Det-CGD: Compressed Gradient Descent with Matrix Stepsizes for Non-Convex Optimization

Hanmin Li HANMIN.LI@KAUST.EDU.SA
Avetik Karagulyan AVETIK.KARAGULYAN@KAUST.EDU.SA
Peter Richtárik PETER.RICHTARIK@KAUST.EDU.SA
King Abdullah University of Science and Technology, Saudi Arabia

Abstract

This paper introduces a new method for minimizing matrix-smooth non-convex objectives through the use of novel Compressed Gradient Descent (CGD) algorithms enhanced with a matrix-valued stepsize. The proposed algorithms are theoretically analyzed first in the single-node and subsequently in the distributed settings. Our theoretical results reveal that the matrix stepsize in CGD can capture the objective's structure and lead to faster convergence compared to a scalar stepsize. As a byproduct of our general results, we emphasize the importance of selecting the compression mechanism and the matrix stepsize in a layer-wise manner, taking advantage of model structure. Moreover, we provide theoretical guarantees for free compression, by designing specific layer-wise compressors for the non-convex matrix smooth objectives. Our findings are supported with empirical evidence.

1. Introduction

The minimization of smooth and non-convex functions is a fundamental problem in various domains of applied mathematics. Most machine learning algorithms rely on solving optimization problems for training and inference, often with structural constraints or non-convex objectives. However, non-convex problems are typically NP-hard to solve, leading to the popular approach of relaxing them to convex problems and using traditional methods. Direct approaches to non-convex optimization have shown success but their convergence and properties are not well understood, making them challenging for large scale optimization.

In this paper, we consider the general minimization problem:

$$\min_{x\in\mathbb{R}^d} f(x), \tag{1}$$

where $f:\mathbb{R}^d\to\mathbb{R}$ is a differentiable function. In order for this problem to have a finite solution we will assume throughout the paper that f is bounded from below.

Assumption 1 *There exists $f^{\inf}\in\mathbb{R}$ such that $f(x)\geq f^{\inf}$ for all $x\in\mathbb{R}^d$.*

Stochastic gradient descent (SGD) is a widely used algorithm for solving optimization problems [5, 15, 37]. In its most general form, it can be written as,

$$x^{k+1}=x^k-\gamma g(x^k), \tag{2}$$

© H. Li, A. Karagulyan & P. Richtárik.

where $g(x^k)$ is a stochastic gradient estimator of $\nabla f(x^k)$ and $\gamma > 0$ is a positive scalar stepsize. A specific variant of interest is compressed gradient descent (CGD) [27], where the gradient estimator $g(x^k)$ is obtained by applying a compressor $\mathcal{C}$ to the initial gradient $\nabla f(x^k)$. Compressors, such as sketches, aim to reduce communication overhead in distributed or federated settings.

The importance of compression in distributed optimization was highlighted by [28], as communication complexity becomes a bottleneck. Compression objectives include compressing the model broadcasted from server to clients and reducing the computational burden of local training. Among these objectives, compressing gradients has the greatest practical impact due to slower upload speeds and the benefits of averaging [25].

A notable class of compressors is sketches, which are linear operators represented by random matrices [41]. An example is the Rand-k compressor, which randomly selects k entries from the input vector and scales them to ensure an unbiased estimator. By communicating only a subset of size k instead of all d coordinates of the gradient, the number of communicated bits is reduced by a factor of d/k.

Besides the assumption that function f is bounded from below, we also assume that it is $\boldsymbol{L}$ matrix smooth, as we are trying to take advantage of the entire information contained in the smoothness matrix $\boldsymbol{L}$ and the stepsize matrix $\boldsymbol{D}$.

Assumption 2 (Matrix smoothness) *There exists $\boldsymbol{L} \in \mathbb{S}^d_+$ s.t. $f(x) \leq f(y) + \langle \nabla f(y), x - y \rangle + \frac{1}{2} \langle \boldsymbol{L}(x - y), x - y \rangle$, holds for all $x, y \in \mathbb{R}^d$.*

The assumption of matrix smoothness, which is a generalization of scalar smoothness, has been shown to be a more powerful tool for improving supervised model training. In [40], the authors proposed using smoothness matrices and suggested a novel communication sparsification strategy to reduce communication complexity in distributed optimization for convex objectives. The technique was adapted to three distributed optimization algorithms in the convex setting, resulting in significant communication complexity savings and consistently outperforming the baselines. The results of this study demonstrate the efficacy of the matrix smoothness assumption in improving distributed optimization algorithms.

The case of block-diagonal smoothness matrices is particularly relevant in various applications, such as neural networks (NN). In this setting, each block corresponds to a layer of the network, and we characterize the smoothness with respect to nodes in the i-th layer by a corresponding matrix $\boldsymbol{L}_i$. Unlike in the scalar setting, we favor the similarity of certain entries of the argument over the others. This is because the information carried by the layers becomes more complex, while the nodes in the same layers are similar. This phenomenon has been observed visually in various studies, such as those by [46] and [50].

Another motivation for using a layer-dependent stepsize has its roots in physics. In nature, the propagation speed of light in media of different densities varies due to frequency variations. Similarly, different layers in neural networks carry different information, metric systems, and scaling. Thus, the stepsizes need to be picked accordingly to achieve optimal convergence.

We study two matrix stepsized CGD-type algorithms and analyze their convergence properties for non-convex matrix-smooth functions. As mentioned earlier, we put special emphasis on the block-diagonal case. We design our sketches and stepsizes in a way that leverages this structure, and we show that in certain cases, we can achieve compression without losing in the overall communication complexity.

