Optimality

www.opt.reapress.com

Opt. Vol. 3, No. 3 (2026) 283–294.

Paper Type: Original Article

Cost-Normalized Constrained Surrogate Search for a Fed-Batch Model: Reported Sequential and Batch Comparisons

Setayesh Zahiri[1], **Zahra Joorbonyan**[1,*]

[1] Morvarid Intelligent Industrial Systems Research Group, Iran; setayeshzahiri70@gmail.com; Zahrajbn@gmail.com.

Citation:

Received: 25 October 2025 *Revised: 16 December 2025* *Accepted: 29 February 2026*	Zahiri, S., Joorbonyan, Z. (2026). Cost-normalized constrained surrogate search for a fed-batch model: Reported sequential and batch comparisons. *Optimality, 3*(3), 283-294.

Abstract

Sequential surrogate search can help allocate limited evaluations of a constrained bioprocess model. This study reports SURA, a cost-normalized feasible-improvement heuristic with greedy batches and a training-feasibility audit. The numerical record compares five methods on a deterministic four-variable fed-batch surrogate and two constrained analytic problems, using 150 evaluations and ten seeds per configured method–problem pair. On the bioprocess problem, SURA's best feasible endpoint titer is 18.804 ± 0.176 g/L, compared with 19.152 ± 0.193 g/L for sequential GP-EI and 14.678 ± 2.244 g/L for budget-matched random search. The reported signed-rank comparisons favor GP-EI over SURA and SURA over random search (Holm $p = 0.0078$ for each). Disabling batching gives a reported finite-reference difference of −1.172 ± 0.191 g/L versus −0.756 ± 0.176 for the full method, but aggregate ablations do not uniquely identify model staleness as the cause. The audit did not trigger, and its fallback was not validated. The implemented kernel is RBF plus a white-noise term; there are no biological interaction contexts, assay-noise observations or cross-program portfolio decisions in these tests. Budgets are evaluation counts rather than equal cumulative costs. Negative reference differences indicate exceeding a finite random-search sample, not a certified optimum. Original simulator, random-stream, acquisition and plot-generation records remain necessary for reproducibility verification.

Keywords: Constrained optimization, Gaussian processes, Fed-batch simulation, Feasible improvement, Cost normalization, Batch search.

1 | Introduction

Metabolic engineering creates process designs whose assessment can require scarce experiments [1–3]. This motivates careful selection of candidate evaluations. The present work addresses a computational precursor to that problem: Search over deterministic simulator responses. It does not report laboratory resource savings or biological-interaction estimation.

Corresponding Author: Zahrajbn@gmail.com

https://doi.org/10.22105/opt.v3i3.120

Licensee System Analytics. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0).

Efficient global optimization uses a surrogate and expected improvement to select expensive evaluations [4]. Bayesian optimization includes a broad range of acquisition, cost, task and fidelity models [5], [6]. ParEGO extends surrogate search to multiobjective problems [7], while design of experiments supplies complementary information-focused methods [8], [9]. Neither tradition can be characterized as assuming that all evaluations have identical cost or value.

Bioprocess settings may include categorical choices, related tasks, uncertain observations and multiple resource constraints. Multi-task and multi-fidelity surrogates already provide mechanisms for modelling related information [10–12]. A context-dependent extension would require a valid covariance and evidence about transfer between contexts. The reported tests instead use a single program per problem, four continuous variables and deterministic observations; their results cannot establish that extension's benefit.

The contribution is a restricted comparison of cost-normalized feasible-improvement search with configured baselines at a fixed evaluation limit. The study distinguishes the executed RBF-based heuristic from a prospective context-aware or multi-program allocation design, and distinguishes a finite reference value from a global optimum.

Section 2 sets out the methodological antecedents and biological-model context. Section 3 defines the executed optimization quantities. Section 4 describes the reported search and audit rules with their implementation dependencies. Section 5 interprets the preserved numerical summaries; Sections 6 and 7 discuss their limits and further tests.

2 | Related Work

2.1 | Bayesian Optimization and Surrogate-Assisted Search

Gaussian-process regression supports expected-improvement search and predictive modelling [4], [13], [14]. Practical and theoretical studies examine acquisition choices, active data selection and convergence under stated assumptions [5], [6], [15–17]. These results do not automatically guarantee performance for a finite candidate pool, a particular constraint heuristic or an implementation with periodically refitted models. Gardner et al. [18] describe expected-improvement-based optimization with inequality constraints.

Gonzalez et al. [19] develop batch selection through local penalization, while Desautels et al. [20] analyze parallel GP-bandit strategies. These are distinct methods with distinct assumptions. SURA's reported greedy distance penalty is an antecedent-informed heuristic; it does not inherit an approximation or regret guarantee from those methods. The configured ParEGO-lite comparator also differs from a full multiobjective implementation [7]. Related-information models include co-kriging and multi-fidelity optimization [10], [12] and learned multi-task modelling [11]. These sources undermine a categorical claim that task relationships must be fixed in advance. For a future context model, transfer strength must be identifiable and the context covariance positive semidefinite. Neither condition is tested by the context-free campaign.

