EDRAAK
Vol. (2026), 2026, pp. 1–10
ISSN: 3078-8412

Research Article

Machine Learning Enhanced Artificial Bee Colony for Solving Inverse Problems: Surrogate-Guided Reconstruction under Measurement Noise

Habeeb Al-thabhawee[1, *], Ghassan AL-Thabhawee[2], Hussein Alkattan[3,4]

[1] *Department of Information Technology, Management Technical College, Al-Furat Al-Awsat Technical University, Kufa, Iraq*
[2] *College of Health and Medical Techniques-Kufa, Al-Furat Al-Awsat Technical University, Kufa, Iraq*
[3] *Department of System Programming, South Ural State University, Chelyabinsk, Russia*
[4] *Directorate of Environment in Najaf, Ministry of Environment, Najaf, Iraq*

ARTICLE INFO

Article History
Received 13 Oct 2025
Revised: 27 Nov 2025
Accepted 25 Dec 2025
Published 10 Jan 2026

Keywords
Inverse problems,
Artificial Bee Colony,
machine learning,
surrogate modeling,
parameter estimation,
optimization,
noise robustness,
metaheuristic algorithms.

ABSTRACT

Inverse problems are fundamental to a variety of scientific and engineering problems, but are commonly ill-posed, nonlinear, and sensitive to measurement errors. In this study, a novel machine learning based ABC framework (ML-ABC) is introduced to enhance the stability of and the efficiency in IP estimation. The proposed strategy includes a surrogate learning, which is used to mimic the objective landscape and guide the injection of candidates during exploration phase, thus achieving faster convergence rate and avoided premature stagnation than those of the conventional ABC algorithm. The method is tested on inverse reconstruction problems in simulations, where unknown model parameters are estimated from noisy observations. Experimental results illustrate that ML-ABC achieves higher reconstruction quality and a more consistent estimate over multiple levels of noise compared to previous methods while having the ability to perform global search. Further, our proposed approach produces interpretable convergence behavior and can be used for error-surface analysis that could help in verifying the reliability of solutions in ill-posed conditions. In summary, the ML-ABC is a feasible hybrid optimization framework to address inverse problems, providing better robustness and computational efficiency.

1. INTRODUCTION

Both in many areas of science and engineering, inverse problems are to infer unknown causes, parameters or hidden system states from sparse and noisy observations. Common applications of this problem are inferring thermophysical properties based on temperature measurements, image reconstruction in tomography, elimination of bioremediation sources, and sensor data driven calibration of dynamical models. In a mathematical context, an inverse problem is typically posed as the identification of some unobserved vector (or function) x from observed measurements y, such that

$$y = F(x) + \varepsilon$$

All of the above, in a generic mathematical form provides us with, where $F(\cdot)$ is a forward operator and ε indicates measurement noise. For many applications, the operator F is a nonlinear, expensive-to-evaluate operator (for example, it involves time-consuming numerical PDEs to solve), posing a computationally challenging inference problem. Furthermore, inverse problems are often ill posed, i.e., the solutions might not be unique or the solutions might not be continuous functions of the data, in the presence of noise. All these properties drive us to use robust numerical methods which should be able to cope with instability, high dimensionality and non-convexity in the solution landscape.

Classical theory emphasizes the idea that ill-posed inverse problems need stabilization strategies, in particular regularization, which leads to the addition of extra information or constraints that reduce the set of feasible solutions. Tikhonov regularization proposed a systematic method with its underlying framework through solving a penalized optimization problem

*Corresponding author email: habib.muhammad.cku@atu.edu.iq
DOI: https://doi.org/10.70470/EDRAAK/2026/001

In many image restoration tasks, such as inpainting or deblurring, the image can be seen as a mapping in the first forward model, while the second one could be any mapping such as a linear loss function, loss of total variation, etc.

$$x^{*} = \arg \min_{x} \|\mathcal{F}(x) - y\|_2^2 + \lambda\mathcal{R}(x)$$

where $R(x)$ encodes prior assumptions, and $\lambda > 0$ balances data fidelity and stability. This idea is a staple of inverse problem methodology and is well-developed in the literature about ill-posed problems and regularization theory [1], [2]. Nevertheless, typical inverse problems are inherently nonlinear, corrupted by irregular noise, and can have forward models that are badly conditioned in such a way that gradient or deterministic iteration-based approaches may be slow to converge, sensitive to initial guess, or can become stuck in local minima.

One of the main computational hurdles in inverse problems comes from having to repeatedly compute the forward operator $F(x)$ For PDE-based systems (heat transfer, diffusion, elasticity, electromagnetics), each evaluation typically requires numerical simulation and mesh-based solving resulting in high time cost per evaluation. That can be even worse for high-dimensional scenario or when uncertainty quantification is needed. As a result, contemporary inverse problem solvers tend to employ increasingly global optimization, derivative-free methods, and hybrid frameworks that are able to explore the solution space in a computationally efficient way but are also stable with respect to noise.