1.1. Contributions

Our paper contributes in the following ways:

- We propose two novel matrix stepsize sketch CGD algorithms in Section 2, which, to the best of our knowledge, are the first attempts to analyze a fixed matrix stepsize for non-convex optimization. We present a unified theorem in Section 3 that guarantees stationarity for minimizing matrix-smooth non-convex functions. The results shows that taking our algorithms improve on their scalar alternatives. The complexities are summarized in Table 1 for some particular cases.
- Assuming the that the server-to-client communication is less expensive [25, 28], we propose distributed versions of our algorithms in Appendix E, following the standard FL scheme, and prove weighted stationarity guarantees. Our theorem recovers the result for DCGD in the scalar case and improves it in general.
- We design our algorithms' sketches and stepsize to take advantage of the layer-wise structure of neural networks, assuming that the smoothness matrix is block-diagonal. In Appendix D, we prove that our algorithms achieve better convergence than classical methods.
- We validate our theoretical results with experiments. The plots and framework are provided in the Appendix.

2. The algorithms

We refer the reader to Appendix A for the mathematical notation and basic definitions. Below we define our two main algorithms:

$$x^{k+1} = x^k - \boldsymbol{D}\boldsymbol{S}^k \nabla f(x^k), \tag{det-CGD1}$$

and

$$x^{k+1} = x^k - \boldsymbol{T}^k \boldsymbol{D} \nabla f(x^k). \tag{det-CGD2}$$

Here, $\boldsymbol{D} \in \mathbb{S}^d_{++}$ is the fixed stepsize matrix. The sequences of random matrices $\boldsymbol{S}^k$ and $\boldsymbol{T}^k$ satisfy the next assumption.

Assumption 3 *We will assume that the random sketches that appear in our algorithms are i.i.d., unbiased, symmetric and positive semi-definite for each algorithm. That is*

$$\begin{aligned} &\boldsymbol{S}^k, \boldsymbol{T}^k \in \mathbb{S}^d_{+}, \quad \boldsymbol{S}^k \overset{iid}{\sim} \mathcal{S} \quad \textit{and} \quad \boldsymbol{T}^k \overset{iid}{\sim} \mathcal{T} \\ &\mathbb{E}\left[\boldsymbol{S}^k\right] = \mathbb{E}\left[\boldsymbol{T}^k\right] = \boldsymbol{I}_d, \quad \textit{for every} \quad k \in \mathbb{N}. \end{aligned}$$

A simple instance of det-CGD1 and det-CGD2 is the vanilla GD. Indeed, if $\boldsymbol{S}^k = \boldsymbol{T}^k = \boldsymbol{I}_d$ and $\boldsymbol{D} = \gamma \boldsymbol{I}_d$, then $x^{k+1} = x^k - \gamma \nabla f(x^k)$. In general, one may view these algorithms as Newton-type methods. In particular, our setting includes the Newton Star (NS) algorithm by [22]: $x^{k+1} = x^k - \left(\nabla^2 f(x^{\inf})\right)^{-1} \nabla f(x^k)$. Despite being impractical, this method converges locally quadratically and, thus, hints that constant matrix stepsize can yield fast convergence guarantees.

The difference between det-CGD1 and det-CGD2 is the update rule. When the sketch $\boldsymbol{S}$ and the stepsize $\boldsymbol{D}$ are commutative w.r.t. matrix product, the algorithms become equivalent. In general, a simple calculation shows that if we take $\boldsymbol{T}^k = \boldsymbol{D}\boldsymbol{S}^k\boldsymbol{D}^{-1}$, then det-CGD1 and det-CGD2 are the same. Defining $\boldsymbol{T}^k$ according to above relation, we recover the unbiasedness $\mathbb{E}\left[\boldsymbol{T}^k\right] = \boldsymbol{D}\mathbb{E}\left[\boldsymbol{S}^k\right]\boldsymbol{D}^{-1} = \boldsymbol{I}_d$, but lose the symmetry. Thus, det-CGD1 and det-CGD2 are not equivalent for our purposes.

3. Main results

Before we state the main result, we present a stepsize condition for det-CGD1 and det-CGD2, respectively:

$$\mathbb{E}\left[\boldsymbol{S}^k\boldsymbol{D}\boldsymbol{L}\boldsymbol{D}\boldsymbol{S}^k\right] \preceq \boldsymbol{D}, \tag{3}$$

and

$$\mathbb{E}\left[\boldsymbol{D}\boldsymbol{T}^k\boldsymbol{L}\boldsymbol{T}^k\boldsymbol{D}\right] \preceq \boldsymbol{D}. \tag{4}$$

In the case of vanilla GD, (3) and (4) become $\gamma < L^{-1}$, which is the standard condition for convergence. Below is the main convergence theorem for both algorithms in the single-node regime.

Theorem 1 *Suppose that Assumptions 1-3 are satisfied. Then, for each $k \geq 0$*

$$\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}\left[\left\|\nabla f(x^k)\right\|^2_{\frac{\boldsymbol{D}}{\det(\boldsymbol{D})^{1/d}}}\right] \leq \frac{2(f(x^0)-f^{\inf})}{\det(\boldsymbol{D})^{1/d}K}. \tag{5}$$

if one of the below conditions is true:

i) The vectors x^k are the iterates of det-CGD1 and $\boldsymbol{D}$ satisfies (3);

ii) The vectors x^k are the iterates of det-CGD2 and $\boldsymbol{D}$ satisfies (4).

Adjusting the weight matrix to $\frac{\boldsymbol{D}}{\det(\boldsymbol{D})^{1/d}}$ allows its determinant to be 1, making the norm on the left-hand side comparable to the standard Euclidean norm. It is important to note that the volume of the normalized ellipsoid $\left\{x \in \mathbb{R}^d \ : \ \|x\|^2_{\boldsymbol{D}/\det(\boldsymbol{D})^{1/d}} \leq 1\right\}$ does not depend on the choice of $\boldsymbol{D} \in \mathbb{S}^d_{++}$. Therefore, the results of (5) are comparable across different $\boldsymbol{D}$ in the sense that the right-hand side of (5) measures the volume of the ellipsoid containing the gradient.

