گزارش عملکرد چهارماهه اخیر
(از زمان برگزاری آخرین جلسه شورای علمی تا اکنون)

نعیمه امیدوار
پژوهشگر پسادکتری
پژوهشکده علوم کامپیوتر

- **مقالات ژورنالی** زیر در ۴ ماه اخیر ارسال شدند که هم‌اکنون تحت داوری می‌باشند:

۱- نتایج دور اول داوری مقاله زیر دریافت شد و نسخه revision مقاله آماده و ارسال گردید که هم اکنون تحت ادامه پروسه داوری قرار دارد. کامنتهای داوران نیز در پیوست ۱ ضمیمه شده است.

1. **N. Omidvar**, S. M. Hosseini, M. Maddah-Ali, “Hybrid-Order Distributed SGD: Balancing Communication Overhead, Computational Complexity, and Convergence Rate for Distributed Learning,” Under review ***(1st revision submitted)*** in **Elsevier Journal of Neurocomputing (Q1)**, 2023. [Date of revision: 2024/02/03]

2. **N. Omidvar**, M. Ahmadi, S. M. Hosseini, “Optimal Service Placement, Request Routing and CPU Sizing in Cooperative Mobile Edge Computing Networks for Delay-Sensitive Applications,” Under review in IEEE Journal on Selected Areas in Communications **(IEEE JSAC)** , special issue on Advanced Optimization Theory and Algorithms for Next Generation Wireless Communication Networks, 2023. [Date of submission: 2023/12/08]

- **مقالات کنفرانسی** زیر در ۴ ماه اخیر ارسال شدند که هم‌اکنون تحت داوری می‌باشند:

3. M. Hosseini, A. Jamshidi, M. Nourmousavi, M. Jafari, **N. Omidvar**, “Extended Deep Submodular Function,” Under review in the 41st **International Conference in Machine Learning (ICML)**, 2024. [Date of submission: 2024/02/15]

4. S. M. Hosseini, A. Jamshidi, M. Jafari, **N. Omidvar**, “A Combinatorial Auction Design to Improve Bidding and Allocation Complexity,” Under review in the 25th **ACM Conference on Economics and Computation (EC)**, 2024. [Date of submission: 2024/02/12]

لازم به توضیح است که کنفرانس آخری یعنی ACM Conference on Economics and Computation (EC) یکی از کنفرانس‌های علمی پیشرو، در رده‌ی معادل **A*** با **5-year Acceptace Rate=24%** که به صورت double blind داوری می‌شود، از ACM SIGecom است که در زمینه‌ی ارائه پیشرفتهای تئوری، تجربی و کاربردی در ارتباط بین computation و economics می‌باشد.

در پایان، لازم به ذکر است که بنده در پروسه **جذب هیات علمی** از دانشکده‌های زیر پذیرش گرفته‌ام:

1. **دانشگاه صنعتی شریف**، دانشکده علوم کامپیوتر و ریاضی،
2. **دانشگاه صنعتی شریف**، گروه سیستم‌های اطلاعات و علوم داده، پژوهشکده جامع علوم و فناوری‌های همگرا،
3. **دانشگاه صنعتی امیرکبیر**، دانشکده مهندسی کامپیوتر،
4. **دانشگاه صنعتی امیرکبیر**، دانشکده مهندسی برق.

از آنجا که مراحل نهایی استخدام احتمالا یک یا دو سال طول می‌کشد، درخواست پسادکتری ارشد در پژوهشکده علوم کامپیوتر IPM را دارم.

پیوست ۱
کامنت‌های داوران برای مقاله اول تحت داوری

From: **Neurocomputing** em@editorialmanager.com
Subject: NEUCOM-D-23-04179
Date: 26 December 2023 at 11:24 PM
To: Naeimeh Omidvar omidvar@ipm.ir

RRef.: Ms. No. NEUCOM-D-23-04179
Hybrid-Order Distributed SGD: Balancing Communication Overhead, Computational Complexity, and Convergence Rate for Distributed Learning
Neurocomputing

Dear Dr. Omidvar,

Please find below the referee reports. Based on these and the corresponding recommendation of the associate editor, I have to inform you that your paper

Hybrid-Order Distributed SGD: Balancing Communication Overhead, Computational Complexity, and Convergence Rate for Distributed Learning with manuscript number: NEUCOM-D-23-04179

in its present form cannot be accepted for publication in Neurocomputing.

However, I would very much like to invite you to revise your paper, seriously taking into account the comments of the reviewers, and to resubmit your revised version by Feb 06, 2024 (mm/dd/yy). Any revision received after that may be treated as a new submission.

