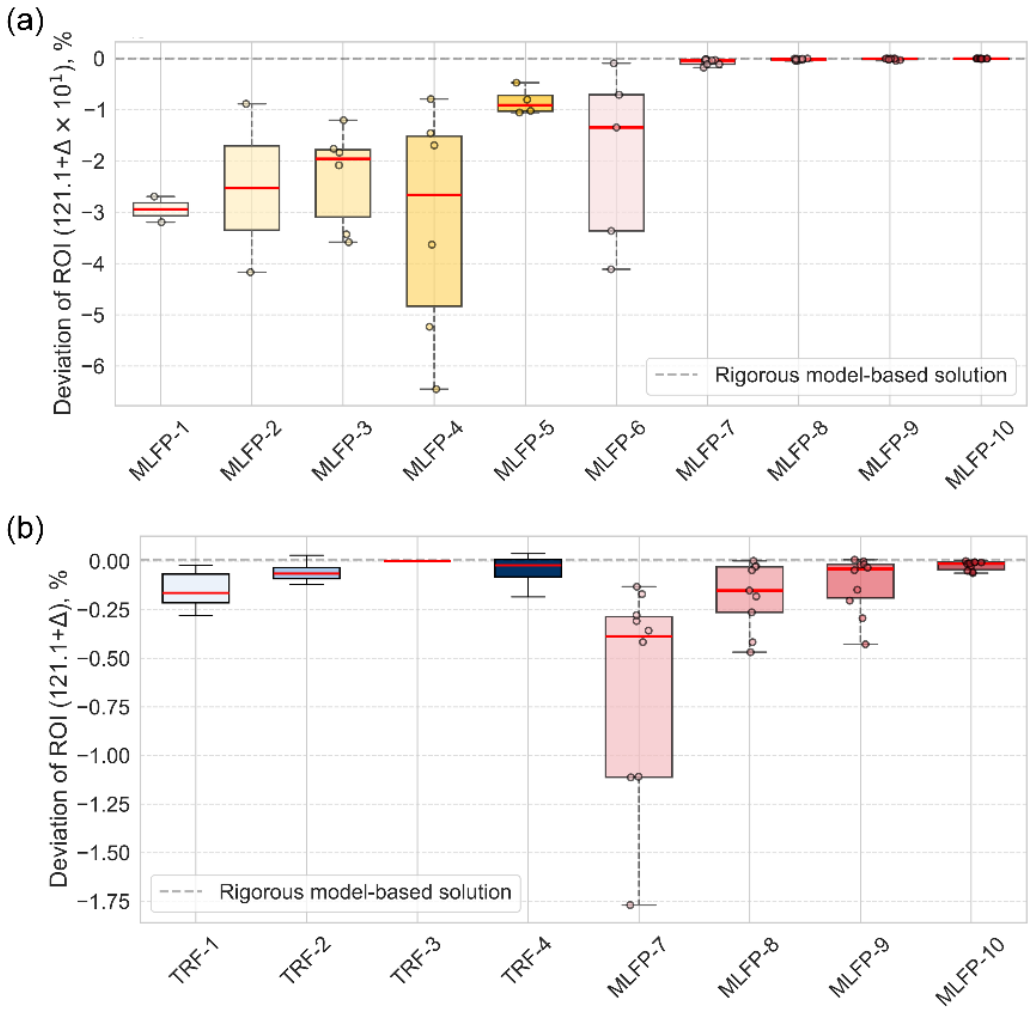

Figure 5. Boxplot of best ROI for WO process obtained from (a) MLFP1-10, and (b) TRF 1-4 and MLFP 7-10 with 10 runs.

variables and 54 constraints.

In this example, we compared the performance of TRF 1-4 and MLFP frameworks under different sampling sizes, and also compared the performance of the AS strategy and LHS strategy proposed in the MLFP framework. In the MLFP framework using the LHS strategy, we generate 100, 200, 350, 500, and 1000 valid sampling points respectively (simulation convergence successful), corresponding to MLFP 1-5; In the MLFP framework using the AS strategy, we also generate 100, 200, 350, 500, and 1000 valid sampling points, respectively, labeled as MLFP 6-10. The benchmark experimental results of the TRF method were taken from the study of Liang et al. [9]. It should be noted that the initial points used by Liang et al. [9] may differ from those of the MLFP framework.