3.1. Optimal matrix stepsize

In this section, we describe how to choose the optimal stepsize that minimizes the iteration complexity. The problem is easier for det-CGD2. We notice that (4) can be explicitly solved. Specifically, it is equivalent to

$$\boldsymbol{D} \preceq \left(\mathbb{E}\left[\boldsymbol{T}^k\boldsymbol{L}\boldsymbol{T}^k\right]\right)^{-1}. \tag{6}$$

We want to emphasize that the RHS matrix is invertible despite the sketches not being so. Indeed. The map $h : \boldsymbol{T} \to \boldsymbol{T}\boldsymbol{L}\boldsymbol{T}$ is convex on $\mathbb{S}^d_+$. Therefore, Jensen's inequality implies

$$\mathbb{E}\left[\boldsymbol{T}^k\boldsymbol{L}\boldsymbol{T}^k\right] \succeq \mathbb{E}\left[\boldsymbol{T}^k\right]\boldsymbol{L}\mathbb{E}\left[\boldsymbol{T}^k\right] = \boldsymbol{L} \succ \boldsymbol{O}_d.$$

This explicit condition on $\boldsymbol{D}$ can assist in determining the optimal stepsize. Since both $\boldsymbol{D}$ and $(\boldsymbol{T}^k\boldsymbol{L}\boldsymbol{T}^k)^{-1}$ are positive definite, then the right-hand side of (5) is minimized exactly when

$$\boldsymbol{D} = \left(\mathbb{E}\left[\boldsymbol{T}^k\boldsymbol{L}\boldsymbol{T}^k\right]\right)^{-1}. \tag{7}$$

Note that the explicit solution of $\boldsymbol{D}$ needs to be calculated only once, at the beginning of the algorithm. It is then fixed for all iterations. The situation is different for det-CGD1. According to (5), the optimal $\boldsymbol{D}$ is defined as the solution of the following constrained optimization problem:

$$\begin{aligned} \text{minimize} \quad & \log\det(\boldsymbol{D}^{-1}) \\ \text{subject to} \quad & \mathbb{E}\left[\boldsymbol{S}^k\boldsymbol{D}\boldsymbol{L}\boldsymbol{D}\boldsymbol{S}^k\right] \preceq \boldsymbol{D} \\ & \boldsymbol{D} \in \mathbb{S}^d_{++}. \end{aligned} \tag{8}$$

Proposition 2 *The optimization problem (8) with respect to stepsize matrix $\boldsymbol{D} \in \mathbb{S}^d_{++}$, is a convex optimization problem with a convex constraint.*

The proof of this proposition can be found in the Appendix. One could consider using the CVXPY [9] package to solve (8), provided that it is first transformed into a Disciplined Convex Programming (DCP) form [17].

Benefits of the layer-wise structure. In the case, when $\boldsymbol{L} = \mathrm{Diag}(\boldsymbol{L}_1, \ldots, \boldsymbol{L}_\ell)$, $\boldsymbol{D} = \mathrm{Diag}(\boldsymbol{D}_1, \ldots, \boldsymbol{D}_\ell)$ and $\boldsymbol{S}^k = \mathrm{Diag}(\boldsymbol{S}^k_1, \ldots, \boldsymbol{S}^k_\ell)$, where $\boldsymbol{L}_i, \boldsymbol{D}_i, \boldsymbol{S}^k_i \in \mathbb{S}^{d_i}_{++}$ we propose a relaxation of (8). In Appendix D we give a table that gives the communication complexity of each algorithm for different sketches and stepsizes. Furthermore, we show that for certain compressors, we are able to get compression for free.

Convergence in distributed setting. We also extend both det-CGD1 and det-CGD2 to the distributed case in Appendix E, where the objective is given as follows,

$$f(x) := \frac{1}{n}\sum_{i=1}^n f_i(x),$$

Under the sum decomposable setting (25), we prove convergence of the two algorithms to a neighborhood of stationarity given certain stepsize conditions are satisfied. Similar to the single-node case, some additional efforts are needed to determine the optimal matrix stepsize.

4. Future work

Matrix stepsize gradient methods are still not well studied and require further analysis. Although many important algorithms have been proposed using scalar stepsizes and are known to have good performance, their matrix analogs have yet to be thoroughly examined. The distributed algorithms proposed in Appendix E follow the structure of DCGD by [27]. However, other federated learning mechanisms such as MARINA, which has variance reduction [13], or EF21 by [39], which has powerful practical performance, should also be explored.

References

[1] Mehiddin Al-Baali and H Khalfan. An overview of some practical quasi-newton methods for unconstrained optimization. *Sultan Qaboos University Journal for Science [SQUJS]*, 12(2): 199–209, 2007.

[2] Mehiddin Al-Baali, Emilio Spedicato, and Francesca Maggioni. Broyden's quasi-Newton methods for a nonlinear system of equations and unconstrained optimization: a review and open problems. *Optimization Methods and Software*, 29(5):937–954, 2014.

[3] Dan Alistarh, Demjan Grubic, Jerry Li, Ryota Tomioka, and Milan Vojnovic. QSGD: Communication-efficient SGD via gradient quantization and encoding. *Advances in neural information processing systems*, 30, 2017.

[4] Charles G Broyden. A class of methods for solving nonlinear simultaneous equations. *Mathematics of computation*, 19(92):577–593, 1965.

[5] Sébastien Bubeck et al. Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, 2015.

[6] Chih-Chung Chang and Chih-Jen Lin. Libsvm: a library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, 2(3):1–27, 2011.

[7] Marina Danilova, Pavel Dvurechensky, Alexander Gasnikov, Eduard Gorbunov, Sergey Guminov, Dmitry Kamzolov, and Innokentiy Shibaev. Recent theoretical advances in non-convex optimization. In *High-Dimensional Optimization and Probability: With a View Towards Data Science*, pages 79–163. Springer, 2022.

[8] John E Dennis, Jr and Jorge J Moré. Quasi-Newton methods, motivation and theory. *SIAM review*, 19(1):46–89, 1977.

