International Journal of Technology

http://ijtech.eng.ui.ac.id

Research Article

A Zeroth-Order Stochastic Gradient Descent Method for Communication-Efficient Federated Learning

Hodaka Nishi[1], Shiro Yano[2], Megumi Miyashita[1,*], Shunta Onishi[1], Yuta Goto[1], Toshiyuki Kondo[1]

[1]Department of Electrical Engineering and Computer Science, Tokyo University of Agriculture and Technology, 2-24-16 Naka-cho, Koganei-shi, Tokyo 184-8588, Japan;

[2]InfoTech Div., Toyota Motor Corporation, Otemachi Bldg. 6F, 1-6-1 Otemachi, Chiyoda-ku, Tokyo, 100-0004, Japan;

[*]Corresponding author: mmiyashita@go.tuat.ac.jp; Tel.: +81 42-388-7699; Fax: +81 42-388-7699

Abstract: Federated learning (FL) has emerged as a key paradigm for decentralized data privacy-preserving machine learning. However, substantial communication costs often hinder its practical application, especially as deep learning models scale to millions or billions of parameters. This communication bottleneck becomes particularly acute in heterogeneous networks with clients who are resource-constrained. To address this challenge, this study proposes a novel FL framework that leverages black-box optimization, specifically the zeroth-order (ZO) method, to reduce communication overhead. The proposed method, named ZO-FedSGD, reframes the learning process to eliminate the need for transmitting high-dimensional model parameters. Instead, each communication round involves exchanging only a constant number of scalar values, including a random seed and function evaluations, making the communication cost independent of the model size. Extensive experiments were conducted to compare ZO-FedSGD with the existing FedAvg algorithm on the MNIST datasets. The evaluation focused on model accuracy and total communication efficiency. Our results reveal a trade-off: ZO-FedSGD required more rounds to converge and achieved a slightly lower final accuracy. However, it demonstrated superior communication efficiency—to reach 90% accuracy, ZO-FedSGD required approximately 10^4 communicated parameters, compared to 10^6 for FedAvg, representing a two-order-of-magnitude reduction. In conclusion, this study validates ZO-FedSGD as a viable and highly efficient alternative for FL in communication-constrained scenarios. It offers a new direction for designing scalable FL systems and a promising solution to the statistical heterogeneity problem.

Keywords: Black-box optimization; Federated learning; Two-point estimation

1. Introduction

Federated learning (FL) (Zhang et al., 2021; Q. Yang et al., 2019; McMahan et al., 2017) is a machine learning (ML) framework that trains models on decentralized data without the need for data aggregation. By exchanging only model updates between clients and a central server, FL facilitates distributed learning while reducing communication costs associated with data transmission and preserving data privacy (Wei et al., 2020). Consequently, it has been applied to fields where ML was previously challenging due to privacy concerns, such as healthcare (Guan et al., 2024; Teo et al., 2024; Nguyen et al., 2023; Kairouz et al., 2021; Rieke et al., 2020), the Internet of Things (IoT) (Dritsas and Trigka, 2025; Jiang et al., 2025; Y. Yang et al., 2025; Nguyen et al., 2021), and so on (L. Li et al., 2020).

However, in recent years, the scale of deep learning models has grown, leading to a significant increase in the number of parameters. For instance, while the BERT model (Devlin et al., 2019) introduced in 2018 had 340 million parameters, the DeepSeek-R1 model (Guo et al., 2025) released in 2025 has 671 billion parameters. Exchanging several hundred billion parameters,

https://doi.org/10.14716/ijtech.v17i1.8195
Received June 2025; Revised November 2025; Accepted December 2025

each represented by a 32-bit value, among hundreds of clients would result in communication loads ranging from tens of gigabytes to tens of terabytes for each round, even when only the model parameters are transmitted. In practice, this is not feasible (S. Wang et al., 2019). Researchers have explored several techniques for reducing communication load to address this challenge. These include quantization, which lowers the numerical precision of the parameters (Reisizadeh et al., 2020), and sparsification, which transmits only the most important updates (J. Li et al., 2024); and knowledge distillation, which exchanges compact knowledge representations instead of full model parameters (Wu et al., 2022). However, all of these methods still require sending some form of parameter-related information and, therefore, remain inherently dependent on the model size. In addition, recent studies have actively investigated the integration of black-box optimization (BBO) with FL as a promising direction for achieving communication efficiency (Ma et al., 2025; Z. Li et al., 2024). For example, DeComFL (Z. Li et al., 2024) leveraged zeroth-order (ZO) optimization to eliminate the dependence of the communication cost on the model dimensionality, thereby reducing the per-round cost from O(d) to O(1). Such research highlights the potential of BBO-based methods in alleviating large-scale FL communication bottlenecks. Current ZO-based FL methods either require complex perturbation schemes or do not perform experiments under non-IID conditions.