2.2 | Design of Experiments and Optimal Allocation

Design of experiments and sequential statistical design provide information-oriented alternatives [8], [9]. Bandit theory concerns allocation under specified reward and feedback assumptions [21]. The present comparison uses Latin-hypercube sampling followed by local restarts, rather than a full optimal-design or multi-program bandit baseline. No result about these configured methods establishes dominance over general experimental practice.

2.3 | Bioprocess Modelling and Optimization

Monod and inhibitory growth models and bioprocess balances provide kinetic context [22–26]. Flux-balance and community models offer richer stoichiometric representations [27–30], but the reported fed-batch simulator does not execute them. ANN–metaheuristic optimization of enzyme and biomass responses to

fermentation-medium composition is another direct process-design antecedent [31], it is a static response-model problem, distinct from the deterministic kinetic feed-policy model used here.

2.4 | Positioning of the Comparison

Table 1 summarizes complementary roles without categorical absence claims. The numerical record tests a cost-normalized batch heuristic on three deterministic constrained problems.

Table 1. Complementary methodological roles; entries are examples rather than exclusions.

Method Role	Bayesian Optimization	Design of Experiments	Bioprocess Modelling
Adaptive choices	Sequential and batch acquisition	Sequential designs where used	Can be coupled to optimization
Cost information	Cost-aware variants	Resource constraints in design	Process/resource accounting
Related information	Task and fidelity covariances	Factor and interaction models	Mechanistic interactions
Constraints	Constrained acquisition variants	Feasible design restrictions	Physical/biological bounds
Model role	Predictive surrogate	Information for model estimation	Kinetic or stoichiometric model

It does not test a covariance for biological interactions, allocation across competing programs, censoring, measured assay uncertainty or plant-scale resource scheduling. These prospective extensions require their own data and validation.

3 | Problem Formulation and Research Questions

3.1 | Questions Addressed by the Numerical Record

The executed comparison asks how the configured methods differ in observed feasible objective value, infeasible evaluation fraction and a dimensionless evaluation-cost proxy. A second question concerns differences between SURA and its reported component variants. Context-transfer accuracy, model misspecification across biological contexts and multi-program allocation remain prospective questions because those conditions are absent. Cost normalization may change the selected points, but a 150-call budget does not establish performance at equal cumulative cost. Testing that claim would require a common cost budget and an explicit resource model. The batch comparison requires all initialization, random-stream, candidate-pool, refit and penalty settings to be matched or recorded. A variant difference can be informative without uniquely identifying model staleness as its cause. A context-aware covariance could be investigated in a separate experiment with actual contexts and a valid positive-semidefinite factor. Including a simpler model as a parameter limit does not guarantee that fitting or finite-budget search will approach that model's performance.

3.2 | System Model and Notation

Each reported problem has four continuous variables, a deterministic objective to maximize and constraints observed from the same evaluation. The optimizer receives at most 150 calls, including infeasible calls. K(x) is a known dimensionless positive proxy used in acquisition normalization; it does not represent measured fermenter hours, money or assay capacity. There is one program per problem and no interaction-context label in the execution.

Table 2. Principal notation for the resource-allocation formulation.

Symbol	Meaning	Unit
x	four continuous decision coordinates	coordinate-dependent
i, j	GP observation indices	dimensionless
f(x)	deterministic objective to maximize	g L^{-1} for bioprocess
g_j(x)	observed constraint residual j	constraint-dependent
K(x)	positive known evaluation-cost proxy	dimensionless
B	shared limit of 150 evaluation calls	calls
a(x)	factored expected improvement × feasibility	objective units
σ_n^2	white-noise observation nugget variance	squared response units
C	finite acquisition candidate pool	set of coordinates
A	already selected batch subset	set of coordinates

3.3 | Formal Problem Statement

Eq. (1) represents the reported RBF plus white-noise observation covariance [13], [32]. Its nonnegative variances and positive length scales give a positive-semidefinite covariance. The white-noise term acts on observation indices; for these deterministic simulator responses it is a fitted numerical nugget, not measured biological assay noise.

Eq. (2) writes the maximization acquisition as expected positive improvement times a modeled joint feasibility probability. This factor equals expected feasible improvement only under the relevant conditional independence assumptions; otherwise it is a factored heuristic. The incumbent f_best is the best observed feasible objective value, defined once at least one feasible observation exists. The acquisition rule before that point remains an implementation dependency.

$$K_{ij} = \sigma_f^2 \exp[-\frac{1}{2}\sum_{\ell=1}^{4}\frac{(x_{i\ell} - x_{j\ell})^2}{L_\ell^2}] + \sigma_n^2 \delta_{ij}. \quad (1)$$

$$a(x) = \mathbb{E}[(f(x) - f_{best})_+]\, p_{feas}(x). \quad (2)$$

For a candidate pool C and previously selected batch subset A, *Eq. (3)* describes the incremental score used by a greedy cost-normalized selection heuristic, with $K(x) > 0$ and a nonnegative distance-related penalty. The exact penalty function, scaling, tie-breaking and candidate-pool refresh require the implementation. This is not an independently solved combinatorial optimum or a documented fixed-cost campaign constraint. *Eq. (4)* defines a recommendation among observed feasible points; if none is feasible, that recommendation is undefined and the reporting rule must specify how the run is handled.