[9] Steven Diamond and Stephen Boyd. CVXPY: A Python-embedded modeling language for convex optimization. *The Journal of Machine Learning Research*, 17(1):2909–2913, 2016.

[10] Aritra Dutta, El Houcine Bergou, Ahmed M Abdelmoniem, Chen-Yu Ho, Atal Narayan Sahu, Marco Canini, and Panos Kalnis. On the discrepancy between the theoretical analysis and practical implementations of compressed communication for distributed deep learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 3817–3824, 2020.

[11] Darina Dvinskikh, Aleksandr Ogaltsov, Alexander Gasnikov, Pavel Dvurechensky, Alexander Tyurin, and Vladimir Spokoiny. Adaptive gradient descent for convex and non-convex stochastic optimization. *arXiv preprint arXiv:1911.08380*, 2019.

[12] Boris Ginsburg, Patrice Castonguay, Oleksii Hrinchuk, Oleksii Kuchaiev, Vitaly Lavrukhin, Ryan Leary, Jason Li, Huyen Nguyen, Yang Zhang, and Jonathan M Cohen. Stochastic gradient methods with layer-wise adaptive moments for training of deep networks. *arXiv preprint arXiv:1905.11286*, 2019.

[13] Eduard Gorbunov, Konstantin P Burlachenko, Zhize Li, and Peter Richtárik. Marina: Faster non-convex distributed learning with compression. In *International Conference on Machine Learning*, pages 3788–3798. PMLR, 2021.

[14] Robert M Gower and Peter Richtárik. Randomized iterative methods for linear systems. *SIAM Journal on Matrix Analysis and Applications*, 36(4):1660–1690, 2015.

[15] Robert Mansel Gower, Nicolas Loizou, Xun Qian, Alibek Sailanbayev, Egor Shulgin, and Peter Richtárik. Sgd: General analysis and improved rates. In *International Conference on Machine Learning*, pages 5200–5209. PMLR, 2019.

[16] William B Gragg and Richard A Tapia. Optimal error bounds for the Newton–Kantorovich theorem. *SIAM Journal on Numerical Analysis*, 11(1):10–13, 1974.

[17] Michael Grant, Stephen Boyd, and Yinyu Ye. Disciplined convex programming. *Global optimization: From theory to implementation*, pages 155–210, 2006.

[18] SV Guminov, Yu E Nesterov, PE Dvurechensky, and AV Gasnikov. Accelerated primal-dual gradient descent with linesearch for convex, nonconvex, and nonsmooth optimization problems. In *Doklady Mathematics*, volume 99, pages 125–128. Springer, 2019.

[19] Filip Hanzely, Konstantin Mishchenko, and Peter Richtárik. SEGA: Variance reduction via gradient sketching. *Advances in Neural Information Processing Systems*, 31, 2018.

[20] Samuel Horvath, Chen-Yu Ho, Ludovit Horvath, Atal Narayan Sahu, Marco Canini, and Peter Richtarik. Natural compression for distributed deep learning. *arXiv preprint arXiv:1905.10988*, 2019.

[21] Samuel Horváth, Dmitry Kovalev, Konstantin Mishchenko, Peter Richtárik, and Sebastian Stich. Stochastic distributed learning with gradient quantization and double-variance reduction. *Optimization Methods and Software*, 38(1):91–106, 2023.

[22] Rustem Islamov, Xun Qian, and Peter Richtárik. Distributed second order methods with fast rates and compressed communication. In *International conference on machine learning*, pages 4617–4628. PMLR, 2021.

[23] Sashank J Reddi, Suvrit Sra, Barnabas Poczos, and Alexander J Smola. Proximal stochastic methods for nonsmooth nonconvex finite-sum optimization. *Advances in neural information processing systems*, 29, 2016.

[24] Prateek Jain, Purushottam Kar, et al. Non-convex optimization for machine learning. *Foundations and Trends® in Machine Learning*, 10(3-4):142–363, 2017.

[25] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210, 2021.

[26] Ahmed Khaled and Peter Richtárik. Better theory for sgd in the nonconvex world. *arXiv preprint arXiv:2002.03329*, 2020.

[27] Sarit Khirirat, Hamid Reza Feyzmahdavian, and Mikael Johansson. Distributed learning with compressed gradients. *arXiv preprint arXiv:1806.06573*, 2018.

[28] Jakub Konečnỳ, H Brendan McMahan, Felix X Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, 2016.

[29] Yin Tat Lee, Zhao Song, and Qiuyi Zhang. Solving empirical risk minimization in the current matrix multiplication time. In *Conference on Learning Theory*, pages 2140–2157. PMLR, 2019.

[30] Zhize Li, Dmitry Kovalev, Xun Qian, and Peter Richtárik. Acceleration for compressed gradient descent in distributed and federated optimization. *arXiv preprint arXiv:2002.11364*, 2020.

[31] Zhize Li, Hongyan Bao, Xiangliang Zhang, and Peter Richtárik. PAGE: A simple and optimal probabilistic gradient estimator for nonconvex optimization. In *International conference on machine learning*, pages 6286–6295. PMLR, 2021.

[32] Artavazd Maranjyan, Mher Safaryan, and Peter Richtárik. GradSkip: Communication-Accelerated Local Gradient Methods with Better Computational Complexity. *arXiv preprint arXiv:2210.16402*, 2022.

[33] H. Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2017. URL `http://arxiv.org/abs/1602.05629`.

[34] George J Miel. Majorizing sequences and error bounds for iterative methods. *Mathematics of Computation*, 34(149):185–202, 1980.

[35] Konstantin Mishchenko, Eduard Gorbunov, Martin Takáč, and Peter Richtárik. Distributed learning with compressed gradient differences. *arXiv preprint arXiv:1901.09269*, 2019.

[36] Konstantin Mishchenko, Grigory Malinovsky, Sebastian Stich, and Peter Richtarik. ProxSkip: Yes! Local gradient steps provably lead to communication acceleration! Finally! In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 15750–15769. PMLR, 17–23 Jul 2022. URL `https://proceedings.mlr.press/v162/mishchenko22b.html`.