To submit your revision, go to http://ees.elsevier.com/neucom/ and login as an Author. You will see a menu item call Submission Needing Revision. Here you will also find your submission record.

The revised material should consist of

- your response to the reviewers' comments (to be uploaded as "Revision notes"),
- the revised PDF of the manuscript,
- the source files that have been used to prepare it (source files in LaTeX or Word, as well as separate figure files; these will be used for the eventual typesetting of the paper)
- and finally, biographies and pictures of all authors.

*** Please note: while submitting the revised manuscript, please double check the author names provided in the submission and make sure to indicate any authorship related changes in the revision. Once a paper is accepted, we do not accept any changes to the author list unless explicit approval is given from co-authors and respective editor handling the submission; this may cause a significant delay in publishing your manuscript. Therefore, please make sure that you include the correct author list in the revised text of your manuscript. ***

Other journal-related information is included below, following the reviewer's comments.

Research Elements (optional)
This journal encourages you to share research objects - including your raw data, methods, protocols, software, hardware and more – which support your original research article in a Research Elements journal. Research Elements are open access, multidisciplinary, peer-reviewed journals which make the objects associated with your research more discoverable, trustworthy and promote replicability and reproducibility. As open access journals, there may be an Article Publishing Charge if your paper is accepted for publication. Find out more about the Research Elements journals at https://www.elsevier.com/authors/tools-and-resources/research-elements-journals?dgcid=ec_em_research_elements_email.

Kind regards,
Zidong Wang, PhD
Editor in Chief
Neurocomputing

Editor's and reviewers' comments:

Reviewer #2: This paper aims to investigate a method that handle with the trade off among communication burden, computation load, and concergence rete for distributed learning. There are several suggestions:
1. Comparions with state-of-art methods in Table 1 is puzzled. Please claim all the variables in Table 1. For example, what does m and N_0 mean?
2. Please explane how to obtain the inquality (c) in (A.14) at Pate 28.
3.Why the training loss becomes greater after 500 iterations for CIFAR100 in Figure 2. I do not think the convergence rate is quite closed to the first-order algorithm.
4. I cannot figure out why the accuracy of the proposed method becomes much better after around 600 iterations in Figure 3. I

think the authors should make some justifications
5. It seems that Figure 5 is nearly the same as Figure 4, exchanging the axis x and the axis y. Is that possible for the authors to give a table to list the accuracy, the communication load, and the computation load of the proposed method and other methods?

Reviewer #3: This paper proposes a new hybrid-order distributed stochastic gradient descent (HO-SGD) algorithm which strikes a better balance between communication overhead, computation load and convergence speed for a general class of non-convex stochastic optimization problems. The idea and derived results in this paper are interesting. However, some minor comments are given below.
1. This paper takes the two-point stochastic gradient approximation method to perform ZO gradient update, then whether other existing ZO gradient approximation methods, such as the single-point gradient approximation method is applicable to the proposed algorithm, and whether similar theoretical results can be obtained?
2. In traditional distributed algorithms, workers only communicate with their neibouring nodes, which results in less communication burden compared to centralized algorithms. However, in the last step of the proposed algorithm, at each iteration, the gradient information of all workers is aggregated, then will this lead to an increase in the communication burden?
3. The author utilizes important ZO/FO scheduler in the proposed algorithm, then how does ZO/FO scheduler works during the implementation of the algorithm? Maybe this should be explained before proposing the algorithm.
4. Whether the condition on boundness of the norm of the gradient in Assumption 3 is common when there exits no set constraint in problem (1)?
5. There are some grammatical errors that should be revised, please check them carefully.

Associate Editor: Two reviews were collected. Reviewer 2 has some concerns about the simulation when comparing with state of art, and Reviewer 3 has some concerns about the algorithm and its implemenation. After reading the paper, I think the proposed method is novel but unclear, and I suggest the authors to give some detailed discusions or remarks on the design and analysis of the algorithm.

==

Additional journal-related information:

Please proceed to the following link to update your personal classifications and keywords, if necessary:
Update Personal Keywords

پیوست ۲
پاسخ‌ها به کامنت‌های داوران برای مقاله اول تحت داوری

Response to the Reviewers (Hybrid-Order Distributed SGD: Balancing Communication Overhead, Computational Complexity, and Convergence Rate for Distributed Learning)

Naeimeh Omidvar, Seyed Mohammad Hosseini, Mohammad Ali Maddah Ali

We greatly appreciate the time and effort the associate editor, the editor, and the reviewers put into our paper, and we would like to thank them for their input and suggestions that greatly helped us to improve the presentation and technical content of our manuscript. We believe that we have addressed all the points raised in the review comments and we have revised the manuscript based on them.