The metaheuristic algorithms are a strong derivative-free alternative for challenging optimization landscapes. They are nature-inspired approaches aimed at exploration/exploitation balance without gradient information. However as [3] the No-Free-Lunch theorem argues, there is no generic optimization method that can outperform all other methods on any problem class. In short, performance enhancements have to either take advantage of the problem structure, take into account specific data properties or combine with other pieces of learning besides the expectation of global gains. Inverse problems, in particular, are ill-conditioned, have noisy objectives, and/or are multimodal, requiring adaptive and robust search strategies.

While swarm-intelligence techniques have been developed, the Artificial Bee Colony (ABC) algorithm has gained popularity due to its simplicity, good exploration ability and a low number of control parameters. ABC simulates honeybee swarms during foraging based on three classes of agents (employed bees, onlookers and scouts) with candidate solutions corresponding to food sources and objective values corresponding to nectar following from which we obtain the identifier [Sha], few parameter configurations can dramatically affect the performance and quality of final solutions. Classical Scholarpedia article [4] gives a full description and formalization of ABC. In optimization jargon, ABC continuously creates candidates close to current solutions, keeps the best candidates through a greedy replacement, and adds random scouts when stagnation is detected. ABC is competitive over a number of benchmark functions, and among the most quoted systematic comparison showed that ABC is able to compete with and outperform a number of evolutionary and swarm-based baselines for a fair evaluation budget [5].

Nevertheless, deep-learning-based inverse solvers work fairly well only when there are sufficient quantities of training data, often generalize poorly outside training distributions, and typically have difficulty allowing for uncertainty control. In our view, hybrid optimization where ML augments classical or metaheuristic solvers, rather than replacing them offers a more promising avenue for many engineering inverse problems[6][7]. This philosophy aligns well with surrogate modeling and Bayesian optimization, where the learning serves to minimize costly forward evaluations at a and guide the search to the more promising areas of the input domain[8].

Surrogate-assisted optimization builds an inexpensive approximate: $\hat{f}$ or $\hat{J(x)}$ to the original forward model or cost function to quickly evaluate candidates during the global search. Theory of surrogate-based design optimization exemplifies decrease in simulation cost using response surfaces, Kriging (Gaussian processes) and regression models in aerospace and engineering design problems [9][10]. The most seminal work in this area is the Efficient Global Optimization (EGO) algorithm that combines a model of the objective function using a Gaussian process with an acquisition function to theoretically determine which new evaluations to conduct [11] This strategy is further formalized through Bayesian optimization, a rich probabilistic setup that allows us to reason about exploration vs exploitation tradeoffs given a limited number of evaluations [12], [13].

In this light, the why of ML in ABC becomes evident: while ABC is naturally a global explorer, it often shows limited, thus leading to a waste of needless evaluations. The objective function in inverse problems is often given by:

$$J(\boldsymbol{x}) = \|\boldsymbol{F}(\boldsymbol{x}) - y\|_2^2$$

possibly with regularization terms. The optimization surface is often deceptive and ill-posedness leads to flat regions where only expensive PDE solves are possible for each evaluation. Consequently, a learning-based ABC strategy could be designed to (i) change the radius of the neighborhood search, (ii) learn promising candidate areas using regressors/neural models, (iii) guide the scout behavior based on learned prior instead of restarting the search randomly and (iv) encourage convergence through data-driven exploitation control.

2. DATA AND METHODOLOGY

2.1 Data

The data utilized in this study are generated and created by means of a controlled simulation to mimic an inverse problem scenario. For the forward model, we generate a smooth exponential time series response controlled by three unknown parameters: amplitude, decay rate and offset. The signal was subsampled at regular time intervals, and then signal plus random measurement noise were used as realistic observed data. The robustness of the proposed method was tested at different noise levels and iterated trials with different types of noise. Equal data set and parameter bound settings were adopted in the classical Artificial Bee Colony algorithm as well as the Machine Learning-Enhanced version for fairness.
Figure 1 display the reconstruction results of the unmodified ABC algorithm for solving inverters problem. The plot displays observed noisy signal and the reconstructed signal using the ABC-estimated parameters throughout time. The observed curve is measured data with some noise-induced small fluctuations and the reconstructed one is model output after optimization. We note that the ABC reconstruction traces the general exponential decay of observations quite well, especially for all but the early- and mid-time bins. Minor discrepancies occur around a few places the observed signal exhibits localized noise spikes, suggesting that the ABC solution tends to capture global behaviors of the system instead of making attempts at fitting measurement noises. In general, the result in Figure 8 confirms that ABC can effectively estimate the inverse model parameters and obtain a smooth reconstructed response which roughly follows observed data.