Building on this line of work, the present study proposes a fundamental method that combines ZO optimization with stochastic gradient descent (SGD), one of the most basic yet widely used optimization techniques. In particular, we propose the ZO optimization-driven FedSGD (ZO-FedSGD) method, in which clients use ZO gradient estimators instead of true gradients. ZO-FedSGD aims to reduce communication costs while maintaining model accuracy. Through comparative experiments with the established FedAvg method (McMahan et al., 2017), we demonstrate that ZO-FedSGD can achieve stable convergence even under stringent communication constraints. The novelty of this study lies in its simple and communication-efficient ZO optimization-based algorithm. This is particularly useful for privacy-sensitive federated learning scenarios where communication resources are constrained, such as IoT devices developed in the field of tiny ML (Lin et al., 2023). In practice, ZO-FedSGD can be implemented by replacing the local backpropagation with a two-point function evaluation on each client, sending the estimated gradients to the server, and aggregating them as in the standard FedSGD.

2. Methods

In this study, we propose ZO-FedSGD that adapts the zero-order optimization to FL. The fundamental idea is to replace the high-dimensional model parameter vectors communicated by methods such as FedSGD with model-size-independent scalar information based on the principles of ZO methods.

2.1 Problem Formulation

In FL, the global objective function $F:R^d \rightarrow R$ can be defined as the weighted average of all clients' loss function as shown in Equation (1):

$$F(W) = \frac{1}{n} \sum_{k=1}^{K} n_k \; f_k \; (w), \tag{1}$$

where w is the d-dimensional model parameter vector, K is the total number of clients, $f_k:R^d \rightarrow R$ is the loss function on client k, n_k is the number of data points on client k, and n is the total number of data points. ZO-FedSGD optimizes its global objective function by treating it as a black-box function whose gradient is not computable.

2.2 Algorithm

In ZO-FedSGD, we find the optimal parameter x for this problem by applying two-point estimation. The update equation for the two-point estimation used in our approach is shown in Equation (2):

$$w_t = w_{t-1} - \alpha r_t$$

$$r_t \sim \mathcal{N}_d(0, \sigma^2 I) \tag{2}$$

$$\alpha_t = \eta \frac{n_k f_k(w_{t-1} + r_t) - n_k f_k(w_{t-1} - r_t)}{n\|r_t\|}$$

where wt is the model parameter vector at round t, and η is the learning rate. r_t is a d-dimensional random vector sampled from a d-dimensional normal distribution, whose covariance matrix is expressed as σ^2I using the identity matrix I. The term σ^2 is the variance of the normal distribution and is a hyperparameter that adjusts the dispersion of each element in the random vector. The term $\alpha_t r_t$ corresponds to the approximate gradient in the ZO method. The server calculates the update coefficient α_t, which aggregates the function values $f_k(w_{t-1} + r_t)$ and $f_k(w_{t-1} - r_t)$ evaluated on each client.

The ZO-FedSGD algorithm is based on FedSGD. Its most distinctive feature is that it communicates only four items instead of directly communicating high-dimensional gradient vectors: two scalar evaluation values calculated by each client, a random seed for sharing random vectors, and the global model update coefficient α_t. Consequently, the communication cost for model updates becomes O(1) independent of the model size. Figure 1 shows an overview of ZO-FedSGD. The detailed procedure of this process is shown in Algorithms 1 and 2.

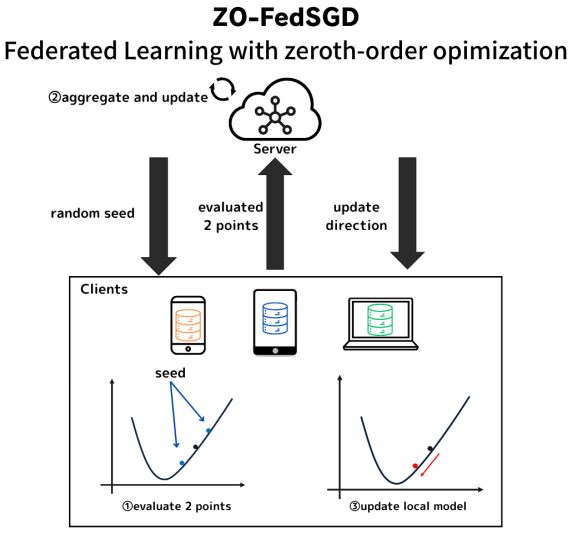


Figure 1 Overview of ZO-FedSGD

Algorithm 1: The Proposed ZO-FedSGD Method (Server-side)