In particular, the summary of the main changes can be identified as follows:

- A new lemma (Lemma 3 in the revised version) is presented and proved, which explains how the inequality (c) in (A.16) is obtained on Page 31 (of the revised paper).
- A new baseline was added and compared to (called NO-OP), to better highlight the contributions of the intermediate ZO updates to the overall performance of the proposed method.
- The experimental results and the associated figures are updated (with better tuning the ZO/FO scheduler).
- The applicability and performance of another zeroth-order (ZO) gradient estimation scheme, named single-point ZO gradient estimation [38], in the proposed algorithm is also investigated here and can be found in the following.
- A subsection on “Discussions on the ZO/FO Scheduler” (Subsection 3.3) is presented before the main algorithm, to clearly highlight the role of the ZO/FO scheduler in the proposed algorithm and its advantages in balancing the communication-computation-convergence trade-off. It also introduces and elaborates on various types of ZO/FO schedulers.
- The paper is thoroughly checked grammatically, and carefully revised wherever needed.

Finally, we would like to note that all the references in the following are numbered with respect to the new submission. In addition, the major revised or new parts in the paper have been marked in blue to be found easily.

Response to Reviewer 2

We thank the reviewer for the detailed and insightful comments on our paper.

- **Reviewer Summary:** *This paper aims to investigate a method that handle with the trade off among communication burden, computation load, and convergence rate for distributed learning.*

  There are several suggestions:

1. **Reviewer Comment:** *Comparisons with state-of-art methods in Table 1 is puzzled. Please claim all the variables in Table 1. For example, what does m and N_0 mean?*

   Response: We thank the reviewer for their comment. Based on the reviewer’s suggestion, in the revised paper, we have included all the variables in Table 1, as shown on the next page.

Table 1: Comparison of the Proposed Method to Various State-of-the-art Methods in the Literature (N : total number of iterations, m : Number of worker nodes, d : model dimension).

Method	Convergence Rate	Communication Load per Iteration	Normalized Computational Load	Comments
HO-SGD (Proposed)	$\mathcal{O}(\frac{d}{\sqrt{mN}})$	$\frac{N-N_0}{N}d+\frac{N_0}{N}$	$\frac{1}{d}\frac{N_0}{N}+\frac{N-N_0}{N}$	N_0 : Number of ZO iterations
syncSGD [13]	$\mathcal{O}(\frac{1}{\sqrt{mN}})$	d	1	Highest communication overhead
RI-SGD [25]	$\mathcal{O}(\frac{\tau}{\sqrt{mN}})$	d/τ	$\mu m+1$	High storage requirement at each worker (τ: the period of model averaging, μ: redundancy factor)
ZO-SGD [23]	$\mathcal{O}(\frac{(d/m)^{1/3}}{(N)^{1/4}})$	1	$\simeq\frac{1}{d}$	
ZO-SVRG-Ave [26]	$\mathcal{O}(\frac{d}{N}+\frac{1}{\min\{d,m\}})$	1	$\mathcal{O}(\frac{K}{d})$	Full dataset storage required at each worker (K: size of dataset)
QSGD [5]	$\mathcal{O}(\frac{1}{N}+\sqrt{d})$	$\mathcal{O}(s^2+s\sqrt{d})$	>1	s: number of quantization levels

2. **Reviewer Comment:** *Please explain how to obtain the inequality (c) in (A.14) at Page 28.*

Response: We thank the reviewer for this comment. To remove the ambiguity, we have included and proved a new lemma as follows, Lemma 3 in the revised paper, which is then used to clearly obtain the aforementioned inequality.

Lemma 3. Let S_p and $s(d)$ denote the unit sphere in $\mathbb{R}^d$ and its surface area, respectively, and $\boldsymbol{G}_\mu(\boldsymbol{x},\boldsymbol{\zeta},\boldsymbol{v})$ be as defined in (A.6). Then, we have

$$\mathbb{E}_{\boldsymbol{v}}\left[\|\,\boldsymbol{G}_\mu\left(\boldsymbol{x},\boldsymbol{\zeta},\boldsymbol{v}\right)\|^2\right]\leq 2d\,\|\,\nabla F\left(\boldsymbol{x},\boldsymbol{\zeta}\right)\|^2+\frac{\mu^2L^2d^2}{2}. \qquad (1)$$

Proof.