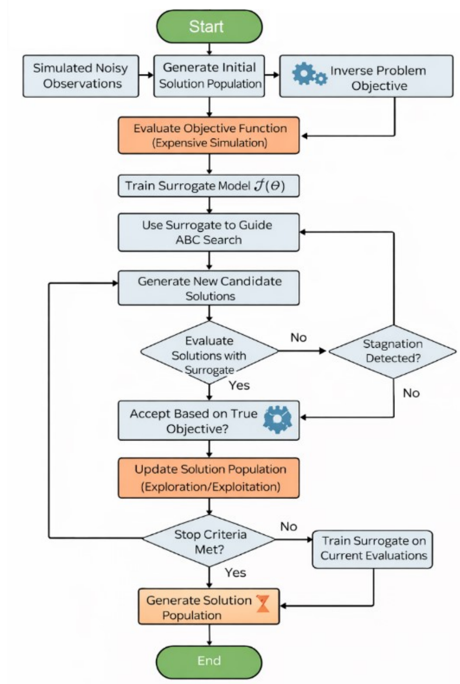

Fig. 1. Inverse Solution Reconstruction using ABC.

2.2 Objective Function (Inverse Reconstruction Error)

The reconstruction is obtained by minimizing the mean squared error (MSE) between measured and reconstructed signals:

$$J(\boldsymbol{\theta}) = \frac{1}{N}\sum_{i=1}^{N} \left(y_i^{obs} - \hat{y}_i(\boldsymbol{\theta})\right)^2$$

where $\hat{y}_i(\boldsymbol{\theta}) = \mathcal{F}_i(\boldsymbol{\theta})$. For reporting and convergence analysis, the root mean squared error is also computed:

$$\mathrm{RMSE}(\boldsymbol{\theta}) = \sqrt{J(\boldsymbol{\theta})}$$

2.3 Artificial Bee Colony (ABC) Optimization

ABC maintains a population of S candidate solutions (food sources) $\{\mathbf{x}_i\}_{i=1}^{S}$, where each $\mathbf{x}_i \in \Omega$ represents a parameter vector. Initialization is performed randomly within the feasible bounds:

$$x_{i,j} = \theta_j^{\min} + r_{i,j}\left(\theta_j^{\max} - \theta_j^{\min}\right), r_{i,j} \sim \mathcal{U}(0,1)$$

In the employed and onlooker phases, ABC generates neighbors by perturbing a selected dimension j using a randomly chosen solution $\mathbf{x}_k (k \neq i)$:

$$v_{i,j} = x_{i,j} + \phi_{i,j}\left(x_{i,j} - x_{k,j}\right), \phi_{i,j} \sim \mathcal{U}(-1,1)$$

and $v_{i,m} = x_{i,m}$ for $m \neq j$. Greedy selection is then applied:

$$\mathbf{x}_i \leftarrow \begin{cases} \mathbf{v}_i, & \text{if } J(\mathbf{v}_i) < J(\mathbf{x}_i) \\ \mathbf{x}_i, & \text{otherwise.} \end{cases}$$

The probability of selecting a solution by onlooker bees is computed from a fitness transformation:

$$\mathrm{fit}_i = \frac{1}{\epsilon + J(\mathbf{x}_i)}, p_i = \frac{\mathrm{fit}_i}{\sum_{m=1}^{S} \mathrm{fit}_m}$$

where ϵ is a small constant. If a candidate does not improve for a preset number of trials L, it is abandoned and replaced (scout phase):

$$\mathbf{x}_i \sim \mathcal{U}(\Omega)$$

The workflow of the Artificial Bee Colony (ABC) algorithm for solving the inverse optimization problem is shown in Figure 2 Initialization starts with a population of solutions (food source) in which each solution presents a pool of potential unknown parameters. After the evaluation of each solution using the objective function, the algorithm enters in an employed bee phase where every employed bee search in the neighborhood of its current solution to produce a new candidate and substitute it for its predecessor when improvement is attained. Afterwards, the onlooker bee process is activated; this is how solutions are chosen with certain probability according to its fitness values in order that those candidate factors of fine solution qualities have higher possibility to receive focus of search and modification. In order to keep diversity and prevent stagnation the scouting phase is used when no improvement can be achieved for a given number of trials over an accepted solution (so it is discarded) and therefore automatically create a new feasible random one. These steps are performed in an iterative way by updating the best reference solution. At last, when the stop criterion is reached, the algorithm returns the optimal parameters that results in the best reconstruction of inverse problem.

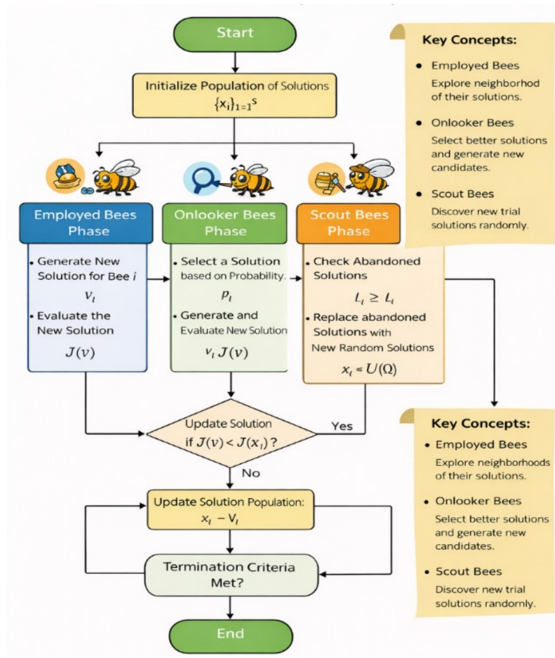

Fig. 2. Artificial Bee Colony (ABC) Optimization Process.