$$x_{next} \in \arg\max_{x \in C \setminus A} \{\frac{a(x)}{K(x)} - \lambda \sum_{u \in A} s(x,u)\}, \quad K(x) > 0. \quad (3)$$

$$\hat{x} \in \arg\max_{x \in \mathcal{D}_{obs}:\, g_j(x) \le 0\ \forall j} f(x). \quad (4)$$

A future context extension could multiply a valid process kernel by a context covariance $C = (1 - \eta)I + \eta R$, where $0 \le \eta \le 1$ and R is positive semidefinite with unit diagonal. This convex combination is valid; unrestricted off-diagonal covariance is not. For two contexts, a factor [[1,a],[a,1]] has eigenvalues $1 \pm a$ and fails when $|a| > 1$. This correction defines a prospective model class and is not attributed to the reported code. Multi-program allocation and a cost-to-target objective likewise require a separate implemented protocol.

4 | Proposed Methodology: SURA

SURA denotes the reported cost-normalized, constrained batch-search configuration. Its executed components are the RBF-plus-white-noise GP machinery, a factored improvement–feasibility score, greedy batches of five with a distance penalty, and an audit based on training-point feasibility errors. Biological-context covariance and program portfolios are proposed extensions rather than evaluated modules.

4.1 | Executed Surrogate Model

The source reports a scikit-learn Gaussian-process regressor using RBF plus WhiteKernel covariance, refitted every ten evaluations [13], [32]. Hyperparameter bounds, normalization, initialization and optimizer restarts must be obtained from the original code. There is no heteroscedastic likelihood, correlated assay-error model or censored-observation analysis in the reported deterministic experiment. The no-kernel variant is described as plain RBF, so its comparison concerns a noise-floor choice rather than information transfer across biological contexts.

4.2 | Cost-Aware Constrained Batch Selection

Candidates are scored using the factored acquisition in *Eq. (2)* divided by the dimensionless K(x). A greedy procedure assembles up to five points while penalizing nearby candidates with the reported coefficient 0.15. The method stops at the shared 150-evaluation limit. Exact constraints on remaining calls, initialization and batch completion need source verification; no reported quality-target stopping rule or confirmatory laboratory run is assumed.

A local penalty can discourage near-duplicates but does not by itself establish submodularity, monotonicity or a constant-factor approximation guarantee. Physical scheduling, shared equipment and stochastic experiment failure were not modelled. Thus selection diversity and a reported acquisition score should not be interpreted as certified operational feasibility or cost efficiency.

The final recommendation is the best observed feasible design in *Eq. (4)*. A confirmatory independent run could assess recommendation uncertainty in a noisy application, but no such run is reported here. In a deterministic simulator, repeated evaluation primarily checks implementation consistency rather than providing an unbiased biological-performance estimate.

4.3 | Prospective Multi-Program Extension

Allocating calls across several process programs could build on related-task surrogates and bandit methods [11], [21]. It would require program-specific domains, costs, feedback and a defined utility for information. This campaign has one program per problem, so it provides no observed allocation, stopping or option-value result across programs.

4.4 | Model Auditing and Conservative Fallbacks

The reported audit checks a false-feasible rate at training points. It enters a feasibility-slack mode above 0.25 and releases that mode below 0.15. The denominator, in-sample prediction rule and slack value need the source implementation. This is not a posterior interval-coverage test or an out-of-sample calibration guarantee. The source reports zero triggers in 28 rounds, leaving the conservative fallback untested.

4.5 | Complexity, Assumptions, and Failure Conditions

Dense GP factorization has cubic time and quadratic storage dependence on the observation count [13]. Prediction cost also depends on training size and the covariance calculations for each candidate; it is not only one kernel evaluation per candidate. The reported 800-point pool and refits every ten evaluations constrain this configuration. Total runtime and its allocation across simulator, GP fitting and acquisition require run logs; incremental updates and negligible computation were not established.

The executed tests assume deterministic stationary simulator responses and a known dimensionless cost field. Noise, censoring, drift, failed evaluations and uncertain resource consumption could alter behavior. These conditions were not replayed as stress tests, and no safety or graceful-degradation guarantee follows from an in-sample audit that never triggered.

The three analytic or simulator configurations also do not represent a sampled population of biological processes. Seed variation measures stochastic search behavior on those instances; it does not measure between-process uncertainty.

The reported search initializes observations, fits or periodically refits the GP, scores a finite candidate pool, selects a point or a penalized batch, evaluates objective and constraints, and appends the observations until 150 calls are used. The audit updates its in-sample feasibility diagnostic at the reported round boundary. Exact initialization, model-refresh order, pool reuse and pseudo-random stream handling remain necessary to implement this sequence reproducibly.