Inputs : The number of clients K, the learning rate η, the number of rounds t
Server Executes:
Initialization: $w^0 \leftarrow$ initial global model
$\forall k: \ w_k^0 \leftarrow w^0$
for $t = 1, 2, \ldots$ **do**
  foreach $k \in K$ in parallel **do**
    $l_{k,t}^+, \ l_{k,t}^- \leftarrow$ ClientEvaluate(seed$_t$)
  end
  Generate random vector r_t with seed seed$_t$
  $l_t^+ = \sum_{k \in K} \frac{n_k}{n} l_{k,t}^+, \quad l_t^- = \sum_{k \in K} \frac{n_k}{n} l_{k,t}^-$
  $\alpha_t = \text{CalculateAlpha}(l_t^+, l_t^-)$
  $w_t = w_{t-1} + \alpha_t r_t$ `// ▷ update global model`
  foreach $k \in K$ in parallel **do**
    ClientUpdate(α_t, seed$_t$)
  end
end
Function *CalculateAlpha*(l_t^+, l_t^-)**:**
  $\alpha_t = \eta \dfrac{n_k l_t^+ - n_k l_t^-}{n \|r_t\|}$
  return α_t

Algorithm 2: The Proposed ZO-FedSGD Method (Client-side)

Function *ClientEvaluate*(*seed$_t$*)**:**
  Generate random vector r_t with seed seed$_t$
  $l_{k,t}^+ = f_k\left(w_{t-1}^k + r_t\right)$
  $l_{k,t}^- = f_k\left(w_{t-1}^k - r_t\right)$
  return $l_{k,t}^+, \ l_{k,t}^-$
Function *ClientUpdate*(α_t, *seed$_t$*)**:**
  Generate random vector r_t with seed seed$_t$
  $w_t^k = w_{t-1}^k + \alpha_t \cdot r_t$ `// ▷ update local model`

In each round, the server first sends a random seed, seed$_t$, to all clients. Each client uses this seed to generate a random vector r_t. Then, the client evaluates its local loss function f_k on its local model w_{t-1} at two points, w_{t-1}^k+r_t and w_{t-1}^k-r_t, and sends the resulting loss values, $l_{k,t}^+$ and $l_{k,t}^-$, back to the server. The server computes a weighted average of these values from all clients, weighted by the number of data points n_k for each client. Using the aggregated results, l_t^+ and l_t^-*s* the server calculates the update coefficient α_t and updates the global model. In two-point estimation, if the two obtained loss values are nearly the same, the next parameter could be nearly the same as the previous parameter. Conversely, if these have a large difference, the value is updated to worsen the direction (Liu et al., 2018; Nesterov and Spokoiny, 2017).

In our preliminary experiments using Algorithms 1 and 2, we encountered a divergent learning process. To stabilize the training, we introduce a stabilized version of CalculateAlpha, which gates the update by explicitly comparing three losses; l_t^+, l_t^- and the previous loss l_{t-1}. The server selects the direction with the smallest loss (setting $\alpha_t \in$+1,-1) and suppresses the update (α_t=0) whenever neither perturbation improves upon loss l_{t-1}. This modification is detailed in Algorithms 3. This approach prevents updates that degrade the solution, thereby achieving stable learning. When $\alpha_t \in$+1,-1, the next model parameter is set to the best-evaluated parameter from the previous step. In this sense, the update rule can be viewed as a special case of the CE method (Rubinstein and Kroese, 2004; Rubinstein and Kroese, 2019), which makes the approach reasonable.

Algorithm 3: The Stabilized *CalculateAlpha* Function

Function *CalculateAlpha*(l^+, l^-, l_{t-1})**:**
  $\alpha_t = \{-1, +1, 0\}$,
  if $l_t^+ = \min(l^+, l^-, l_{t-1})$ **then**
    $\alpha_t = 1$
  end
  else if $l_t^- = \min(l^+, l^-, l_{t-1})$ **then**
    $\alpha_t = -1$
  end
  else if $l_{t-1} = \min(l^+, l^-, l_{t-1})$ **then**
    $\alpha_t = 0$
  end
  return α_t

3. Result and Discussion

In our evaluation experiments, we conducted a performance comparison between ZO-FedSGD and FedAvg, a conventional FL method, which served as the baseline.

3.1 Experimental Setting

We used the MNIST dataset (60,000 training images and 10,000 test images) (LeCun et al., 1998) for our experiments and utilized FedScale (Lai et al., 2022) as the execution environment. For the model, we employed a LeNet (LeCun et al., 1989)-based CNN, which is the default implementation in FedScale.