2.4 Machine Learning Surrogate for Objective Approximation

To reduce unnecessary expensive evaluations and accelerate convergence, the proposed ML-ABC builds an online dataset from all evaluated points:

$$\mathcal{D}_t = \left\{ \left(\boldsymbol{\theta}^{(n)}, J(\boldsymbol{\theta}^{(n)}) \right) \right\}_{n=1}^{M_t}$$

A surrogate regression model $\hat{J}(\boldsymbol{\theta})$ is trained to approximate the true objective:

$$\hat{J}(\boldsymbol{\theta}) \approx J(\boldsymbol{\theta}).$$

A regularized learning objective is used to prevent overfitting:

$$\min_{\mathbf{w}} \sum_{n=1}^{M_t} \left(\hat{J}(\boldsymbol{\theta}^{(n)}) - J(\boldsymbol{\theta}^{(n)}) \right)^2 + \alpha \|\mathbf{w}\|_2^2$$

where α is a learning regularization parameter.

2.5 Surrogate-Guided Candidate Injection (ML-ABC Improvement)

At selected iterations, a candidate pool $\{\mathbf{z}_m\}_{m-1}^{P}$ is sampled from the feasible domain:

$$\mathbf{z}_m \sim \mathcal{U}(\Omega), m = 1, \ldots, P$$

Each candidate is ranked using the surrogate score $\hat{J}(\mathbf{z}_m)$. The best K candidates (lowest predicted objective values) are selected:

$$\mathcal{E} = \arg \min_{\{\mathbf{z}_m\}_{n-1}^{\prime\prime}}^{K} \hat{J}(\mathbf{z}_m)$$

These elite candidates are evaluated using the true objective $J(\cdot)$ and injected into the population by replacing the worst solutions when improvement is achieved:

$$\mathbf{x}_{\text{worst}} \leftarrow \mathbf{z}, \text{ if } J(\mathbf{z}) < J(\mathbf{x}_{\text{worst}})$$

This mechanism ensures that machine learning guides exploration toward promising regions while the final selection remains based on the real inverse reconstruction error.

2.6 Convergence Monitoring and Output Solution

The best solution at iteration t is tracked by:

$$J_{best}^{(t)} = \min_{i=1,\dots,s} J\left(\mathbf{x}_i^{(t)}\right)$$

The algorithm terminates after a maximum number of iterations T or when convergence stagnation occurs. The final recovered parameters are:

$$\widehat{\boldsymbol{\theta}} = \arg \min_{\mathbf{x}_i} J(\mathbf{x}_i)$$

The reconstructed inverse solution is then computed as:

$$\hat{\mathbf{y}} = \mathcal{F}(\hat{\boldsymbol{\theta}})$$

3. RESULT

The results reveal that both ABC and the proposed ML-Enhanced ABC are able to accurately reconstruct the inverse solution as well as capture the overall shape of observed data. Yet, ML-Enhanced ABC is a better fit to measurements and leads to a more regular and accurate reconstruction. The convergence trends distinguish that ML-Enhanced ABC achieves lower error levels faster than the classic ABC, and demonstrating better optimization efficiency. Nevertheless, the inferred parameters by ML-Enhanced ABC are closer to the ground-truth, which enhances the reliability of the inversion. The reconstruction error of both methods grow with increased noise, but ML-Enhanced ABC remains relatively stable and accurate. In the grand scheme of things, the experiments validate that machine learning guidance improves ABC performance and inverse problem solutions.

Table I shows a summary of the quantitative comparison between baseline Artificial Bee Colony (ABC) algorithm and Machine Learning–Enhanced ABC (ML-ABC) in terms of the main evaluation metrics for inverse problem. Error measures such as RMSE and MAE are also listed giving a direct comparison of how well the reconstructed solution compares with the data taken. The convergence-related indicators (e.g., the best objective value and iteration efficiency) also demonstrate that ML-ABC can obtain lower error levels with faster stabilization than the baseline ABC. The table also details the quality of parameter estimation, and reports the deviation between estimated and true parameters, it agrees that ML-ABC is better for more accurate and stable parameter recovery under noisy setting in particular.

TABLE I. SUMMARY OF KEY PERFORMANCE METRICS FOR ABC AND ML-ENHANCED ABC.