The convergence curve of the MLFP series methods is shown in **Figure 2**. According to **Figure 2** (b), it can be seen that MLFP-7 to MLFP-10 can converge to the vicinity of the KKT point (as indicated by the black dashed line in the figure), while the results in **Figure 2** (a) show that MLFP-1 to MLFP-5 did not converge to the vicinity of the KKT point. The shaded area next to the convergence curve in **Figure 2** represents the difference between the predicted and actual values of the model. It can be seen that as the number of effective sampling points increases, the shaded area gradually narrows, indicating that the difference between the predicted and actual values continues to decrease. In other words, increasing the number of effective sampling points helps improve the prediction accuracy of the model. It can also be observed that as the iteration point approaches the true optimal value, the shaded area gradually shrinks, indicating that the samples generated by the adaptive sampling strategy are mainly concentrated near the optimal solution based on the strict model. To verify this hypothesis, this article combines samples obtained from 30 running experiments, divides the sampling results of the MLFP method into 5 groups, and assigns the same number of sampling points

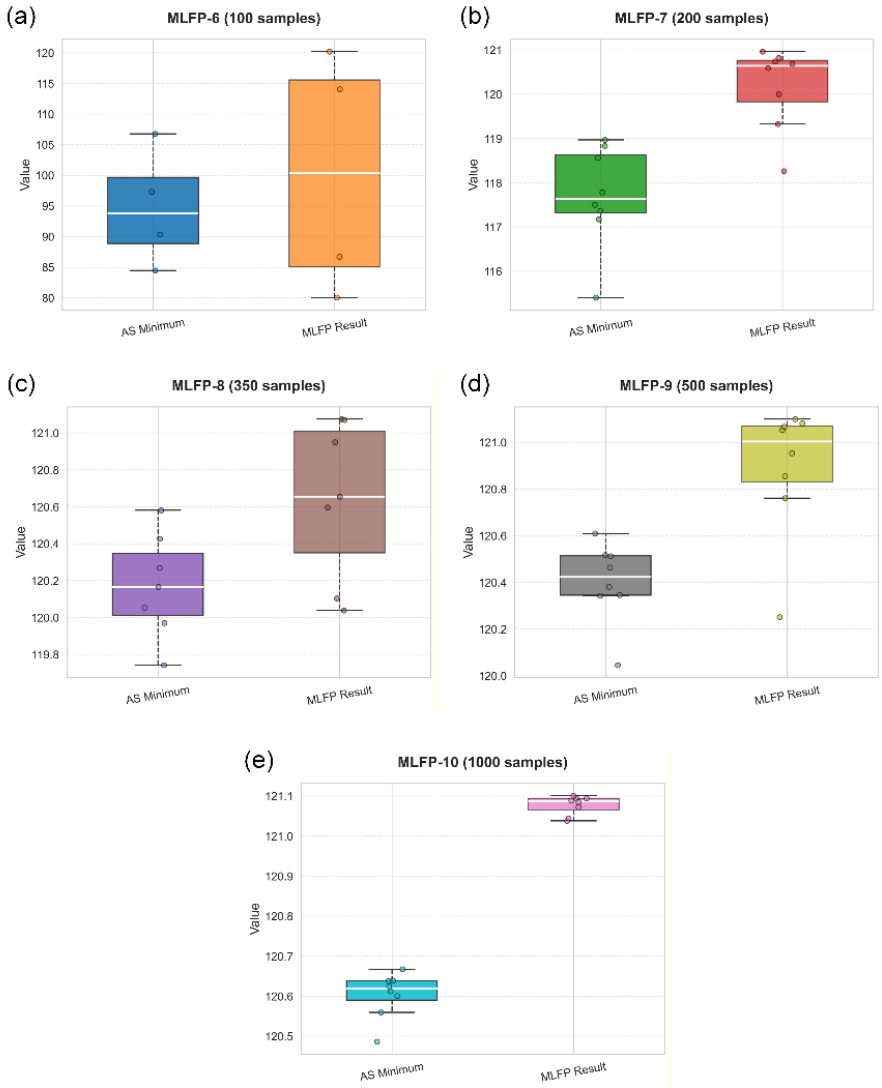

Figure 6. Comparison of the minimum objective value obtained with adaptive sampling and MLFP optimization results in WO process.

to each group. Then, LHS and AS strategies are used to conduct sampling experiments on each group, and the probability density function of the obtained samples is shown in **Figure 3**. **Figure 3** shows that under the same sampling size, the peak value of AS strategy sampling results is closer to 0 compared to LHS, indicating that its sample distribution is closer to the optimal solution based on the strict model; And as the number of sampling points increases, the peak of AS strategy sampling results will further approach 0.