5 | Experimental Results and Analysis

5.1 | Setup: Problems, Methods, Protocol, and Scope

The source reports three four-dimensional maximization problems. The primary fed-batch model uses Monod growth with maximum rate 0.35 h^{-1}, saturation constant 0.5 g/L and yield 0.6 g/g; inlet substrate is 20 g/L and initial volume is 1.0 L. Luedeking–Piret coefficients are reported as 0.4 and 0.05, over 25 hours, with an adaptive Runge–Kutta relative tolerance of 10^{-4} [22], [31], [32]. The decisions are two feed rates in [0,0.5] L/h, switch time in [1], [24] h and initial substrate in [0,20] g/L.

Final volume must not exceed 2.2 L and residual substrate must not exceed 1.0 g/L. Complete concentration or amount balances, coefficient units, initial biomass and product, dilution terms and absolute tolerances are needed for reproduction.

A 5,000-draw feasible-fraction estimate is 3.1 percent; it is a sampling estimate, not a proven feasible-box measure. Problems (ii) and (iii) use a negative quadratic and a bilinear objective with ball constraints, respectively. Their reported reference optima are 0.0 and 3.125; exact domains, centers, radii and derivations are missing from the inspected description and are required to certify these optima.

The reported baselines are random search; a 60-point Latin-hypercube design [33] followed by penalized Nelder–Mead restarts [34], GP-EI [4], [35], using RBF plus WhiteKernel, refits every ten evaluations and an 800-point pool [13], [32], and ParEGO-lite [7], fitting random Chebyshev scalarizations of negated objective and constraint violation. SURA uses the same reported GP machinery with cost normalization, greedy batches of five, distance penalty 0.15 and the feasibility audit. Cost proxies range from 1 to 1.9 for problem (i) and from 1 to 1.5 for the analytic problems. All methods receive 150 calls, including infeasible calls.

A separate 5,000-call random search supplies finite reference values 18.049 g/L, −0.836 and 2.481. Ten seeds are reported per method–problem pair, with three ten-seed SURA variants on problem (i). ParEGO-lite is a configured constrained-search scalarization, not a complete comparison of Pareto fronts.

There are no latent interaction contexts, noisy assay measurements, cross-program choices or equal cumulative-cost budgets in this campaign. The source reports NumPy 2.1.3, SciPy 1.14.1, scikit-learn 1.5.2, statsmodels 0.14.5 and Matplotlib 3.9.2, and a total of 414 seconds on two cores for 180 optimizer runs plus 15,000 reference calls. Original logs are required to verify this provenance. Shared seed labels do not by themselves establish common candidate initializations or paired random streams; the exact coupling is needed to assess the paired tests.

5.2 | Bioprocess Comparison

Table 3 reports the bioprocess summaries. Random search has a 97.1 percent infeasible fraction and best feasible titer 14.678 g/L. The four structured methods exceed the finite 5,000-call reference on this problem: GP-EI gives 19.152 g/L, the DoE/restart pipeline 18.914, SURA 18.804 and ParEGO-lite 18.593. SURA is therefore numerically below both GP-EI and the DoE pipeline.

Its reported Holm-corrected contrast favors GP-EI ($p=0.0078$), while the DoE contrast does not reject ($p=0.8457$). SURA has the smallest displayed seed standard deviation for this metric, 0.176, but that dispersion is not predictive calibration. These outcomes concern one kinetic configuration and do not show superiority over classical practice as a whole. Original numerical cells are retained.

Table 3. Reported fed-batch results: Observed best feasible titer, signed finite-reference difference, infeasible fraction and dimensionless mean evaluation-cost proxy (mean ± standard deviation over ten seeds).

Method	Best Feasible Titer (g/L)	Regret vs Reference (g/L)	Infeasible Fraction	Mean Cost (Units)
Random search	14.678 ± 2.244	3.370 ± 2.244	0.971	1.451
LHS DoE + N-M restarts	18.914 ± 0.803	-0.865 ± 0.803	0.547	1.322
Sequential GP-EI	19.152 ± 0.193	-1.104 ± 0.193	0.729	1.336
ParEGO-lite	18.593 ± 0.483	-0.544 ± 0.483	0.803	1.391
SURA (full)	18.804 ± 0.176	-0.756 ± 0.176	0.725	1.362

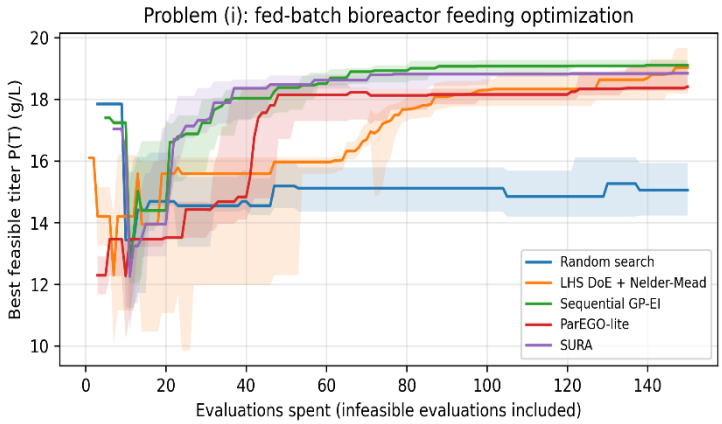


Fig. 1. Original reported convergence plot.