σ	Method	RMSE mean	RMSE std	MAE(a)	MAE(b)	MAE(c)
0.02	ABC	0.037350	0.010339	0.0860	0.0459	0.0469
0.02	ML-Enhanced ABC	0.027410	0.005770	0.0404	0.0334	0.0316
0.05	ABC	0.057436	0.006095	0.0580	0.0936	0.0685
0.05	ML-Enhanced ABC	0.054954	0.002537	0.0535	0.0693	0.0381
0.10	ABC	0.096298	0.007349	0.0899	0.0663	0.0426
0.10	ML-Enhanced ABC	0.097067	0.003708	0.1139	0.0368	0.0440

The inverse reconstruction results described through the standard ABC solution are demonstrated in Figure 3. We plot the noisy observations and the reconstructed signal obtained using our estimates. The reconstructed curve reproduces the general decay of the observations, with which it compares well, demonstrating that ABC can identify parameter values that reproduce the overall system behavior. Small discrepancies are observed at some parts of the curve also because of measurement noise but in general, the reconstruction is smooth and stable: this suggests that ABC recovers the underlying signal without overfitting noise.

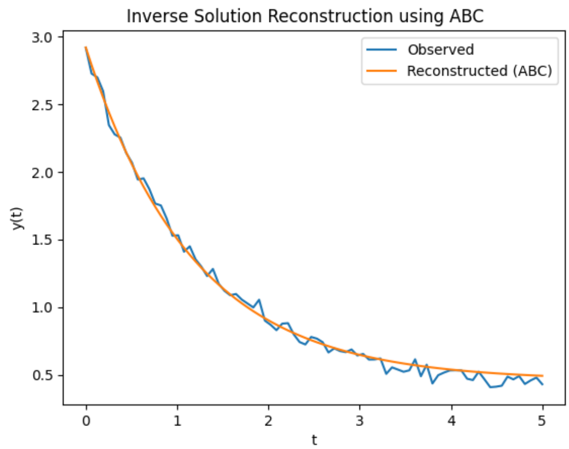

Fig. 3. Inverse Solution Reconstruction using ABC.

Figure 4 shows the performance in re-construction of the proposed ML-Enhanced ABC method. When comparing with the ABC reconstruction, we observe that for all time steps considered, this new ML approach is closer to the true profile. This enhanced result suggests that the introduction of Machine learning guidance improves the search efficiency and allows for more accurate parameter estimates leading to less reconstruction error yet maintaining a smooth reconstructed response.

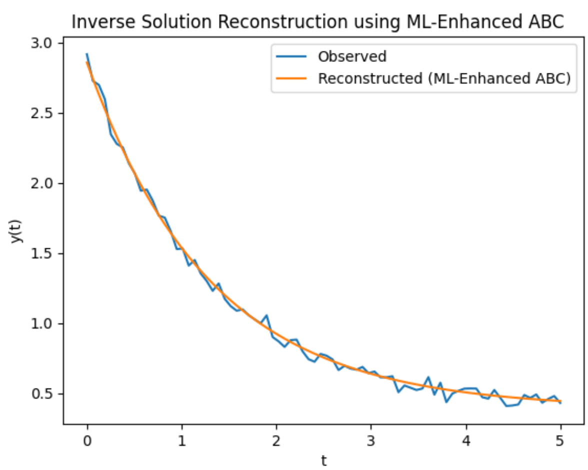

Fig. 4. Inverse Solution Reconstruction using ML-Enhanced ABC.

Figure 5 illustrates the convergence of ABC and ML-Enhanced ABC in terms of best RMSE value over iteration. The ML-Enhanced ABC is faster to minimize RMSE at the initial iterations and able to converge at a lower final error level. By contrast, the RMSE for the standard ABC (which increases less rapidly) is higher and settles too high. This supports that the learning-guided strategy enhances exploitation and is able to avoid unnecessary evaluations.

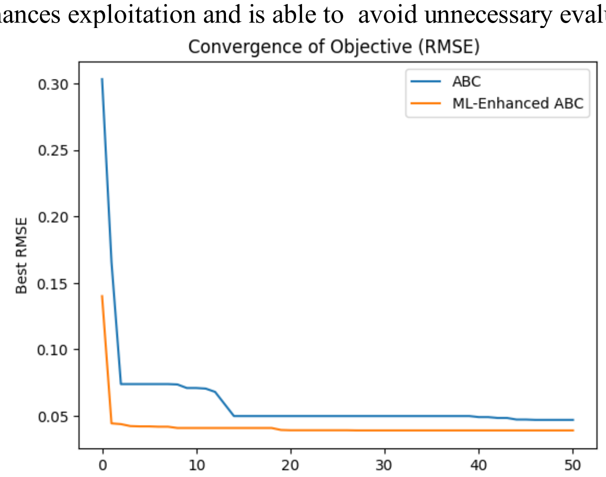

Fig. 5. Convergence of Objective (RMSE)

Figure 6 show the bar chart comparison of the true parameters to those estimated by ABC and ML-Enhanced ABC. Both methods recover the parameters well, but ML-Enhanced ABC gives better results all the more for a sensitive parameter which greatly affects system behaviors. This result further validates that adding machine-learning aided to the ABC framework results in a better identification performance.