The performance comparison results between TRF 1-4 and MLFP 1-10 are shown in **Figure 4**, where the dashed line represents the optimal solution obtained by using IPOPT solver to solve the strict model. Compared with TRF 1-4, MLFP 7-8 can achieve an optimal return on investment (ROI) of 121.1 and require significantly fewer function evaluations. **Figure 5** shows the optimization results of TRF 1-4 and MLFP 1-10 in 10 independent running experiments. As shown in **Figure 5** (a), under the same number of function evaluations, the stability of MLFP 6-10 is significantly better than that of MLFP 1-5, indicating that the MLFP method using the proposed adaptive sampling strategy performs better than the MLFP method using the LHS strategy; And as the number of sampling points increases, the stability of MLFP 6-10 further improves, as evidenced by the decrease in variance of the optimized ROI obtained from multiple experiments. Combining **Figure 4** (a) and **Figure 5**, it can be observed that TRF-4 and MLFP-10 require similar number of function evaluations (915 and 1002, respectively), and both can converge to the optimal solution based on the strict model; But the stability of MLFP-10 is better because its box line height, i.e. interquartile range (IQR) and whisker plot range, are much narrower than TRF-4.

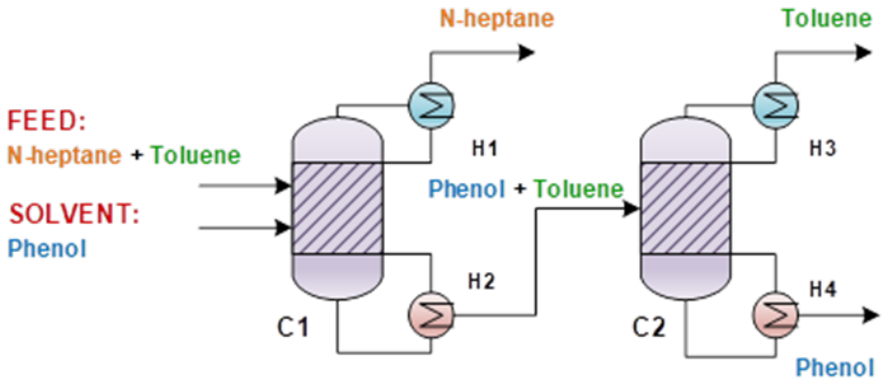

Figure 7. Process flow diagram for separation of toluene form n-heptane using solvent phenol in Ma et al.'s work [13].

In addition, we also compared the minimum target values obtained by AS and MLFP 6-10, as shown in **Figure 6**. In 10 runs, compared with the minimum value obtained by adaptive sampling, the median optimization results of MLFP 6-10 increased by 6.87%, 1.97%, 0.50%, 0.48%, and 0.39%, respectively.

Extractive distillation (ED) for toluene and n-heptane separation

Process simulation-based optimization is a typical BBO problem involving expensive function evaluations or unknown functions. In this subsection, we test an example of extractive distillation to separate toluene form n-heptane using solvent phenol provided by Ma et al. [13]. The process flowsheet is provided in **Figure 7**. The process mainly consists of two columns. The first column (C1) is used to separate n-heptane, and the second column (C2) is used for separation of toluene form solvent phenol. The purity requirement for both products is not less than 0.98. All specific constraints and utility prices are referenced from Fan et al. [10]. The optimization problem using the surrogate models mainly achieves the minimum operating cost by adjusting the amount of extractant, reflux ratio, and distillate flow rate, as shown in Eq.(13).