[37] Eric Moulines and Francis Bach. Non-asymptotic analysis of stochastic approximation algorithms for machine learning. *Advances in neural information processing systems*, 24, 2011.

[38] Lianke Qin, Zhao Song, Lichen Zhang, and Danyang Zhuo. An online and unified algorithm for projection matrix vector multiplication with application to empirical risk minimization. In *International Conference on Artificial Intelligence and Statistics*, pages 101–156. PMLR, 2023.

[39] Peter Richtárik, Igor Sokolov, and Ilyas Fatkhullin. EF21: A new, simpler, theoretically better, and practically faster error feedback. *Advances in Neural Information Processing Systems*, 34: 4384–4396, 2021.

[40] Mher Safaryan, Filip Hanzely, and Peter Richtárik. Smoothness matrices beat smoothness constants: Better communication compression techniques for distributed optimization. *Advances in Neural Information Processing Systems*, 34:25688–25702, 2021.

[41] Mher Safaryan, Egor Shulgin, and Peter Richtárik. Uncertainty principle for communication compression in distributed and federated learning and the search for an optimal compressor. *Information and Inference: A Journal of the IMA*, 11(2):557–580, 2022.

[42] Zhao Song, Yitan Wang, Zheng Yu, and Lichen Zhang. Sketching for first order method: Efficient algorithm for low-bandwidth channel and vulnerability. In *International Conference on Machine Learning*, pages 32365–32417. PMLR, 2023.

[43] Sebastian U Stich. Unified optimal analysis of the (stochastic) gradient method. *arXiv preprint arXiv:1907.04232*, 2019.

[44] Bokun Wang, Mher Safaryan, and Peter Richtárik. Theoretically better and numerically faster distributed optimization with smoothness-aware quantization techniques. *Advances in Neural Information Processing Systems*, 35:9841–9852, 2022.

[45] Tetsuro Yamamoto. A convergence theorem for newton-like methods in banach spaces. *Numerische Mathematik*, 51:545–557, 1987.

[46] Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. Understanding neural networks through deep visualization. *arXiv preprint arXiv:1506.06579*, 2015.

[47] Adams Wei Yu, Lei Huang, Qihang Lin, Ruslan Salakhutdinov, and Jaime Carbonell. Block-normalized gradient method: An empirical study for training deep neural network. *arXiv preprint arXiv:1707.04822*, 2017.

[48] Jingzhao Zhang, Sai Praneeth Karimireddy, Andreas Veit, Seungyeon Kim, Sashank Reddi, Sanjiv Kumar, and Suvrit Sra. Why are adaptive methods good for attention models? *Advances in Neural Information Processing Systems*, 33:15383–15393, 2020.

[49] Qinghe Zheng, Xinyu Tian, Nan Jiang, and Mingqiang Yang. Layer-wise learning based stochastic gradient descent method for the optimization of deep convolutional neural network. *Journal of Intelligent & Fuzzy Systems*, 37(4):5641–5654, 2019.

[50] Luisa M Zintgraf, Taco S Cohen, Tameem Adel, and Max Welling. Visualizing deep neural network decisions: Prediction difference analysis. *arXiv preprint arXiv:1702.04595*, 2017.

Contents

A Preliminaries **11**

B Related work **11**

C Single node case **12**
C.1 Proof of Theorem 1 . . . 12
C.2 Proof of Proposition 2 . . . 13

D Leveraging the layer-wise structure **14**
D.1 Proof of Theorem 4 . . . 16
D.2 Bernoulli-q sketch for det-CGD2 . . . 17
D.3 General cases for det-CGD1 . . . 19
D.4 General cases for det-CGD2 . . . 19
D.5 Interpretations of Table 1 . . . 19
D.5.1 Comparison of row 5 and 7 . . . 19
D.5.2 Comparison of row 6 and 7 . . . 20

E Distributed Case **21**
E.1 Distributed det-CGD1 . . . 21
E.2 Proof of Theorem 7 . . . 23
E.3 Convexity of the constraints . . . 27
E.3.1 Proof of Corollary 8 . . . 28
E.4 Distributed det-CGD2 . . . 29
E.4.1 Analysis of distributed det-CGD2 . . . 29
E.4.2 Optimal stepsize . . . 32
E.5 DCGD with constant stepsize . . . 32

F Proofs of technical lemmas **32**
F.1 Proof of Lemma 3 . . . 33
F.2 Proof of Lemma 9 . . . 34
F.3 Proof of Lemma 10 . . . 34
F.4 Proof of Lemma 11 . . . 34
F.5 Proof of Lemma 12 . . . 35
F.6 Proof of Lemma 13 . . . 35
F.7 Proof of Lemma 15 . . . 35

G Experiments **36**
G.1 Single node case . . . 36
G.1.1 Comparison to CGD with scalar stepsize, scalar smoothness constant . . . 36
G.1.2 Comparison of the two algorithms under the same stepsize . . . 39
G.2 Distributed case . . . 39
G.2.1 Comparison to standard DCGD in the distributed case . . . 41