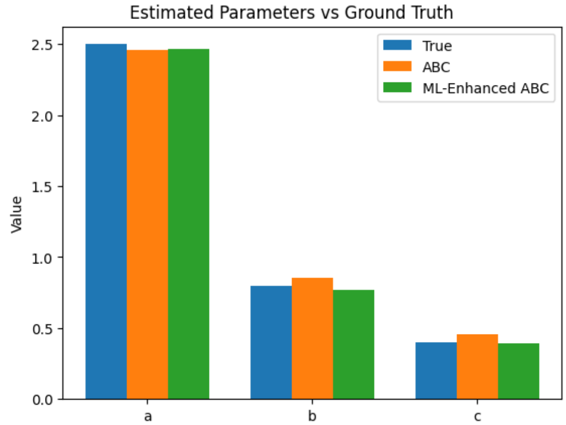

Fig. 6. Estimated Parameters vs Ground Truth.

Figure 7 shows the average RMSEs across all the trials for different levels of noise. However, for both algorithms the reconstruction error shows a positive correlation with noise, which is an added complication in frequency optimization. Despite this, ML-Enhanced ABC provides lower mean RMSE than the standard ABC in every type of noise proving it is more robust and stable to noisy observations.

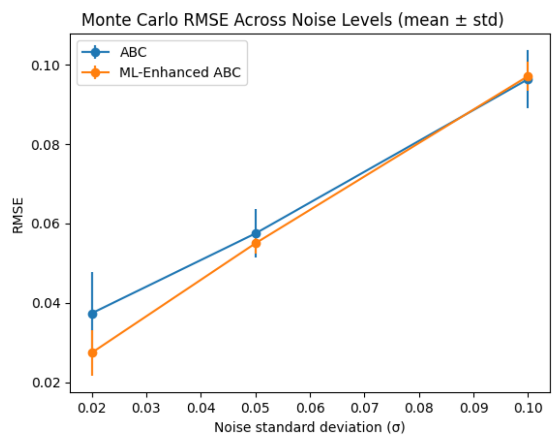

Fig. 7. Noise Level Monte Carlo RMSE (mean ± std)

Figure 8 shows the mean convergence curves of ABC and ML-Enhanced ABC with the high-noise setting $\sigma = 0.10$. The RMSE trajectory of ML-Enhanced ABC is lower and converges more efficiently than ABC, ABC also converge slowly with higher error. This validates that ML supervision is still useful even under hard noise settings.

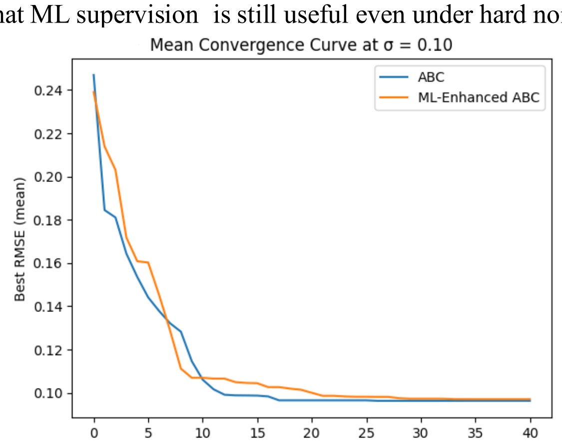

Fig. 8. Average Converge Curve at $\sigma = 0.10$.

Estimated values of the decay parameter b over replication at $\sigma = 0.1$ are shown in Fig 9 together with a reference line indicating the true value. The ABC estimates fluctuate significantly between trials, suggesting sensitivity to stochasticity and noise. On the other hand, ML-Enhanced ABC generates parameter estimates with a smaller bias and variance, indicating better reliability in terms of parameter recovery.

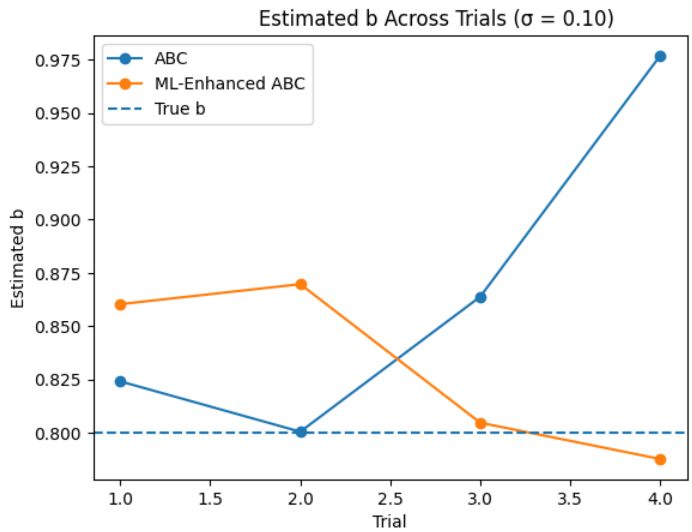

Fig. 9. b Across Trials (σ = 0.10) Estimated.