$$\begin{aligned} \min_{F,r_1,D_1,r_2,D_2\in\mathbb{R}} \quad & c_{hu}(Q_2+Q_4)-c_{cu}(Q_1+Q_3) \\ s.t. \quad & (x_n,x_t,Q_1,Q_2,Q_3,Q_4)=s(F,r_1,D_1,r_2,D_2) \\ & 0.98\le x_n\le 1 \\ & 0.98\le x_t\le 1 \end{aligned} \tag{13}$$

where, c_{hu} and c_{cu} is the cost coefficients for cold and hot utilities; Q_1 and Q_3 are duties of condensers C1 and C2, Q_2 and Q_4 are duties of reboilers C1 and C2, kW; F is the mass flow of solvent phenol, kg·hr^{-1}; r_1 and r_2 are the reflux ratio of C1 and C2; D_1 and D_2 are the distillate flow of C1 and C2; x_n and x_t are the mole fraction of n-heptane in C1 distillate and toluene in C2 distillate; $s(\cdot)$ is the surrogate model for extractive distillation process. The variable range and parameter values are obtained from Fan et al. [10]. In this optimization problem, there are 1, 290 variables and 1289 constraints.

According to Ma et al. [13], the surrogate models constructed using either ReLU neural networks or ALAMO exhibit great relative errors ranging from 0.2% to 25%. Despite employing complex basis functions, these models still fail to fully capture behaviors of the process system due to inherent limitations in their nonlinear representation capabilities. We use the proposed MLFP framework to generate a surrogate model for the entire process where 1014 valid sampling points are generated.

We then use MLFP to solve the problem (13). The convergence curves of the objective value, and n-heptane and toluene molar fractions are depicted in **Figure 8**. As illustrated in **Figure 8**, the predicted objective value, and n-heptane and tolumen molar fractions from the surrogated models are in good agreement with those validated from Aspen rigorous simulation, indidating the high model prediction accuracy. The best energy cost is 134.60 \$ hr^{-1}. The achieved purities of n-heptane and toluene products are 0.98 and 0.98 respectively, which satisfy the purity requirements.

To further illustrate the capability of the proposed MLFP framework, we also use genetic algorithm (GA), particle swarm optimization (PSO) to solve problem (13) where the surrogate model is obtained from the proposed MLFP framework. The reported network structure with the highest accuracy in Ma et al. [13] is also used to develop ReLU NN by using the sampling points generated from LHS, which is then reformulated into an MILP problem and optimized by Gurobi. All optimization results using the surrogate models are validated in Aspen Plus. The