Fixed-run best-so-far values and their median should be nondecreasing in deterministic maximization, yet several displayed traces decline. The original feasible-run inclusion, missing-value handling and plot-generation records are required to reconcile the curves with the stated statistic; no monotone best-so-far certification is inferred from this figure.

5.3 | Component Ablations

Table 4 and *Fig. 2* report the bioprocess variants. Batch size one gives a finite-reference difference of −1.172 ± 0.191 g/L compared with −0.756 ± 0.176 for the full configuration (reported $W = 0$ and $p = 0.0020$). Its GP-EI contrast does not reject ($p = 0.5566$), which does not establish equivalence. The no-kernel and no-audit contrasts also do not reject. These variant summaries support a batch-size comparison but do not isolate staleness: The baseline GP-EI is refitted only every ten calls, and model fitting, penalty, random-stream and candidate-history differences require verification.

The audit never triggered, yet the no-audit summary differs from the full method; identical behavior cannot be asserted without the original implementation and coupled streams. The 28-round count also needs reconciliation with initialization and a five-call batch limit. Absence of rejection does not establish equality [36].

Table 4. Reported SURA variants on the bioprocess problem: Signed finite-reference differences (mean ± standard deviation over ten seeds) and paired signed-rank summaries.

Variant	Regret vs Reference (g/L)	Infeasible Fraction	Wilcoxon vs Full: W	p
SURA (full)	-0.756 ± 0.176	0.725	-	-
w/o interaction kernel	-0.812 ± 0.374	0.739	26.0	0.9219
w/o batch (batch = 1)	-1.172 ± 0.191	0.809	0.0	0.0020
w/o audit	-0.830 ± 0.223	0.743	21.0	0.5566

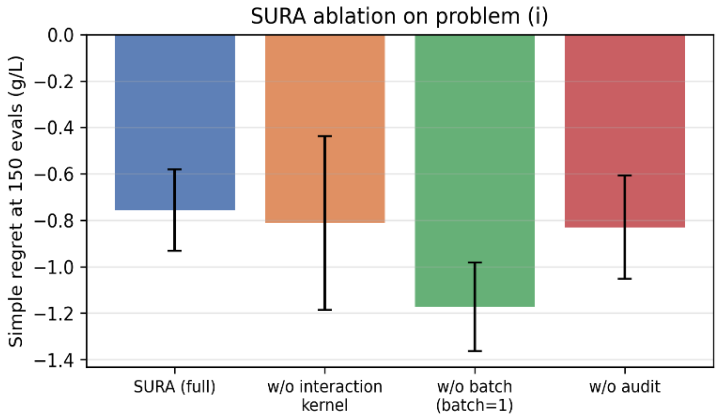


Fig. 2. Original ablation summaries.

The ordinate labeled simple regret is a signed difference from a finite random-search reference, not a nonnegative optimality gap. Error bars are reported as seed standard deviations; original arrays are required to verify them.

5.4 | Analytic-Problem Comparisons

Table 5 reports the two analytic configurations. Using the source's stated optimum references, GP-EI has the smallest displayed gaps, 0.165 and 0.276. SURA has 0.514 and 0.335, differences of 0.349 and 0.059 from GP-EI; the primary titer difference is 0.348 g/L. These quantities have different objective units and cannot be pooled as one common gap. GP-EI's problem-(ii) contrast is significant after the reported correction (Holm $p = 0.0195$), while the problem-(iii) contrast does not reject. SURA improves over DoE on both analytic problems and over ParEGO-lite on (iii) according to the reported tests. Positive finite-reference differences for DoE on both analytic problems and ParEGO-lite on (iii) show that not all structured methods exceed the random-search reference in every problem. Exact analytic constraints remain necessary to verify the claimed optimum gaps. Exact domains and constraints are needed to certify the optimum references.

Table 5. Reported analytic-problem summaries: Signed difference from a finite random-search reference, gap to the stated optimum reference and infeasible fraction.