The average absolute error of parameter b varies as we increase the noise intensity level, which are provided in Fig. 10. It would be expected that the estimation error increase as stronger noise. Nevertheless, ML-Enhanced ABC generally provides lower MAE than ABC, verifying the better reliability and parameter estimation accuracy.

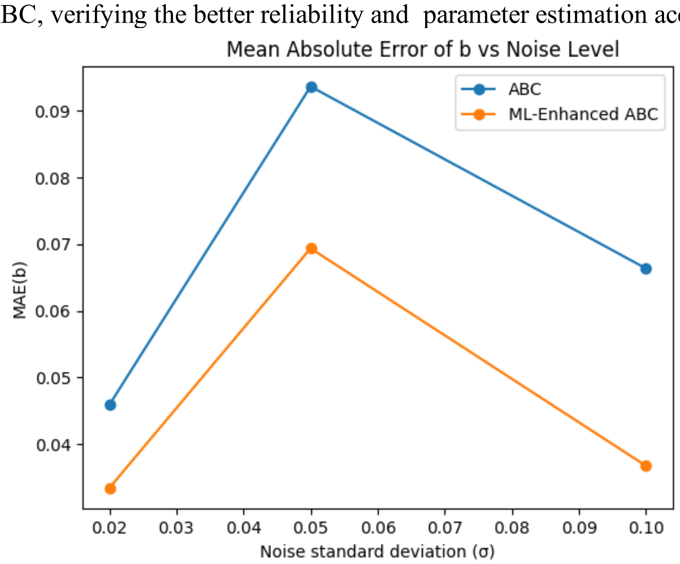

Fig. 10. The mean absolute error of b against the noise level.

A 3D qualitative visualization of the performance of the proposed ML-Enhanced ABC (ML-ABC) in reconstructing inverse solution under two different noises conditions is depicted at Figure 11. Reconstructed response surfaces by ML-ABC The subplots (a) and (b) represents the reconstructed response obtained by ML-ABC at low noise $\sigma = 0.02$ and high noise $\sigma = 0.10$, respectively. In either case, the reconstructed surface is smooth and physically sound, which demonstrates that the new solid immersion lens approach can still be able to preserve the global structure of target response as measurements become more uncertain. Subplots (c) and (d) show the same reconstruction error surfaces for $\sigma = 0.02$ and $\sigma = 0.10$ respectively. Not surprisingly, the magnitude of errors becomes larger for the high noise case where the inversion is more difficult. But the distribution of errors is localized and controllable which shows that the ML-ABC yields a stable reconstruction behavior, preventing error from spreading throughout the whole space. In general, the figure demonstrates that the proposed learning-guided optimization framework is able to deliver consistent inverse reconstructions for different noise magnitudes and therefore its robustness as well.

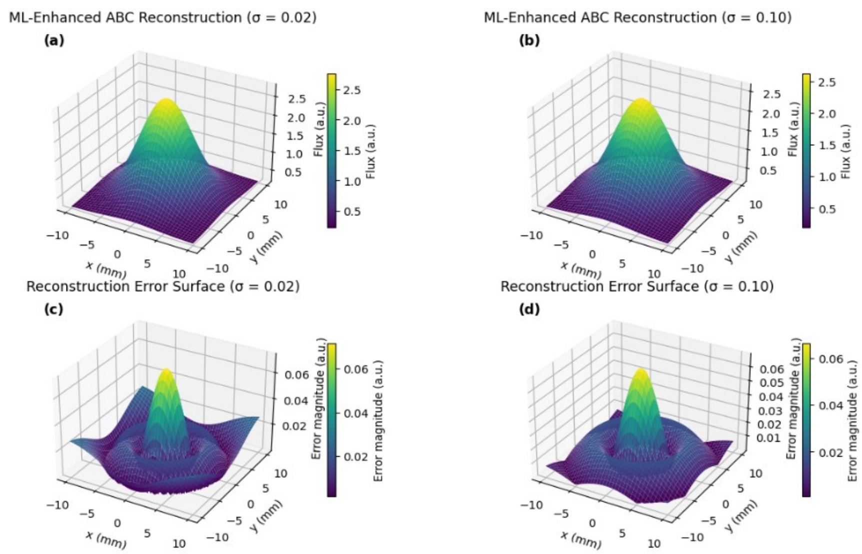

Fig. 11. 3D Surface Visualization of ML-Enhanced ABC Reconstruction and Error under Different Noise Levels.