$$\begin{aligned}
&\mathbb{E}_{\boldsymbol{v}}\left[\|\,\boldsymbol{G}_\mu\left(\boldsymbol{x},\boldsymbol{\zeta},\boldsymbol{v}\right)\|^2\right],\\
&\quad=\frac{1}{s(d)}\int_{S_p}\frac{d^2}{\mu^2}\left|F\left(\boldsymbol{x}+\mu\boldsymbol{v},\boldsymbol{\zeta}\right)-F\left(\boldsymbol{x},\boldsymbol{\zeta}\right)\right|^2\|\,\boldsymbol{v}\,\|^2\,d\boldsymbol{v},\\
&\quad=\frac{d^2}{s(d)\mu^2}\int_{S_p}\Big[F\left(\boldsymbol{x}+\mu\boldsymbol{v},\boldsymbol{\zeta}\right)-F\left(\boldsymbol{x},\boldsymbol{\zeta}\right)-\left\langle\nabla F\left(\boldsymbol{x},\boldsymbol{\zeta}\right),\mu\boldsymbol{v}\right\rangle+\left\langle\nabla F\left(\boldsymbol{x},\boldsymbol{\zeta}\right),\mu\boldsymbol{v}\right\rangle\Big]^2d\boldsymbol{v},\\
&\quad\overset{\text{(a)}}{\leq}\frac{d^2}{s(d)\mu^2}\int_{S_p}\Big[2\left(F\left(\boldsymbol{x}+\mu\boldsymbol{v},\boldsymbol{\zeta}\right)-F\left(\boldsymbol{x},\boldsymbol{\zeta}\right)-\left\langle\nabla F\left(\boldsymbol{x},\boldsymbol{\zeta}\right),\mu\boldsymbol{v}\right\rangle\right)^2\\
&\qquad\qquad+2\left(\left\langle\nabla F\left(\boldsymbol{x},\boldsymbol{\zeta}\right),\mu\boldsymbol{v}\right\rangle\right)^2\Big]d\boldsymbol{v},\\
&\quad\overset{\text{(b)}}{\leq}\frac{d^2}{s(d)\mu^2}\left[\int_{S_p}2\left(\frac{L\mu^2}{2}\,\|\,\boldsymbol{v}\,\|^2\right)^2d\boldsymbol{v}+2\mu^2\int_{S_p}\nabla F\left(\boldsymbol{x},\boldsymbol{\zeta}\right)^T\boldsymbol{v}\boldsymbol{v}^T\nabla F\left(\boldsymbol{x},\boldsymbol{\zeta}\right)d\boldsymbol{v}\right],\\
&\quad\overset{\text{(c)}}{\leq}\frac{d^2}{s(d)\mu^2}\left[\frac{L^2\mu^4}{2}s(d)+2\mu^2\frac{s(d)}{d}\,\|\,\nabla F\left(\boldsymbol{x},\boldsymbol{\zeta}\right)\|^2\right],\\
&\quad=2d\,\|\,\nabla F\left(\boldsymbol{x},\boldsymbol{\zeta}\right)\|^2+\frac{\mu^2L^2d^2}{2}, \qquad (2)
\end{aligned}$$

where inequality (a) is due to the fact that $\forall a,b\in\mathbb{R}:\ (a+b)^2\leq 2a^2+2b^2$, and inequality

(b) is due to Assumption 2, which results in [32]: $\forall \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^d, \forall \boldsymbol{\zeta}$,

$$F(\boldsymbol{x}, \boldsymbol{\zeta}) \leq F(\boldsymbol{y}, \boldsymbol{\zeta}) + \langle \nabla F(\boldsymbol{y}, \boldsymbol{\zeta}), \boldsymbol{x} - \boldsymbol{y} \rangle + \frac{L}{2} \parallel \boldsymbol{x}_1 - \boldsymbol{x}_2 \parallel^2 .$$

Finally, inequality (c) is due to the fact that $\int_{S_p} \boldsymbol{v}\boldsymbol{v}^T d\boldsymbol{v} = \frac{s(d)}{d} I_d$, where I_d is the identity matrix in $\mathbb{R}^d$.