The data distribution was evaluated under two settings: IID and non-IID. For the IID condition, the training data were distributed randomly and equally among all clients. In contrast, for the non-IID condition, the data were partitioned such that each client holds training data consisting of only two label types.

To evaluate statistical variations and ensure the reliability of our results, each experimental condition was executed three times with different initial random seeds. The mean performance and confidence interval (CI)s across these multiple runs demonstrate the statistical significance of our findings.

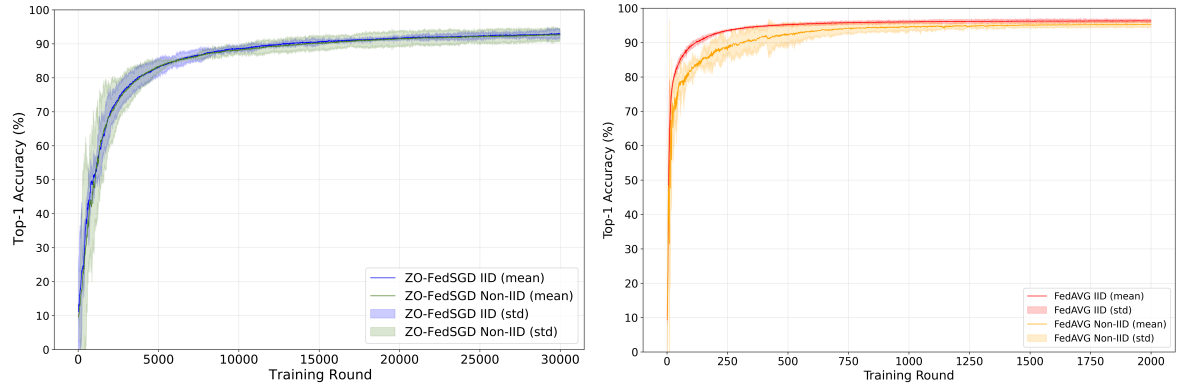
The evaluation metrics are prediction accuracy (i.e., classification accuracy on the test data) and communication cost (i.e., the total cost required to reach a specific target accuracy). The communication cost for FedAvg is calculated as 2 × (number of model parameters) × (number of rounds) because the model is communicated twice per round. The cost for ZO-FedSGD is calculated as 4 × (number of rounds) because it involves the communication of a random seed, two evaluated values, and the update coefficient α in each round. The experimental conditions are detailed in Table 1. The "Data Used per Round" in each experiment is denoted as (number of updates per round) × (minibatch size).

Table 1 Summary of experimental settings. The proposed ZO-FedSGD is evaluated against FedAvg under IID and Non-IID settings

Condition	Method	Data Distribution	Number of Clients	Data distributed per Client	Data Used per Round
Condition1	ZO-FedSGD	IID	10	6000	1 × 6000
Condition2	ZO-FedSGD	Non-IID	10	6000	1 × 6000
Condition3	FedAvg	IID	10	6000	3 × 2000
Condition4	FedAvg	Non-IID	10	6000	3 × 2000

3.2 Prediction Accuracy

Figure 2 shows the prediction accuracy results for the four experimental conditions. Figure 2 plots the number of rounds on the horizontal axis and the achieved accuracy on the vertical axis. As shown in Figure 2, FedAvg reached over 95% accuracy within 2,000 rounds. In contrast, ZO-FedSGD accuracy remained at approximately 92% even after 30,000 rounds. Furthermore, the CIs across three independent runs with different initial seeds demonstrate that both algorithms exhibit stable and consistent convergence behavior, with relatively small performance variance across different initializations. This statistical consistency validates the reliability of our experimental results and indicates that random initialization effects do not cause the observed performance differences.



(a) (b)

Figure 2 Top-1 test accuracy for MNIST classification under IID and Non-IID data distributions: (a) Top-1 test accuracy of the ZO-FedSGD algorithm. (Conditions 1, 2) (b) Top-1 test accuracy of the FedAvg algorithm. (Conditions 3, 4)

The lower final accuracy of ZO-FedSGD can be attributed to the inherent low learning efficiency of ZO methods and the algorithm employed in this experiment, which restricts the update coefficient α_t to one of three values (-1, 0, or 1). With the fixed update coefficient for the search, the model is unable to make fine adjustments in the vicinity of the optimal solution, which may have led to stagnated convergence. We expect that this accuracy limitation could be overcome by introducing more advanced BBO techniques, such as Bayesian optimization (X. Wang et al., 2023), which balances exploration and exploitation, or by using ZO methods with adaptive learning rates (Chen et al., 2019).