Method	(ii) Regret vs Ref.	(ii) Gap to Optimum	(ii) Infeas.	(iii) Regret vs Ref.	(iii) Gap to Optimum	(iii) Infeas.
Random search	+6.273	7.109	0.890	+0.880	1.524	0.871
LHS DoE + N-M restarts	+2.708	3.544	0.405	+0.302	0.945	0.529
Sequential GP-EI	-0.671	0.165	0.755	-0.367	0.276	0.929
ParEGO-lite	-0.469	0.367	0.625	+0.231	0.874	0.816
SURA (full)	-0.322	0.514	0.730	-0.308	0.335	0.947

5.5 | Statistical Analysis

Table 6 preserves the reported Friedman and Holm-corrected signed-rank summaries [37–39]. The Friedman chi-square tails for the rounded statistics 24.72, 33.84 and 25.92 with four degrees of freedom are approximately 5.73×10^{-5}, 8.04×10^{-7} and 3.28×10^{-5}, consistent with the displayed rounding. For ten nonzero pairs, the smallest exact two-sided signed-rank p-value is $2/2^{10} \approx 0.001953$.

The source displays this as 0.0020 for several contrasts, including GP-EI on (i); the GP-EI contrast on (ii) instead has $W = 3$ and $p = 0.0098$. The reported significant comparisons favor SURA over random search on all three problems, over DoE on (ii) and (iii), and over ParEGO-lite on (iii), while favoring GP-EI on (i) and (ii). Other contrasts do not establish equivalence. Original paired arrays, tie handling and correction

families are required to certify the analysis; search seeds on three fixed configurations do not validate a population of biological processes.

Table 6. Inferential statistics: Friedman tests across the five methods and paired Wilcoxon (with Holm correction) of SURA against each baseline on regret at 150 evaluations, per problem.

Problem	Friedman Chi2 (df = 4)	Friedman p	SURA Contrast	W	p	Holm p
(i)	24.72	5.7e-05	vs random	0.0	0.0020	0.0078
			vs LHS+N-M	25.0	0.8457	0.8457
			vs GP-EI	0.0	0.0020	0.0078
			vs ParEGO-lite	13.0	0.1602	0.3203
(ii)	33.84	8e-07	vs random	0.0	0.0020	0.0078
			vs LHS+N-M	0.0	0.0020	0.0078
			vs GP-EI	3.0	0.0098	0.0195
			vs ParEGO-lite	18.0	0.3750	0.3750
(iii)	25.92	3.3e-05	vs random	0.0	0.0020	0.0078
			vs LHS+N-M	2.0	0.0059	0.0117
			vs GP-EI	21.0	0.5566	0.5566
			vs ParEGO-lite	1.0	0.0039	0.0117

5.6 | Findings, Honest Losses, and Scope

On the primary problem, all four structured methods exceed the finite 5,000-call random-search reference at 150 calls. This is a sample-reference comparison, not an optimality result. Constraint violations remain common: GP-EI and SURA spend reported fractions 0.729 and 0.725 of calls on infeasible configurations, while DoE has the smallest displayed fraction, 0.547. The batch-size-one variant has a better reported mean than the batch-five configuration, without establishing that sequential selection against an always-fresh model is the sole cause.

The cost field affects point selection but budgets are equal call counts. If each reported run uses all 150 calls, multiplying the displayed mean cost by 150 gives approximate mean proxy expenditure; this would still not establish a common-cost comparison or measured resource savings. Context-transfer accuracy and misspecification were not tested, the audit fallback never activated, and no portfolio allocation occurred. A fitted nugget is not an interaction-uncertainty estimate.

The numerical record comprises original manuscript tables and embedded plots. The source program, raw optimizer traces and result arrays were not available with the inspected manuscript, so independent regeneration and released-package verification are not claimed. The plotted nonmonotonic best-so-far traces and inactive-audit variant difference require reconciliation from those materials.

6 | Discussion: Limitations and Counterarguments

6.1 | Limitations

The tests use a simple kinetic simulator and two analytic configurations with deterministic observations. Missing full balances, initial states, exact analytic domains, constraint models, initialization and stream coupling limit reproducibility. There is no measured assay-noise, censoring, drift, contamination, scheduling, resource-failure or simulator–plant mismatch study. A prospective context kernel and multi-program design therefore remain model proposals rather than validated capabilities.

6.2 | Alternative Methods

Established constrained, multi-task and batch Bayesian-optimization methods [7], [10–12], [18–20] are relevant alternatives. Their capabilities should not be excluded by a literature-gap claim. A useful next comparison would specify the biological information and resource accounting and compare suitably implemented methods at matched cumulative cost. An in-sample diagnostic that never triggered is insufficient to certify conservative behavior under model failure.

7 | Conclusion

The reported campaign compares a cost-normalized constrained batch heuristic with four configured alternatives. On the fed-batch surrogate, SURA gives 18.804 ± 0.176 g/L, below GP-EI's 19.152 ± 0.193 g/L and the DoE pipeline's displayed mean of 18.914 g/L, while improving on budget-matched random search. The batch-size-one variant has a better reported mean finite-reference difference than full SURA. These observations support a limited algorithm-configuration comparison and do not validate biological interaction modelling, program portfolios, equal-cost resource savings or a unique explanation for the batch difference.

Further work should first supply the complete simulator, analytic domains, paired arrays, random-stream conventions and plot-generation code. A cost-budget experiment should record cumulative expenditure, while a context-transfer experiment requires actual related contexts and a valid covariance. Independent audit evaluation should include held-out errors and triggered fallback behavior. These tests would make the proposed extensions assessable without assuming that the present context-free results already establish them.