□

3. **Reviewer Comment:** *Why the training loss becomes greater after 500 iterations for CIFAR100 in Figure 2. I do not think the convergence rate is quite closed to the first-order algorithm.*

Response: We thank the reviewer for their precise reading of our paper. We would like to note that although we theoretically guarantee the convergence of the proposed algorithm to the stationary point, it is not guaranteed to *monotonically* converge to the stationary point. Therefore, the proposed method is not necessarily a monotone method, and hence, as the algorithm proceeds, the objective function (i.e., the training loss) can temporarily become greater in some iterations. This non-monotone behavior is common in many optimization methods for non-convex (and even in many convex) optimization problems, e.g, in the SGD method [13], [1], and is more obvious in our method due to the zeroth-order iterations which approximate the stochastic first-order gradients with some biased errors. Moreover, we would like to note that in the context of machine learning, the most important performance criterion is *the test accuracy*, not *the training loss*, and these two performance metrics do not necessarily change in accordance with each other. In fact, it is well known that attaining near-zero training loss is a sign of *over-fitting* which leads to low testing accuracy. Hence, an optimization algorithm that achieves the best optimality performance (i.e., the least training loss) does not necessarily lead to a good learning generalization (i.e., the testing accuracy).

As explained in the paper, in the proposed hybrid-order method, the zeroth-order iterations provide a new exploration mechanism that efficiently prevents over-fitting (i.e., the premature vanishing of the loss function without achieving high test accuracies) by exploring better. Consequently, although the zeroth-order iterations may cause some temporary increases in the objective function (and hence, the proposed method may require more iterations for the loss function to converge), the proposed method can efficiently prevent over-fitting and provide higher test accuracies.

In particular, in Figure 2 of the previous version for CIFAR100, after around 500 iterations, the proposed algorithm reaches some neighborhood of a stationary point, and therefore, in order to better *explore* the learning environment (i.e., the search space) to enhance the accuracy and also improve the communication and computation overheads, the proposed algorithm begins to perform more zeroth-order update iterations. This exploration comes with the cost of a temporary increase in the loss function. However, gradually, the iterate (the model) gets closer again to the stationary point and the loss function becomes smaller and smaller until convergence, while a significantly improved accuracy has been reached, as shown in Fig. 3.

To better clarify these points in the paper, we have added a summary of the above justifications to the discussions of the experimental results in Section 5.2 of the revised paper, as the following paragraphs:

> Moreover, although the convergence of the proposed algorithm to the stationary point is theoretically guaranteed, it is not guaranteed to converge in a monotonic manner. Therefore, the proposed method is not necessarily a monotone method, and hence, as the algorithm proceeds, the objective function (i.e., the training loss) can temporarily become greater in some iterations. This non-monotone behavior is common in many optimization methods for non-convex (and even in many convex) optimization problems, e.g, in the SGD method [13], [1], and is more obvious in our method due to the zeroth-order updates which approximate the stochastic first-order gradients with some errors.

and

> Considering Fig.s 2 and 3, it is noted that although the zeroth-order iterations may cause some temporary increases in the objective function (and consequently, the proposed method may require more iterations for the loss function to converge), the proposed method efficiently prevents over-fitting (i.e., premature vanishing of the loss function without achieving high test accuracies) and hence provides superior test accuracies. In particular, as can be seen for CIFAR10, after around 700 iterations, the proposed algorithm has reached some neighborhood of a stationary point, and therefore, in order to better explore the underlying domain (i.e., the search space) to enhance the accuracy and also improve the communication and computation overheads, the proposed algorithm begins to apply more zeroth-order update iterations. This exploration comes with the cost of a temporary increase in the loss function. However, gradually, the iterate (the model) gets closer again to the stationary point, and the loss function becomes smaller and smaller until convergence, while a significantly improved accuracy is reached, as shown in Fig. 3.

Finally, in the updated figures in the revised paper, we have better tuned the parameters of the adaptive ZO/FO scheduler, and hence, the training loss and the test accuracies converge more smoothly, and the previous sudden jumps are significantly reduced.

4. **Reviewer Comment:** *I cannot figure out why the accuracy of the proposed method becomes much better after around 600 iterations in Figure 3. I think the authors should make some justifications.*

   Response: Please kindly refer to the reply to the previous comment.

5. **Reviewer Comment:** *It seems that Figure 5 is nearly the same as Figure 4, exchanging the axis x and the axis y. Is that possible for the authors to give a table to list the accuracy, the communication load, and the computation load of the proposed method and other methods?*

   Response:

   We would like to note that these two figures together aim to demonstrate a trade-off between the communication load and computation load at various accuracies. As a result, we believe that summarizing these results into a table reporting the performance metrics (the communication and computation loads) at just one specific achieved accuracy is not very informative and even might be miss-leading. Therefore, we believe that having these figures would more clearly compare the communication and computation loads of the proposed method and the baselines at different reached test accuracies. As an instance, it can be verified from these figures that with the same level of communication or computation overhead, the proposed method can provide significantly higher test accuracies, demonstrating superior generalization compared to all the other baselines. Finally, please note that under the updated experimental results in the revised paper, these two figures are not similar to each other anymore.