Although not included in the comparison conditions in this experiment, we found that setting the hyperparameter σ^2 for the variance of the multidimensional normal distribution used in two-point estimation to a smaller value slowed down the learning speed but tended to improve the maximum accuracy across all rounds. Therefore, it is expected that a more accurate model can be obtained by determining an appropriate σ^2 for a predetermined number of rounds.

3.3 Communication Cost

Figure 3 and Table 2 show the results for the communication cost of ZO-FedSGD and FedAvg. Figure 3 plots the achieved accuracy on the horizontal axis against the total number of communicated parameters required to reach that accuracy on the vertical axis. The quantitative comparisons in Table 2 demonstrate that ZO-FedSGD has a significant advantage in communication efficiency. For example, ZO-FedSGD required only 50,720 communicated parameters to achieve 90% accuracy under IID conditions, whereas FedAvg required 5,285,280. This two-order-of-magnitude reduction in total communication is achieved by our method, which utilizes zeroth-order optimization to reduce the per-round communication volume to O(1) making it independent of the model size. However, as Table 2 also indicates, this efficiency entails a trade-off. ZO-FedSGD required a much larger number of rounds to converge, and its final accuracy

was lower than that of FedAvg. To reach the same 90% accuracy (IID), ZO-FedSGD required 3,680 rounds, whereas FedAvg required only 121 rounds. This reflects the fundamental difference in learning efficiency between the first-order methods, which use the true gradient, and the ZO methods, which approximate the gradient using only the limited information from function evaluations. ZO-FedSGD offers a compelling solution for resource-constrained federated environments where communication bandwidth is the primary bottleneck that dramatically reduces the communication burden while maintaining competitive accuracy levels.

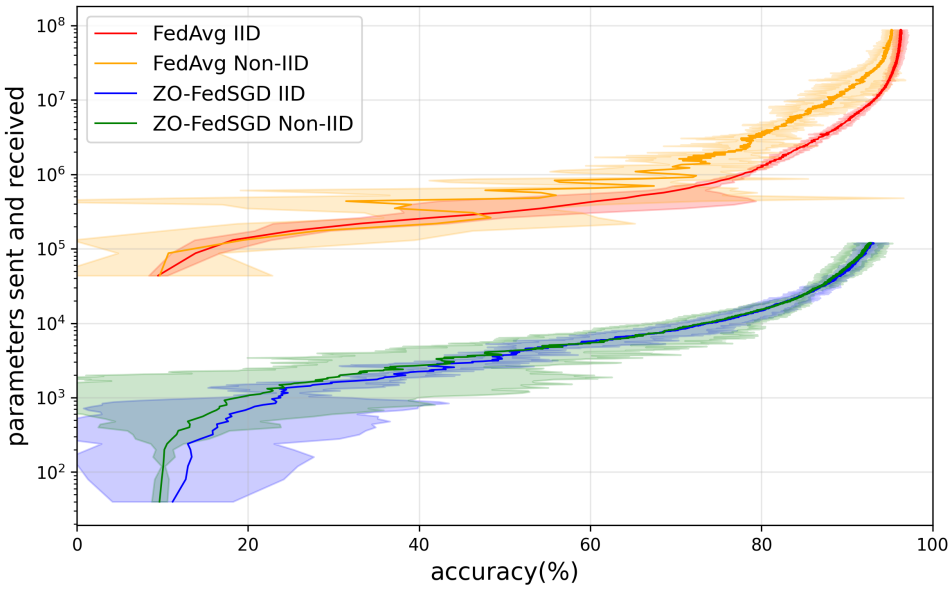


Figure 3 Communication cost of the ZO-FedSGD and FedAvg algorithms required to reach a target Top-1 test accuracy on the MNIST dataset. The plot compares the cumulative number of communicated parameters under both IID and non-IID data distributions, with each curve representing the average of three independent runs. (Conditions 1, 2, 3, and 4)

Table 2 Number of rounds and communication cost required for ZO-FedSGD and FedAvg to achieve target Top-1 test accuracy on MNIST under IID and non-IID settings. (Conditions 1, 2 and 4)

Condition	Accuracy(%)	Round	Communicated params
ZO-FedSGD IID	70.0	2000	8,000
	80.0	3680	14,720
	90.0	12680	50,720
ZO-FedSGD Non-IID	70.0	2090	8,360
	80.0	3820	15,280
	90.0	14050	56,200
FedAvg IID	70.0	14	611,520
	80.0	29	1,266,720
	90.0	121	5,285,280
FedAvg Non-IID	70.0	21	917,280
	80.0	73	3,188,640
	90.0	304	13,278,720