Appendix A. Preliminaries

The usual Euclidean norm on $\mathbb{R}^d$ is defined as $\|\cdot\|$. We use bold capital letters to denote matrices. By $\boldsymbol{I}_d$ we denote the $d \times d$ identity matrix, and by $\boldsymbol{O}_d$ we denote the $d \times d$ zero matrix. Let $\mathbb{S}^d_{++}$ (resp. $\mathbb{S}^d_{+}$) be the set of $d \times d$ symmetric positive definite (resp. semi-definite) matrices. Given $\boldsymbol{Q} \in \mathbb{S}^d_{++}$ and $x \in \mathbb{R}^d$, we write $\|x\|_{\boldsymbol{Q}} := \sqrt{\langle \boldsymbol{Q}x, x\rangle}$, where $\langle \cdot, \cdot \rangle$ is the standard Euclidean inner product on $\mathbb{R}^d$. For a matrix $\boldsymbol{A} \in \mathbb{S}^d_{++}$, we define by $\lambda_{\max}(\boldsymbol{A})$ (resp. $\lambda_{\min}(\boldsymbol{A})$) the largest (resp. smallest) eigenvalue of the matrix $\boldsymbol{A}$. Let $\boldsymbol{A}_i \in \mathbb{R}^{d_i \times d_i}$ and $d = d_1 + \ldots + d_\ell$. Then the matrix $\boldsymbol{A} = \mathrm{Diag}(\boldsymbol{A}_1, \ldots, \boldsymbol{A}_\ell)$ is defined as a block diagonal $d \times d$ matrix where the i-th block is equal to $\boldsymbol{A}_i$. We will use $\mathrm{diag}(\boldsymbol{A}) \in \mathbb{R}^{d \times d}$ to denote the diagonal of any matrix $\boldsymbol{A} \in \mathbb{R}^{d \times d}$. Given a function $f : \mathbb{R}^d \to \mathbb{R}$, its gradient and its Hessian at point $x \in \mathbb{R}^d$ are respectively denoted as $\nabla f(x)$ and $\nabla^2 f(x)$.

Appendix B. Related work

Many successful convex optimization techniques have been adapted for use in the non-convex setting. Here is a non-exhaustive list: adaptivity [11, 48], variance reduction [23, 31], and acceleration [18]. A paper of particular importance for our work is that of [26], which proposes a unified scheme for analyzing stochastic gradient descent in the non-convex regime. A comprehensive overview of non-convex optimization can be found in [7, 24].

A classical example of a matrix stepsized method is Newton's method. This method has been popular in the optimization community for a long time [16, 34, 45]. However, computing the stepsize as the inverse Hessian of the current iteration results in significant computational complexity. Instead, quasi-Newton methods use an easily computable estimator to replace the inverse of the Hessian [1, 2, 4, 8]. An example is the Newton-Star algorithm [22], which we discuss in Section 2.

[14] analyzed sketched gradient descent by making the compressors unbiased with a sketch-and-project trick. They provided an analysis of the resulting algorithm for the linear feasibility problem. Later, [19] proposed a variance-reduced version of this method. Sketches are also of independent interest. In particular, [42] described a way of designing the distribution of sketch matrices, while [29, 38] used sketches in solving empirical risk minimization problems.

Leveraging the layer-wise structure of neural networks has been widely studied for optimizing the training loss function. For example, [49] propose SGD with different scalar stepsizes for each layer, [12, 47] propose layer-wise normalization for Stochastic Normalized Gradient Descent, and [10, 44] propose layer-wise compression in the distributed setting.

DCGD, proposed by [27], has since been improved in various ways, such as in [20, 30]. There is also a large body of literature on other federated learning algorithms with unbiased compressors [3, 13, 21, 32, 35, 36].

Appendix C. Single node case

C.1. Proof of Theorem 1

i) Using Assumption 2 with $x = x^{k+1} = x^k - \boldsymbol{D}\boldsymbol{S}^k\nabla f(x^k)$ and $y = x^k$, we get

$$\begin{aligned}\mathbb{E}\left[f(x^{k+1}) \mid x^k\right] &\leq \mathbb{E}\Big[f(x^k) + \left\langle \nabla f(x^k), -\boldsymbol{D}\boldsymbol{S}^k\nabla f(x^k)\right\rangle \\ &\quad + \frac{1}{2}\left\langle \boldsymbol{L}(-\boldsymbol{D}\boldsymbol{S}^k\nabla f(x^k)), -\boldsymbol{D}\boldsymbol{S}^k\nabla f(x^k)\right\rangle \mid x^k\Big] \\ &= f(x^k) - \left\langle \nabla f(x^k), \boldsymbol{D}\mathbb{E}\left[\boldsymbol{S}^k\right]\nabla f(x^k)\right\rangle + \frac{1}{2}\left\langle \mathbb{E}\left[\boldsymbol{S}^k\boldsymbol{D}\boldsymbol{L}\boldsymbol{D}\boldsymbol{S}^k\right]\nabla f(x^k), \nabla f(x^k)\right\rangle.\end{aligned}$$

From the unbiasedness of the sketch $\boldsymbol{S}^k$

$$\begin{aligned}\mathbb{E}\left[f(x^{k+1}) \mid x^k\right] &\leq f(x^k) - \left\langle \nabla f(x^k), \boldsymbol{D}\nabla f(x^k)\right\rangle + \frac{1}{2}\left\langle \mathbb{E}\left[\boldsymbol{S}^k\boldsymbol{D}\boldsymbol{L}\boldsymbol{D}\boldsymbol{S}^k\right]\nabla f(x^k), \nabla f(x^k)\right\rangle \\ &\overset{(3)}{\leq} f(x^k) - \left\langle \nabla f(x^k), \boldsymbol{D}\nabla f(x^k)\right\rangle + \frac{1}{2}\left\langle \boldsymbol{D}\nabla f(x^k), \nabla f(x^k)\right\rangle \\ &= f(x^k) - \frac{1}{2}\left\langle \nabla f(x^k), \boldsymbol{D}\nabla f(x^k)\right\rangle \\ &= f(x^k) - \frac{1}{2}\left\|\nabla f(x^k)\right\|^2_{\boldsymbol{D}}. \qquad (9)\end{aligned}$$

Next, by subtracting $f^{\inf}$ from both sides of (9), taking expectation and applying the tower property, we get

$$\begin{aligned}\mathbb{E}\left[f(x^{k+1})\right] - f^{\inf} &= \mathbb{E}\left[\mathbb{E}\left[f(x^{k+1}) \mid x^k\right]\right] - f^{\inf} \\ &\overset{(9)}{\leq} \mathbb{E}\left[f(x^k) - \frac{1}{2}\left\|\nabla f(x^k)\right\|^2_{\boldsymbol{D}}\right] - f^{\inf} \\ &= \mathbb{E}\left[f(x^k)\right] - f^{\inf} - \frac{1}{2}\mathbb{E}\left[\left\|\nabla f(x^k)\right\|^2_{\boldsymbol{D}}\right].\end{aligned}$$