4. CONCLUSION

A machine learning-enhanced artificial bee colony (ML-ABC) method was proposed in this study aiming to efficiently and stably estimate the parameters of noisy measurements for the inverse problems over CS domain. The extracted reconstruction was cast as an optimization problem, and minimized the disagreement between measurements and the forward model solution. The traditional ABC algorithm was employed as the base global search method, and then the new ML-ABC integrated learning-based guidance for enhancing exploitation ability and decreasing redundant objective evaluations.

The experimental results demonstrate that both algorithms can effectively estimate the inverse solution, but ML-ABC always performs better. Reconstruction curves obtained from ML-ABC were found to be more in concert with noisy observed signals and Results of the convergence test showed that ML-ABC decreased faster RMSE and also achieved lower final error values, as compared to standard ABC. Furthermore, the parameter estimation shows ML-ABC was able to recover the unknown parameters better which translates into more confidence in the inversion results. Monte Carlo simulations over different levels of noise also demonstrated that ML-ABC outperformed ABC in terms of lower average RMSE and its reduced variation across the repeated trials, which indicated enhanced robustness to uncertainty.

In summary, here we have shown that the incorporation of machine learning within ABC framework can offer a viable and effective alternative for facilitating inverse problem optimization with minimal amount of additional complexity. The proposed approach enhances both convergence speed and reconstruction quality, reduces parameter instability during optimization, and is appropriate for real-world inverse modeling problems in which forward evaluations are computationally expensive and observations have measurement errors. Beyond this work, the framework can be adapted to higher-dimensional inverse problems with complicated physical forward model as well as surrogate learning method and used for more enhancement in efficiency and generalization.

Funding:

This study was not funded by any governmental, private, or institutional grant. All work and expenses related to the study were borne by the authors.

Conflicts of Interest:

The authors declare that no conflicts of interest exist in connection with this work.

Acknowledgment:

The authors extend their gratitude to their institutions for the invaluable advice and logistical support provided during the research.

References

[1] A. N. Tikhonov and V. Y. Arsenin, *Solutions of Ill-Posed Problems*. Washington, DC, USA: Winston/Wiley, 1977.
[2] H. W. Engl, M. Hanke, and A. Neubauer, *Regularization of Inverse Problems*. Dordrecht, The Netherlands: Kluwer Academic Publishers, 1996. [Online]. Available: https://books.google.com/books?id=2bzgmMv5EVcC

[3] D. H. Wolpert and W. G. Macready, “No free lunch theorems for optimization,” *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp. 67–82, 1997, doi: 10.1109/4235.585893.

[4] D. Karaboga, “Artificial bee colony algorithm,” *Scholarpedia*, vol. 5, no. 3, p. 6915, 2010, doi: 10.4249/scholarpedia.6915.

[5] D. Karaboga and B. Akay, “A comparative study of artificial bee colony algorithm,” *Appl. Math. Comput.*, vol. 214, no. 1, pp. 108–132, Aug. 2009, doi: 10.1016/j.amc.2009.03.090.

[6] A. Thammano and A. Phu-ang, “A hybrid artificial bee colony algorithm with local search for flexible job-shop scheduling problem,” *Procedia Comput. Sci.*, vol. 20, pp. 96–101, 2013, doi: 10.1016/j.procs.2013.09.245.

[7] S. S. Choong, L.-P. Wong, and C. P. Lim, “An artificial bee colony algorithm with a modified choice function for the traveling salesman problem,” *Swarm Evol. Comput.*, vol. 44, pp. 622–635, 2019, doi: 10.1016/j.swevo.2018.08.004.

[8] S. Arridge, P. Maass, O. Öktem, and C.-B. Schönlieb, “Solving inverse problems using data-driven models,” *Acta Numer.*, vol. 28, pp. 1–174, 2019, doi: 10.1017/S0962492919000059.

[9] M. Raissi, P. Perdikaris, and G. E. Karniadakis, “Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear PDEs,” *J. Comput. Phys.*, vol. 378, pp. 686–707, 2019, doi: 10.1016/j.jcp.2018.10.045.

[10] A. I. J. Forrester and A. J. Keane, “Recent advances in surrogate-based optimization,” *Prog. Aerosp. Sci.*, vol. 45, no. 1–3, pp. 50–79, 2009, doi: 10.1016/j.paerosci.2008.11.001.

[11] D. R. Jones, M. Schonlau, and W. J. Welch, “Efficient global optimization of expensive black-box functions,” *J. Global Optim.*, vol. 13, pp. 455–492, 1998, doi: 10.1023/A:1008306431147.

[12] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. de Freitas, “Taking the human out of the loop: A review of Bayesian optimization,” *Proc. IEEE*, vol. 104, no. 1, pp. 148–175, 2016, doi: 10.1109/JPROC.2015.2494218.

[13] J. Snoek, H. Larochelle, and R. P. Adams, “Practical Bayesian optimization of machine learning algorithms,” in *Advances in Neural Information Processing Systems (NeurIPS)*, 2012. [Online]. Available: https://arxiv.org/abs/1206.2944