4. Conclusions

In this study, we proposed and experimentally evaluated ZO-FedSGD based on the ZO method to address the challenge of communication costs in FL associated with the increasing scale of models. The experimental results revealed a clear trade-off between communication and learning efficiency when comparing our proposed method to the existing FL approach, FedAvg. ZO-FedSGD required a higher number of rounds to converge and did not reach the same final model accuracy as FedAvg. However, in terms of the total communication cost required to achieve a certain level of accuracy, the proposed method demonstrated overwhelming efficiency. Specifically, achieving 90% accuracy on MNIST required approximately 10^4 communicated parameters with ZO-FedSGD, while FedAvg required around 10^6, representing a reduction of roughly two orders of magnitude. This substantial reduction highlights that ZO-FedSGD is a viable and practical option in environments where communication resources are severely constrained. In conclusion, this study presents a new direction for breaking through the limitation of communication efficiency in FL. It is expected to contribute to the future development of FL technologies for large-scale models and to efforts addressing the data heterogeneity problem. Nevertheless, this study did not experiment with additional datasets, such as CIFAR-10 or Fashion-MNIST, or with more complex architectures, such as CNN-4, ResNet-8, and MLP-3. Extending the evaluation in these directions represents a key for future research. Future work will focus on improving accuracy by incorporating advanced BBO methods, such as adaptive noise, variance reduction, and Bayesian sampling. Additionally, we plan to verify the scalability of our approach with regard to the number of clients and model size.

Acknowledgements

This work was supported by JSPS KAKENHI Grant Number JP25K03018.

Author Contributions

Conceptualization, S.Y.; methodology, S.Y.; software, S.O., Y.G and M.M.; formal analysis, S.Y., S.O. and M.M.; investigation, S.O. and Y.G.; resources, T.K.; data curation, M.M. and S.O.; writing—original draft preparation, H.N., M.M and S.O.; writing—review and editing, H.N., M.M., S.Y. and T.K. ; visualization, H.N. and S.O.; supervision, S.Y., M.M. and T.K.; project administration, S.Y., M.M and T.K.; funding acquisition, T.K. All authors have read and agreed to the published version of the manuscript.

Conflict of Interest

The authors declare that they have no conflict of interest.

Supplementary Materials

S1. Federated Learning (FL)

FL is a distributed ML framework proposed by McMahan et al., 2017. Conventional ML aggregates all data on a server for training. On the other hand, FL only exchanges information, e.g., model parameters, without aggregating clients' raw data. This approach is expected to enhance data privacy and reduce communication load.

S2. Black-Box Optimization

BBO (Y. Wang et al., 2018; Golovin et al., 2017; Hansen et al., 2010) is a general term for optimization problems that do not use analytical information about an objective function, such as its gradient. It must be optimized using only the objective function value. An example of BBO is the hyperparameter optimization of ML models (Turner et al., 2021; Feurer and Hutter, 2019).

ZO methods (Liu et al., 2020; Golovin et al., 2019) are optimization techniques that solve BBO problems by approximating the gradient of the objective function $f : Rd \rightarrow R$ using finite differences. ZO methods are inspired by gradient-based optimization methods. ZO methods approximate the gradient from evaluations at multiple randomly perturbed points. As an example, one of the update equations for two-point estimation is shown in Equation (S1).

$$\hat{\nabla} f(x_t) = \frac{f(x_t + \mu u) - f(x_t - \mu u)}{2\mu} u, \tag{S1}$$

where $x_t \in R^d$ is current parameter, is a hyperparameter, and u is a random direction vector sampled from a standard normal distribution. $\hat{\nabla} f(x_t)$ denotes the approximated gradient at the point xt. The two-point estimation evaluates points $x_t+\mu u$ and x_t-μu, then approximates the gradient direction and magnitude from their difference. ZO methods tend to exhibit slower convergence because they cannot utilize the true gradient. However, they provide a powerful solution for problems in which gradient computation is infeasible (Ghadimi and Lan, 2013).

Declaration of AI

Generative AI tools (e.g., ChatGPT, OpenAI) were used to improve the language and grammar of this manuscript. The authors reviewed and took full responsibility for the content of the manuscript.