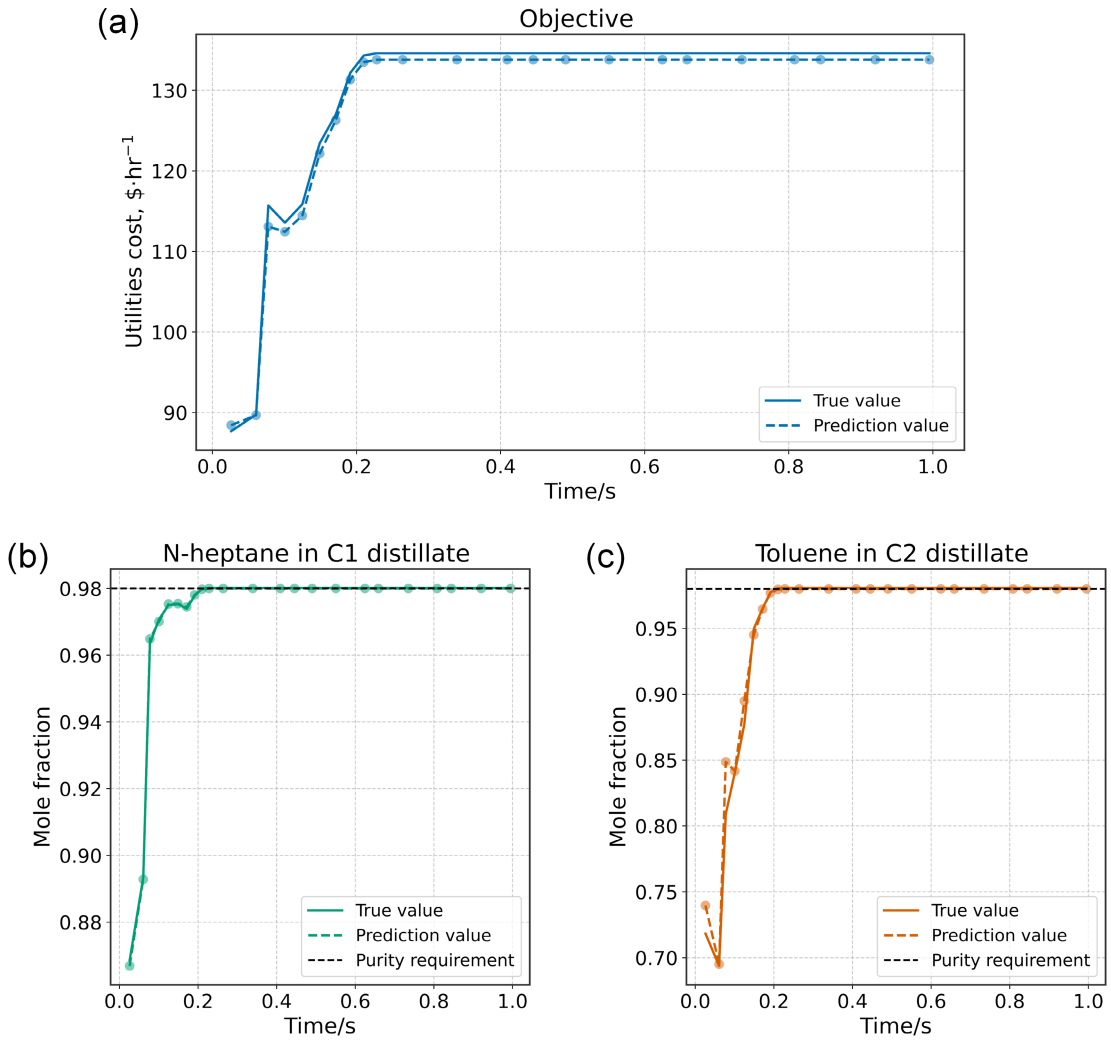

Figure 8. Convergence curves for ED process where (a) objective values, (b) n-heptane molar fraction in the top distillate of C1 and (c) toluene molar fraction in the top distillate of C2.

Table 1: Validated optimization results for ED process from different optimization strategies.

Algorithms	Objective ($·hr−1)	Purity		No. of samples/Initial points	Optimization time (sec)	Total time (min)
		x_n	x_t			
Surrogate-GA	154.77	0.99	0.98	2400	3.59	62.14
Rigorous-GA	177.07	0.98	1.00	-	2898.55	48.31
Surrogate-PSO	134.73	0.98	0.98	2400	3.94	62.15
Rigorous-PSO	151.60	0.98	0.98	-	2919.47	48.66
SQP (Aspen)	231.18	0.99	0.99	1	Failed	-
ASBO-I (best)	248.60	valid	valid	160	-	-
ReLU NN	154.13	0.96	0.96	2400	2.57	79.11
MLFP	134.60	0.98	0.98	2400	0.99	62.08

validated results are provided in **Table 1**. The best results from the adaptive sampling Bayesian optimization (ASBO-I) reported by Fan et al. [10] also provided in **Table 1** for comparison.

From Table 1, it is shown that SQP (Aspen) fails to converge and halts at the initial point provided. ASBO-I requires the fewest samples, but its best objective is 248.60 $\$\cdot hr^{-1}$, which is 84.7% higher than that of 134.60 $\$\cdot hr^{-1}$ obtained from MLFP. Although the reformulated ReLU NN using the network structure reported in Ma et al. [13] obtains the best objective of 154.13 $\$\cdot hr^{-1}$, which is lower than ASBO-I, but still higher than that of 134.60 $\$\cdot hr^{-1}$ from MLFP. Moreover, the purity of n-heptane and toluene is 0.96, which does not meet the purity requirements. In contrast, MLFP exhibits the best performance, generating the lowest objective value of 134.60 $\$\cdot hr^{-1}$ and all purity constraints are met. It should be noted that the total computational time for MLFP reported in **Table 1** includes the optimization time, the effort for evaluating the black-box function and the time for training and pre-training the SVM.

CONCLUSION

In this work, we proposed a novel machine learning powered feasible path optimization (MLFP) framework for solving general BBO problems where complex equality and inequality constraints are involved. The framework includes three main porcedures. First, an efficient adaptive sampling strategy is proposed, which filters out potentially infeasible samples with SVM and focuses on regions likely to contain optimal solutions. Second, machine learning models are leveraged to develop surrogates of black-boxes. Third, the feasible path algorithm is employed, which divides the entire optimization problem into two subproblems including a small optimization problem in the outer level and a simulation problem in the inner level (refering to the use of surrogate models for prediction in this work). Therefore, there are only independent variables that need to be updated rather than all, thus accelerating the theoretical convergence.