Letting $\Delta^k := \mathbb{E}\left[f(x^k)\right] - f^{\inf}$, the last inequality can be written as $\Delta^{k+1} \leq \Delta^k - \frac{1}{2}\mathbb{E}\left[\left\|\nabla f(x^k)\right\|^2_{\boldsymbol{D}}\right]$. Summing these inequalities for $k = 0, 1, \ldots, K-1$, we get a telescoping effect leading to

$$\Delta^K \leq \Delta^0 - \frac{1}{2}\sum_{k=0}^{K-1}\mathbb{E}\left[\left\|\nabla f(x^k)\right\|^2_{\boldsymbol{D}}\right].$$

It remains to rearrange the terms of this inequality, divide both sides by $K\det(\boldsymbol{D})^{1/d}$, and use the inequality $\Delta^K \geq 0$.

ii) Similar to the previous case, using matrix smoothness for $x = x^{k+1} = x^k - \boldsymbol{T}^k\boldsymbol{D}\nabla f(x^k)$ and $y = x^k$, we get

$$\begin{aligned}\mathbb{E}\left[f(x^{k+1}) \mid x^k\right] &\leq \mathbb{E}\Big[f(x^k) + \left\langle \nabla f(x^k), -\boldsymbol{T}^k\boldsymbol{D}\nabla f(x^k)\right\rangle \qquad (10) \\ &\quad + \frac{1}{2}\left\langle \boldsymbol{L}(-\boldsymbol{T}^k\boldsymbol{D}\nabla f(x^k)), -\boldsymbol{T}^k\boldsymbol{D}\nabla f(x^k)\right\rangle \mid x^k\Big] \\ &= f(x^k) - \left\langle \nabla f(x^k), \mathbb{E}\left[\boldsymbol{T}^k\right]\boldsymbol{D}\nabla f(x^k)\right\rangle \qquad (11) \\ &\quad + \frac{1}{2}\left\langle \mathbb{E}\left[\boldsymbol{D}(\boldsymbol{T}^k)^\top\boldsymbol{L}\boldsymbol{T}^k\boldsymbol{D}\right]\nabla f(x^k), \nabla f(x^k)\right\rangle.\end{aligned}$$

From Assumption 3 and condition (4) we deduce

$$\begin{aligned}\mathbb{E}\left[f(x^{k+1}) \mid x^k\right] &\leq f(x^k) - \left\langle \nabla f(x^k), \boldsymbol{D}\nabla f(x^k)\right\rangle + \frac{1}{2}\left\langle \boldsymbol{D}\nabla f(x^k), \nabla f(x^k)\right\rangle \\ &= f(x^k) - \frac{1}{2}\left\langle \nabla f(x^k), \boldsymbol{D}\nabla f(x^k)\right\rangle \\ &= f(x^k) - \frac{1}{2}\left\|\nabla f(x^k)\right\|^2_{\boldsymbol{D}}. \end{aligned} \tag{12}$$

Thus, we obtain the same upper bound on $\mathbb{E}\left[f(x^{k+1}) \mid x^k\right]$ as in (9). Following the steps from the first part, we conclude the proof.

C.2. Proof of Proposition 2

Let us rewrite (3) using quadratic forms. That is for every non-zero $v \in \mathbb{R}^d$, the following inequality must be true:

$$v^\top \mathbb{E}\left[\boldsymbol{S}^k \boldsymbol{D}\boldsymbol{L}\boldsymbol{D}\boldsymbol{S}^k\right] v \leq v^\top \boldsymbol{D} v, \qquad \forall v \neq 0$$

Notice that both sides of this inequality are real numbers, thus can be written equivalently as

$$\mathrm{tr}(v^\top \mathbb{E}\left[\boldsymbol{S}^k \boldsymbol{D}\boldsymbol{L}\boldsymbol{D}\boldsymbol{S}^k\right] v) \leq \mathrm{tr}(v^\top \boldsymbol{D} v), \qquad \forall v \neq 0$$

The LHS can be modified in the following way

$$\begin{aligned}\mathrm{tr}(v^\top \mathbb{E}\left[\boldsymbol{S}^k \boldsymbol{D}\boldsymbol{L}\boldsymbol{D}\boldsymbol{S}^k\right] v) &\overset{\text{I}}{=} \mathrm{tr}\left(\mathbb{E}\left[v^\top \boldsymbol{S}^k \boldsymbol{D}\boldsymbol{L}\boldsymbol{D}\boldsymbol{S}^k v\right]\right) \\ &\overset{\text{II}}{=} \mathbb{E}\left[\mathrm{tr}(v^\top \boldsymbol{S}^k \boldsymbol{D}\boldsymbol{L}\boldsymbol{D}\boldsymbol{S}^k v)\right] \\ &\overset{\text{III}}{=} \mathbb{E}\left[\mathrm{tr}(\boldsymbol{L}^{\frac{1}{2}}\boldsymbol{D}\boldsymbol{S}^k v v^\top \boldsymbol{S}^k \boldsymbol{D}\boldsymbol{L}^{\frac{1}{2}})\right] \\ &\overset{\text{IV}}{=} \mathrm{tr}\left(\mathbb{E}\left[\boldsymbol{L}^{\frac{1}{2}}\boldsymbol{D}\boldsymbol{S}^k v v^\top \boldsymbol{S}^k \boldsymbol{D}\boldsymbol{L}^{\frac{1}{2}}\right]\right) \\ &\overset{\text{V}}{=} \mathrm{tr}\left(\boldsymbol{L}^{\frac{1}{2}}\boldsymbol{D}\mathbb{E}\left[\boldsymbol{S}^k v v^\top \boldsymbol{S}^k\right]\boldsymbol{D}\boldsymbol{L}^{\frac{1}{2}}\right),\end{aligned}$$

where I, V are due to the linearity of expectation, II, IV are due to the linearity of trace operator, III is obtained using the cyclic property of trace. Therefore, we can write the condition (3) equivalently as

$$\mathrm{tr}\left(\boldsymbol{L}^{\frac{1}{2}}\boldsymbol{D}\mathbb{E}\left[\boldsymbol{S}^k v v^\top \boldsymbol{S}^k\right]\boldsymbol{D}\boldsymbol{L}^{\frac{1}{2}}\right) \leq \mathrm{tr}(v v^\top \boldsymbol{D}), \qquad \forall v \neq 0.$$