Data Availability

Full simulator parameters, initial conditions, analytic definitions, seed-level outputs, acquisition traces and figure data are required to reproduce the reported results. Their release and access location require author confirmation.

Code Availability

The source identifies exp_06.py, exp_06_results.json, exp_06_f1.png and exp_06_f2.png. These materials were not available with the inspected manuscript; a complete versioned package and access location should be supplied before a release or reproducibility claim is made.

Funding

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Conflicts of Interest

The authors declare no conflicts of interest.

Author Contributions

Zare Baghiabad, F.: Conceptualization, methodology, software, and writing, original draft. Zahiri, S.: data curation, formal analysis, and visualization. Joorbonyan, Z.: Validation, supervision, and writing, review and editing.

Generative-AI Disclosure

The source reports generative-AI assistance with text and code. The authors should confirm the final disclosure and remain responsible for accuracy, provenance and approval of the revised content.

References

[1] Ro, D. K., Paradise, E. M., Ouellet, M., Fisher, K. J., Newman, K. L., Ndungu, J. M., ... & Keasling, J. D. (2006). Production of the antimalarial drug precursor artemisinic acid in engineered yeast. *Nature*, *440*(7086), 940-943. https://doi.org/10.1038/nature04640.

[2] Peralta-Yahya, P. P., Zhang, F., Del Cardayre, S. B., & Keasling, J. D. (2012). Microbial engineering for the production of advanced biofuels. *Nature*, *488*(7411), 320-328. https://doi.org/10.1038/nature11478.

[3] Nielsen, J., & Keasling, J. D. (2016). Engineering cellular metabolism. *Cell*, *164*(6), 1185-1197. https://doi.org/10.1016/j.cell.2016.02.004.

[4] Jones, D. R., Schonlau, M., & Welch, W. J. (1998). Efficient global optimization of expensive black-box functions. *Journal of global optimization*, *13*(4), 455-492. https://doi.org/10.1023/a:1008306431147.

[5] Snoek, J., Larochelle, H., & Adams, R. (2012). Practical bayesian optimization of machine learning algorithms. *Advances in neural information processing systems*, *25*, 2951–2959. https://doi.org/10.5555/2999325.2999464

[6] Shahriari, B., Swersky, K., Wang, Z., Adams, R. P., & De Freitas, N. (2015). Taking the human out of the loop: A review of Bayesian optimization. *Proceedings of the IEEE*, *104*(1), 148-175. https://doi.org/10.1109/jproc.2015.2494218.

[7] Knowles, J. (2006). ParEGO: A hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems. *IEEE transactions on evolutionary computation*, *10*(1), 50-66. https://doi.org/10.1109/tevc.2005.851274.

[8] Box, G. E., Hunter, W. H., & Hunter, S. (1978). *Statistics for experimenters* (Vol. 664). New York: John Wiley and sons. https://hwbdocs.env.nm.gov/Los%20Alamos%20National%20Labs/TA%2054/11528.pdf

[9] Montgomery, D. C. (2005). *Design and analysis of experiments*. Hoboken. https://books.google.com/books/about/Design_and_Analysis_of_Experiments_8th_E.html?id=XQAcAAAAQBAJ

[10] Kennedy, M. C., & O'Hagan, A. (2000). Predicting the output from a complex computer code when fast approximations are available. *Biometrika*, *87*(1), 1-13. https://doi.org/10.1093/biomet/87.1.1

[11] Swersky, K., Snoek, J., & Adams, R. (2013). Multi-task bayesian optimization. *Advances in neural information processing systems*, *26*, 2004–2012. https://doi.org/10.5555/2999792.2999836

[12] Forrester, A. I., Sóbester, A., & Keane, A. J. (2007). Multi-fidelity optimization via surrogate modelling. *Proceedings: Mathematical, physical and engineering sciences*, 463(2088), 3251-3269. https://doi.org/10.1098/rspa.2007.1900.

[13] Rasmussen, C. E., & Williams, C. K. (2006). Gaussian processes for machine learning, ser. Adaptive computation and machine learning. *Cambridge, MA, UsA: MIT press*, *38*, 715-719. https://doi.org/10.7551/mitpress/3206.001.0001

[14] Forrester, A., Sobester, A., & Keane, A. (2008). *Engineering design via surrogate modelling: A practical guide*. John Wiley & Sons. https://doi.org/10.1002/9780470770801.

[15] MacKay, D. J. (1992). Information-based objective functions for active data selection. *Neural computation*, *4*(4), 590-604. https://doi.org/10.1162/neco.1992.4.4.590.

[16] Settles, B. (2012). *Active learning*. Springer Cham. https://doi.org/10.2200/S00429ED1V01Y201207AIM018.