Computational studies show that the proposed adaptive sampling approach needs far fewer samples than the commonly used LHS strategy. The feasible path method greatly boosts computational speed, finishing optimization in just a few seconds. For benchmark problems with known global optima, the MLFP converges to solutions around the KKT point. For Williams-Otto optimization problem, MLFP shows similar performance to TRFs, but requires fewer evaluations. For an extractive distillation case, the optimal objective value obtained with MLFP is 84.7% lower than the best result of ASBO-I from its 10 runs.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge financial support from the National Key Research and Development Program of China (2024YFC3909800). Jie Li would like to appreciate the financial support from Engineering and Physical Sciences Research Council (EP/T03145X/1, EP/V051008/1).

REFERENCES

1. Shields BJ, Stevens J, Li J, Parasram M, Damani F, Alvarado JIM, Janey JM, Adams RP, Doyle AG. Bayesian reaction optimization as a tool for chemical synthesis. Nature 590:89-96 (2021). https://doi.org/10.1038/s41586-021-03213-y
2. Rangarajan S, Bhan A, Daoutidis P. Identification and analysis of synthesis routes in complex catalytic reaction networks for biomass upgrading. Applied Catalysis B: Environmental 145:149-160 (2014). https://doi.org/10.1016/j.apcatb.2013.01.030
3. Meijer HEH, Govaert LE. Mechanical performance of polymer systems: the relation between structure and properties. Progress in Polymer Science 30:915-938 (2005). https://doi.org/10.1016/j.progpolymsci.2005.06.009
4. Dedeyne JN, Geerts M, Reyniers PA, Wéry F, Van Geem KM, Marin GB. Computational fluid dynamics?based optimization of dimpled steam cracking reactors for reduced co₂ emissions. AIChE Journal 66: (2020). https://doi.org/10.1002/aic.16255
5. Lang YD, Malacina A, Biegler LT, Munteanu S, Madsen JI, Zitney SE. Reduced order model based on principal component analysis for process simulation and optimization. Energy Fuels 23:1695-1706 (2009). https://doi.org/10.1021/ef800984v
6. Na J, Lim Y, Han C. A modified DIRECT algorithm for hidden constraints in an LNG process optimization. Energy 126:488-500 (2017). https://doi.org/10.1016/j.energy.2017.03.047
7. Bertsimas D, Margaritis G. Global Optimization: A Machine Learning Approach. *arXiv*. Preprint posted online November 3, 2023. Accessed January 12, 2024. http://arxiv.org/abs/2311.01742
8. Cozad A, Sahinidis NV, Miller DC. Learning surrogate models for simulation?based optimization. AIChE Journal 60:2211-2227 (2014). https://doi.org/10.1002/aic.14418
9. Liang R, Han Y, Hu H, Chen B, Yuan Z, Biegler LT. Efficient trust region filter modeling strategies for computationally expensive black-box optimization. Computers & Chemical Engineering 189:108816 (2024). https://doi.org/10.1016/j.compchemeng.2024.108816
10. Fan S, Hong X, Liao Z, Ren C, Yang Y, Wang J, Yang Y. Adaptive sampling bayesian algorithm for constrained black?box optimization problems.

AIChE Journal 71: (2025). https://doi.org/10.1002/aic.18715
11. Ma Y, McLaughlan M, Zhang N, Li J. Novel feasible path optimisation algorithms using steady-state and/or pseudo-transient simulations. Computers & Chemical Engineering 143:107058 (2020). https://doi.org/10.1016/j.compchemeng.2020.107058
12. Ma Y, Gao X, Liu C, Li J. Improved SQP and SLSQP algorithms for feasible path-based process optimisation. Computers & Chemical Engineering 188:108751 (2024). https://doi.org/10.1016/j.compchemeng.2024.108751
13. Ma K, Sahinidis NV, Bindlish R, Bury SJ, Haghpanah R, Rajagopalan S. Data-driven strategies for extractive distillation unit optimization. Computers & Chemical Engineering 167:107970 (2022). https://doi.org/10.1016/j.compchemeng.2022.107970