Response to Reviewer 3

We thank the reviewer for the detailed and insightful comments on our paper.

- **Reviewer Summary:** *This paper proposes a new hybrid-order distributed stochastic gradient descent (HO-SGD) algorithm which strikes a better balance between communication overhead, computation load and convergence speed for a general class of non-convex stochastic optimization problems. The idea and derived results in this paper are interesting.*

  However, some minor comments are given below.

1. **Reviewer Comment:** *This paper takes the two-point stochastic gradient approximation method to perform ZO gradient update, then whether other existing ZO gradient approximation methods, such as the single-point gradient approximation method is applicable to the proposed algorithm, and whether similar theoretical results can be obtained?*

   Response:

   We thank the reviewer for this comment. As mentioned by the reviewer, in the ZO iterations of our proposed method, we have used the two-point ZO gradient estimation as follows:

   $$\boldsymbol{G}_{\mu}^{\text{two-pint}}\big(\boldsymbol{x},\boldsymbol{\zeta},\boldsymbol{v}\big) \triangleq \frac{d}{\mu}\Big[F\left(\boldsymbol{x}+\mu\boldsymbol{v},\boldsymbol{\zeta}\right)-F\left(\boldsymbol{x},\boldsymbol{\zeta}\right)\Big]\boldsymbol{v}, \tag{3}$$

   where $\boldsymbol{v}$ is a random vector uniformly drawn from the unit sphere S_p. However, any other ZO gradient estimation is also applicable to the proposed hybrid-order method. In particular, the single-point ZO gradient estimation [38]:

   $$\boldsymbol{G}_{\mu}^{\text{single-pint}}\big(\boldsymbol{x},\boldsymbol{\zeta},\boldsymbol{v}\big) \triangleq \frac{d}{\mu}\Big[F\left(\boldsymbol{x}+\mu\boldsymbol{v},\boldsymbol{\zeta}\right)\Big]\boldsymbol{v} \tag{4}$$

   can be used in Algorithm 1 to perform ZO gradient updates. In the following, we have compared the performance results of the proposed algorithm under two-point and single-point ZO gradient estimations, for training ResNet50 using CIFAR10 on four parallel NVIDIA RTX A4000 GPUs.