References

Chen, X., Liu, S., Xu, K., Li, X., Lin, X., Hong, M., & Cox, D. (2019). Zo-adamm: Zeroth-order adaptive momentum method for black-box optimization. *Advances in Neural Information Processing Systems*, *32*. https://doi.org/10.48550/arXiv.1910.06513

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. In J. Burstein, C. Doran, & T. Solorio (Eds.), *Proceedings of the 2019 conference of the north american chapter of the association for computational linguistics: Human language technologies, volume 1 (long and short papers)* (pp. 4171–4186). Association for Computational Linguistics. https://doi.org/10.18653/v1/N19-1423

Dritsas, E., & Trigka, M. (2025). Federated learning for iot: A survey of techniques, challenges, and applications. *Journal of Sensor and Actuator Networks*, *14*(1), 9. https://doi.org/10.3390/jsan14010009

Feurer, M., & Hutter, F. (2019). Hyperparameter optimization. In F. Hutter, L. Kotthoff, & J. Vanschoren (Eds.), *Automated machine learning: Methods, systems, challenges* (pp. 3–33). Springer International Publishing. https://doi.org/10.1007/978-3-030-05318-5_1

Ghadimi, S., & Lan, G. (2013). Stochastic first- and zeroth-order methods for non-convex stochastic programming. *arXiv preprint*. https://doi.org/10.48550/arXiv.1309.5549

Golovin, D., Karro, J., Kochanski, G., Lee, C., Song, X., & Zhang, Q. (2019). Gradientless descent: High-dimensional zeroth-order optimization. *arXiv preprint*. https://doi.org/10.48550/arXiv.1911.06317

Golovin, D., Solnik, B., Moitra, S., Kochanski, G., Karro, J., & Sculley, D. (2017). Google vizier: A service for black-box optimization. *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1487–1495. https://doi.org/10.1145/3097983.3098043

Guan, H., Yap, P.-T., Bozoki, A., & Liu, M. (2024). Federated learning for medical image analysis: A survey. *Pattern Recognition*, *151*, 110424. https://doi.org/10.1016/j.patcog.2024.110424

Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., et al. (2025). Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint*. https://doi.org/10.48550/arXiv.2501.12948

Hansen, N., Auger, A., Ros, R., Finck, S., & Posik, P. (2010). Comparing results of 31 algorithms from the black-box optimization benchmarking bbob-2009. *Proceedings of the 12th Annual Conference Companion on Genetic and Evolutionary Computation*, 1689–1696. https://doi.org/10.1145/1830761.1830790

Jiang, Z., Chua, F.-F., & Lim, A. H.-L. (2025). Privacy-preserving data uploading scheme based on threshold secret sharing algorithm for internet of vehicles. *International Journal of Technology*, *16*(3), 731–747. https://doi.org/10.14716/ijtech.v16i3.7260

Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A. N., et al. (2021). Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, *14*(1–2), 1–210. https://doi.org/10.1561/2200000083

Lai, F., Dai, Y., Singapuram, S., Liu, J., Zhu, X., Madhyastha, H., et al. (2022). Fedscale: Benchmarking model and system performance of federated learning at scale. *International Conference on Machine Learning*, 11814–11827. https://doi.org/10.1145/3477114.3488760

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., et al. (1989). Backpropagation applied to handwritten zip code recognition. *Neural Computation*, *1*(4), 541–551. https://doi.org/10.1162/neco.1989.1.4.541

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*(11), 2278–2324. https://doi.org/10.1109/5.726791

Li, J., Zhang, Y., Li, Y., Gong, X., & Wang, W. (2024). Fedsparse: A communication-efficient federated learning framework based on sparse updates. *Electronics*, *13*(24), 5042. https://doi.org/10.3390/electronics13245042

Li, L., Fan, Y., Tse, M., & Lin, K. Y. (2020). A review of applications in federated learning. *Computers & Industrial Engineering*, *149*, 106854. https://doi.org/10.1016/j.cie.2020.106854

Li, Z., Ying, B., Liu, Z., Dong, C., & Yang, H. (2024). Achieving dimension-free communication in federated learning via zeroth-order optimization. *arXiv preprint*. https://doi.org/10.48550/arXiv.2405.15861

Lin, J., Zhu, L., Chen, W., Wang, W., & Han, S. (2023). Tiny machine learning: Progress and futures [feature]. *IEEE Circuits and Systems Magazine*, *23*(3), 8–34. https://doi.org/10.1109/MCAS.2023.3302182

Liu, S., Chen, P., Kailkhura, B., Zhang, G., Hero, A. O., & Varshney, P. K. (2020). A primer on zeroth-order optimization in signal processing and machine learning: Principles, recent advances, and applications. *IEEE Signal Processing Magazine*, *37*(5), 43–54. https://doi.org/10.1109/MSP.2020.3003837

Liu, S., Chen, P., Zhu, W., & Carin, L. (2018). Zeroth-order stochastic variance reduction for nonconvex optimization. *Advances in Neural Information Processing Systems*. https://doi.org/10.48550/arXiv.1805.10367

Ma, X., Wang, J., & Zhang, X. (2025). Data-free black-box federated learning via zeroth-order gradient estimation. *Proceedings of the AAAI Conference on Artificial Intelligence*, *39*(18), 19314–19322. https://doi.org/10.1609/aaai.v39i18.34126