[17] Bull, A. D. (2011). Convergence rates of efficient global optimization algorithms. *Journal of machine learning research*, *12*(10), 2879–2904. https://www.jmlr.org/papers/volume12/bull11a/bull11a.pdf

[18] Gardner, J. R., Kusner, M. J., Xu, Z. E., Weinberger, K. Q., & Cunningham, J. P. (2014). Bayesian optimization with inequality constraints. *Proceedings of the 31st international conference on international conference on machine learning* (Vol. 2014, pp. 937-945). JMLR.org. https://doi.org/10.5555/3044805.3044997

[19] González, J., Dai, Z., Hennig, P., & Lawrence, N. (2016). *Batch Bayesian optimization via local penalization*. https://doi.org/10.48550/arXiv.1505.08052

[20] Desautels, T., Krause, A., & Burdick, J. W. (2014). Parallelizing exploration-exploitation tradeoffs in gaussian process bandit optimization. *The journal of machine learning research*, *15*(1), 3873-3923. https://www.jmlr.org/papers/volume15/desautels14a/desautels14a.pdf

[21] Bubeck, S., & Cesa-Bianchi, N. (2012). Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and trends® in machine learning*, *5*(1), 1-122. https://doi.org/10.1561/2200000024.

[22] Monod, J. (1949). The growth of bacterial cultures. *Annual review of microbiology*, *3*, 371–394. https://doi.org/10.1146/annurev.mi.03.100149.002103.

[23] Andrews, J., F. (1968). A mathematical model for the continuous culture of microorganisms utilizing inhibitory substrates. *Biotechnology and bioengineering 10*(6), 707–723. https://doi.org/10.1002/bit.260100602.

[24] Bailey, J. E., & Ollis, D. F. (2018). *Biochemical engineering fundamentals*. McGraw-Hill. https://modps71.lib.kmutt.ac.th/xmlui/handle/123456789/324

[25] Doran, P. M. (1995). *Bioprocess engineering principles*. Academic Press, London. https://books.google.com/books/about/Bioprocess_Engineering_Principles.html?id=PZaEAQAACAAJ

[26] Bastin, G., & Dochain, D. (1990). *On-line estimation and adaptive control of bioreactors* (Vol. 1, p. 379). Amsterdam: Elsevier. https://d1wqtxts1xzle7.cloudfront.net/54061539/G._Bastin_Auth._On-line_Estimation_and_Adaptive_Control_of_Bioreactors-libre.pdf?1501895262

[27] Orth, J. D., Thiele, I., & Palsson, B. Ø. (2010). What is flux balance analysis?. *Nature biotechnology*, *28*(3), 245-248. https://doi.org/10.1038/nbt.1614.

[28] Zomorrodi, A. R., & Maranas, C. D. (2012). OptCom: A multi-level optimization framework for the metabolic modeling and analysis of microbial communities. *PLoS computational biology*, *8*(2), e1002363. https://doi.org/10.1371/journal.pcbi.1002363.

[29] Zomorrodi, A. R., & Segrè, D. (2016). Synthetic ecology of microbes: mathematical models and applications. *Journal of molecular biology*, *428*(5), 837-8 61. https://doi.org/10.1016/j.jmb.2015.10.019.

[30] Mahadevan, R., Edwards, J. S., & Doyle, F. J. (2002). Dynamic flux balance analysis of diauxic growth in Escherichia coli. *Biophysical journal*, *83*(3), 1331-1340. https://doi.org/10.1016/S0006-3495(02)73903-9.

[31] Lamtar Gholipoor, M., Alimoradi, M., & Fakheri, S. (2024). A novel metaheuristic approach inspired by trees social relationships and models for fermentation medium. *Metaheuristic algorithms with applications, 1 (1), 1-11*. https://doi.org/10.22105/maa.v1i1.17.

[32] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *the journal of machine learning research*, *12*, 2825-2830. https://doi.org/10.5555/1953048.2078195

[33] McKay, M. D., Beckman, R. J., & Conover, W. J. (2000). A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, *42*(1), 55-61. https://doi.org/10.1080/00401706.1979.10489755.

[34] Nelder, J. A., & Mead, R. (1965). A simplex method for function minimization. *The computer journal*, *7*(4), 308-313. https://doi.org/10.1093/comjnl/7.4.308.

[35] Močkus, J. (1974). On Bayesian methods for seeking the extremum. *IFIP technical conference on optimization techniques* (pp. 400-404). Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/3-540-07165-2_55.

[36] M. Friedman. The use of ranks to avoid the assumption of normality implicit in the analysis of variance. Journal of the American Statistical Association 32(200), 675–701 (1937). https://doi.org/10.1080/01621459.1937.10503522.

[37] Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrics Bulletin 1*(6), 80–83. https://doi.org/10.2307/3001968.

[38] S. Holm. (1979). A simple sequentially rejective multiple test procedure. *Scandinavian journal of statistics 6*(2), 65–70. https://scispace.com/pdf/a-simple-sequentially-rejective-multiple-test-procedure-458rh2uo9s.pdf

[39] Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of machine learning research*, *7*, 1-30. https://doi.org/10.5555/1248547.1248548