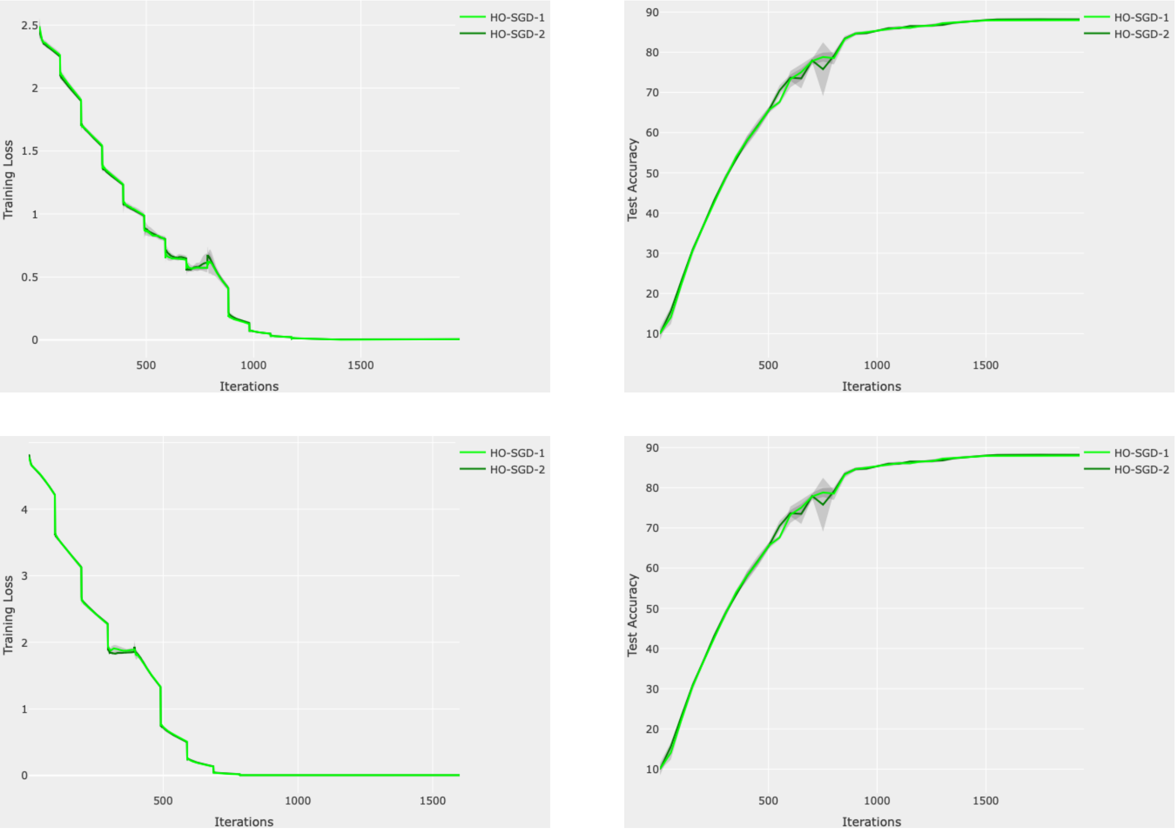

Figure 1: Comparison of the performance of the proposed algorithm under single-point stochastic gradient estimation (named HO-SGD-1) and two-point ZO stochastic gradient estimation (named HO-SGD-2) for training CIFAR10 (top row) and CIFAR100 (bottom row) on ResNet-50.

As can be seen from these figures, the performance of the proposed algorithm under the aforementioned ZO gradient estimations is almost the same. This is mainly because the two-point and single-point ZO gradient estimations are equivalent in expectation:

$$
\begin{aligned}
\mathbb{E}_{\boldsymbol{v}}\left[\boldsymbol{G}_{\mu}^{\text{two-pint}}(\boldsymbol{x},\boldsymbol{\zeta},\boldsymbol{v})\right] &= \frac{d}{\mu}\int_{S_p}\Big(F\left(\boldsymbol{x}+\mu\boldsymbol{v},\boldsymbol{\zeta}\right)-F\left(\boldsymbol{x},\boldsymbol{\zeta}\right)\Big)\boldsymbol{v}d\boldsymbol{v},\\
&= \frac{d}{\mu}\int_{S_p} F\left(\boldsymbol{x}+\mu\boldsymbol{v},\boldsymbol{\zeta}\right)\boldsymbol{v}d\boldsymbol{v} - \frac{d}{\mu}F\left(\boldsymbol{x},\boldsymbol{\zeta}\right)\Big)\int_{S_p}\boldsymbol{v}d\boldsymbol{v},\\
&= \frac{d}{\mu}\int_{S_p} F\left(\boldsymbol{x}+\mu\boldsymbol{v},\boldsymbol{\zeta}\right)\boldsymbol{v}d\boldsymbol{v},\\
&= \mathbb{E}_{\boldsymbol{v}}\left[\boldsymbol{G}_{\mu}^{\text{single-pint}}(\boldsymbol{x},\boldsymbol{\zeta},\boldsymbol{v})\right], \qquad (5)
\end{aligned}
$$

where the second last inequality is due to the fact that $\int_{S_p}\boldsymbol{v}d\boldsymbol{v}=0$, and the last inequality is due to the definition of the single-point ZO gradient estimation as stated in (4). As such, we believe that following similar proof lines as in our paper, similar theoretical results can be obtained for the proposed method with single-point ZO gradient estimation.

We would like to note that since the performance of the single-point ZO gradient estimation was found to be very similar to that of the two-point ZO gradient estimation (already used in the paper), we did not include the HO-SGD-1 scheme in the revised paper.

2. **Reviewer Comment:** *In traditional distributed algorithms, workers only communicate with their neighboring nodes, which results in less communication burden compared to centralized algorithms. However, in the last step of the proposed algorithm, at each iteration, the gradient information of all workers is aggregated, then will this lead to an increase in the communication burden?*

   Response:

   We would like to first note that in the considered problem in the paper, the worker nodes aim to solve *one (global)* optimization problem in a distributed manner. Therefore, the target models that they try to learn are the same. As such, any algorithm to solve such a distributed learning problem with a global model will need two main phases (which can be performed iteratively): the local training phase and the *global* aggregation phase. Considering the aggregation phase, although aggregation of the updates among only the local neighboring nodes can reduce the overall required communication, it will drastically lead to high model discrepancies between local model updates of different (non-neighboring) nodes and hence, significantly slow down the convergence of the local models to one global model [Ref. 1][1]. This is due to the fact that the residual errors in the locally trained models accumulate and propagate through the iterations and lead to a slow convergence to the stationary point of the problem. That is why for a fast-converging distributed learning/optimization, the aggregation phase is preferred to be done among *all* the worker nodes to synchronize the local models and mitigate their discrepancy, e.g., see the related works illustrated in the paper such as [7-14], [25], [28], and [29], to name just a few.