McMahan, B., Moore, E., Ramage, D., Hampson, S., & y Arcas, B. A. (2017). Communication-efficient learning of deep networks from decentralized data. *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, 1273–1282. https://doi.org/10.48550/arXiv.1602.05629

Nesterov, Y., & Spokoiny, V. (2017). Random gradient-free minimization of convex functions. *Foundations of Computational Mathematics*, *17*(2), 527–566. https://doi.org/10.1007/s10208-015-9296-2

Nguyen, D. C., Ding, M., Pathirana, P. N., Seneviratne, A., Li, J., & Poor, H. V. (2021). Federated learning for internet of things: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, *23*(3), 1622–1658. https://doi.org/10.1109/COMST.2021.3075439

Nguyen, D. C., Pham, Q.-V., Pathirana, P. N., Ding, M., Seneviratne, A., Lin, Z., et al. (2023). Federated learning for smart healthcare: A survey. *ACM Computing Surveys*, *55*(3), 1–37. https://doi.org/10.1145/3501296

Reisizadeh, A., Mokhtari, A., Hassani, H., Jadbabaie, A., & Pedarsani, R. (2020). Fedpaq: A communication-efficient federated learning method with periodic averaging and quantization. *International Conference on Artificial Intelligence and Statistics*, 2021–2031. https://doi.org/10.48550/arXiv.1909.13014

Rieke, N., Hancox, J., Li, W., Milletari, F., Roth, H. R., Albarqouni, S., et al. (2020). The future of digital health with federated learning. *NPJ Digital Medicine*, *3*(1), 1–7. https://doi.org/10.1038/s41746-020-00323-1

Rubinstein, R. Y., & Kroese, D. P. (2004). *The cross-entropy method: A unified approach to combinatorial optimization, monte-carlo simulation and machine learning*. Springer Science & Business Media. https://doi.org/10.1007/978-1-4757-4321-0

Teo, Z. L., Jin, L., Li, S., Miao, D., Zhang, X., Ng, W. Y., et al. (2024). Federated machine learning in healthcare: A systematic review on clinical applications and technical architecture. *Cell Reports Medicine*, *5*(2), 101419. https://doi.org/10.1016/j.xcrm.2024.101419

Turner, R., Eriksson, D., McCourt, M., Kiili, J., Laaksonen, E., Xu, Z., et al. (2021). Bayesian optimization is superior to random search for machine learning hyperparameter tuning: Analysis of the black-box optimization challenge 2020. *NeurIPS 2020 Competition and Demonstration Track*, 3–26. https://doi.org/10.48550/arXiv.2104.10201

Wang, S., Tuor, T., Salonidis, T., Leung, K. K., Makaya, C., He, T., et al. (2019). Adaptive federated learning in resource constrained edge computing systems. *IEEE Journal on Selected Areas in Communications*, *37*(6), 1205–1221. https://doi.org/10.1109/JSAC.2019.2904348

Wang, X., Jin, Y., Schmitt, S., & Olhofer, M. (2023). Recent advances in bayesian optimization. *ACM Computing Surveys*, *55*(13s), 287:1–287:36. https://doi.org/10.1145/3582078

Wang, Y., Du, S., Balakrishnan, S., & Singh, A. (2018). Stochastic zeroth-order optimization in high dimensions. *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, 1356–1365. https://doi.org/10.48550/arXiv.1710.10551

Wei, K., Li, J., Ding, M., Ma, C., Yang, H. H., Farokhi, F., et al. (2020). Federated learning with differential privacy: Algorithms and performance analysis. *IEEE Transactions on Information Forensics and Security*, *15*, 3454–3469. https://doi.org/10.1109/TIFS.2020.2988575

Wu, C., Wu, F., Lyu, L., Huan, Y., & Xie, X. (2022). Communication-efficient federated learning via knowledge distillation. *Nature Communications*, *13*(1), 2032. https://doi.org/10.1038/s41467-022-29763-x

Yang, Q., Liu, Y., Chen, T., & Tong, Y. (2019). Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology*, *10*(2), 1–19. https://doi.org/10.1145/3298981

Yang, Y., Yang, Z., Wang, L., Zhu, L., & Wang, M. (2025). Dynamic personalized federated learning via representation-driven clustering. *IEEE Internet of Things Journal*. https://doi.org/10.1109/JIOT.2025.3577661

Zhang, C., Xie, Y., Bai, H., Yu, B., Li, W., & Gao, Y. (2021). A survey on federated learning. *Knowledge-Based Systems*, *216*, 106775. https://doi.org/10.1016/j.knosys.2021.106775