   Furthermore, under the proposed method specifically, at the iterations with ZO update, the synchronization of the local models is done through communicating just a scalar. This contributes to a huge saving in the communication load of the proposed method in practice while having a guaranteed fast convergence and a low computational complexity as well. To elaborate more, consider the most communication-efficient distributed learning methods, i.e., model-averaging methods as described in the paper, which communicate a vector of d scalars (d: the model dimension) at the end of every τ iterations. As such, their communication load per iteration is $\frac{d}{\tau}$. On the other hand, under a periodic ZO/FO scheduler for example, our proposed method performs one FO iteration after every $(\tau - 1)$ ZO iterations, which contributes to communicating $\frac{d+(\tau-1)}{\tau}$ scalers per iteration. Therefore, the communication overhead of our method is only $\frac{\tau-1}{\tau} < 1$ more than the communication overhead of the most communication-efficient distributed learning/optimization methods. That is why although under the proposed method, the intermediate models are always synchronized among all the worker nodes (which contributes to its fast convergence speed), it does not lead to a high communication burden.

   In summary, since in our paper we aim to strike a better balance between communication overhead, computational complexity, and convergence speed, we have considered aggregation with all the worker nodes, rather than the neighboring nodes only.

   Finally, applying these kinds of message passing algorithms (i.e., local communication for aggregation, as mentioned by the reviewer) to the distributed learning/optimization methods can significantly complicate the theoretical convergence analysis and convergence rate guarantees. We believe that investigating this direction can be an interesting direction for future works.

3. **Reviewer Comment:** *The author utilizes important ZO/FO scheduler in the proposed algorithm, then how does ZO/FO scheduler works during the implementation of the algorithm? Maybe this should be explained before proposing the algorithm.*

   Response:

   As suggested by the reviewer, to make the paper's presentation more clear, before presenting Algorithm 1, we have better highlighted discussions on the ZO/FO scheduler, in *Subsection 3.3: Discussions on the ZO/FO Scheduler*, which clearly defines the role of the ZO/FO scheduler in the proposed algorithm and its advantages in balancing the communication-computation-convergence trade-off by properly balancing the exploration and exploitation steps. We have also introduced and elaborated several types of ZO/FO schedulers in this section.

[1] [Ref. 1] Viviani, Paolo. "Deep Learning at Scale with Nearest Neighbors Communications." (2019).

In addition and to better clarify, we have also included the following explanations at the beginning of Section 5.2 on Experimental Results.

> First note that, as illustrated before, an important component of the proposed scheme that can be tuned is the ZO/FO scheduler. In fact, the scheduler introduces a trade-off in training, in which as it schedules more iterations with FO updates, the convergence speed increases, but the communication overhead and the computational load level up too. One exploration is to see the effects of different schedulers on the performance of the proposed method. As such, we first examined various types for the ZO/FO scheduler, including periodic, monotonic, and different adaptive (dynamic) schedulers. We also used a grid search to tune the parameters of each type of scheduler. The details of such experiments can be found in Appendix B. Based on the results of this experiment, a dynamic scheduler had the best performance among the other aforementioned types of schedulers. As such, in the rest of the experiments, we set the ZO/FO scheduler as this one with a tuned set of parameters for each dataset. For more details on the adopted scheduler, please see Appendix B.

Besides, we would like to note that the theoretical convergence analyses of our proposed algorithm are presented for any general ZO/FO scheduler. Though in practice, the choice of the scheduler can affect the empirical performance of the algorithm, and in Appendix B of the paper, we have explored the effects of different types of ZO/FO schedulers on the performance of the proposed method. Finally, would like to note that theoretical analyses on optimizing the ZO/FO schedulers and deriving costume-made convergence results for special cases of the scheduler (such as periodic, monotonic, and adaptive rules) are out of the scope of this paper, and are regarded as interesting directions for extending this work.

4. **Reviewer Comment:** *Whether the condition on boundness of the norm of the gradient in Assumption 3 is common when there exits no set constraint in problem (1)?*

   Response: We thank the reviewer for this thoughtful comment. Following the reviewer's comment, we carefully and thoroughly reviewed our proofs and figured out that the aforementioned assumption on the boundedness of the norm of the true gradient is not required and has not been used in our proofs. Therefore, we deleted the aforementioned assumption in the revised paper.

5. **Reviewer Comment:** *There are some grammatical errors that should be revised, please check them carefully.*

   Response: We thank the reviewer for their careful reading of our paper. Following this comment, we thoroughly and carefully checked the paper grammatically, and revised it wherever needed.