We then define function $g_v : \mathbb{S}^d_{++} \to \mathbb{R}$ for some fixed $v \neq 0$ as

$$g_v(\boldsymbol{D}) := \mathrm{tr}\left(\boldsymbol{L}^{\frac{1}{2}}\boldsymbol{D}\mathbb{E}\left[\boldsymbol{S}^k v v^\top \boldsymbol{S}^k\right]\boldsymbol{D}\boldsymbol{L}^{\frac{1}{2}}\right) - \mathrm{tr}(v v^\top \boldsymbol{D}). \tag{13}$$

We want to show that for every fixed $v \neq 0$, g is a convex function w.r.t $\boldsymbol{D}$, so that in this case, the sub-level set $\{\boldsymbol{D} \in \mathbb{S}^d_{++} \mid g_v(\boldsymbol{D}) \leq 0\}$ is convex.

- Notice that $vv^\top$ is a rank-1 matrix whose eigenvalues are all zero except one of them is $\|v\|^2 > 0$. We also have $(vv^\top)^\top = (v^\top)^\top v^\top = vv^\top$, so it is also a symmetric matrix. Thus we conclude that $vv^\top \in \mathbb{S}^d_+$ for every choice of v, we use $\boldsymbol{V} = vv^\top$ to denote it.

- If $\boldsymbol{S}^k = \boldsymbol{O}_d$, then the first term is equal to $\boldsymbol{O}_d$ and the function $g_v(\boldsymbol{D})$ is linear, thus, also convex. Now, let us assume $\boldsymbol{S}^k$ is nonzero. Similarly $\boldsymbol{S}^k v v^\top \boldsymbol{S}^k = \boldsymbol{S}^k v (\boldsymbol{S}^k v)^T$ is also a symmetric positive semi-definite matrix whose eigenvalues are all 0 except one of them is $\|\boldsymbol{S}^k v\|^2$, this tells us that its expectation over $\boldsymbol{S}^k$ is still a symmetric positive semi-definite matrix, we use $\boldsymbol{R} = \mathbb{E}\left[\boldsymbol{S}^k v v^\top \boldsymbol{S}^k\right]$ to denote it.

Now we can write function g_v as

$$g_v(\boldsymbol{D}) = \mathrm{tr}(\boldsymbol{L}^{\frac{1}{2}} \boldsymbol{D} \boldsymbol{R} \boldsymbol{D} \boldsymbol{L}^{\frac{1}{2}}) - \mathrm{tr}(\boldsymbol{V}\boldsymbol{D}).$$

We present the following lemma that guarantees the convexity of the first term.

Lemma 3 *For every matrix $\boldsymbol{R} \in \mathbb{S}^d_+$, we define*

$$f(\boldsymbol{D}) = \mathrm{tr}(\boldsymbol{L}^{\frac{1}{2}} \boldsymbol{D} \boldsymbol{R} \boldsymbol{D} \boldsymbol{L}^{\frac{1}{2}}), \tag{14}$$

where $\boldsymbol{L}, \boldsymbol{D} \in \mathbb{S}^d_{++}$. Then function $f : \mathbb{S}^d_{++} \to \mathbb{R}$ is a convex function.

The proof can be found in Appendix F.1. According to Lemma 3, the first term of $g_v(\boldsymbol{D})$ is a convex function, and we know that the second term is linear in $\boldsymbol{D}$. As a result, $g_v(\boldsymbol{D})$ is a convex function w.r.t. $\boldsymbol{D}$ for every $v \neq 0$, thus the sub-level set $\{\boldsymbol{D} \in \mathbb{S}^d_{++} \mid g_v(\boldsymbol{D}) \leq 0\}$ is a convex set for every $v \neq 0$. The intersection of all those convex sets corresponding to every $v \neq 0$ is still a convex set, which tells us the original condition (3) is convex. This concludes the proof of the proposition.

Appendix D. Leveraging the layer-wise structure

In this section we focus on the block-diagonal case of $\boldsymbol{L}$ for both det-CGD1 and det-CGD2. In particular, we propose hyper-parameters of det-CGD1 designed specifically for training NNs. Let us assume that $\boldsymbol{L} = \mathrm{Diag}(\boldsymbol{L}_1, \ldots, \boldsymbol{L}_\ell)$, where $\boldsymbol{L}_i \in \mathbb{S}^{d_i}_{++}$. This setting is a generalization of the classical smoothness condition, as in the latter case $\boldsymbol{L}_i = L\boldsymbol{I}_{d_i}$ for all $i = 1, \ldots, \ell$. Respectively, we choose both the sketches and the stepsize to be block diagonal: $\boldsymbol{D} = \mathrm{Diag}(\boldsymbol{D}_1, \ldots, \boldsymbol{D}_\ell)$ and $\boldsymbol{S}^k = \mathrm{Diag}(\boldsymbol{S}^k_1, \ldots, \boldsymbol{S}^k_\ell)$, where $\boldsymbol{D}_i, \boldsymbol{S}^k_i \in \mathbb{S}^{d_i}_{++}$.

Let us notice that the left hand side of the inequality constraint in (8) has quadratic dependence on $\boldsymbol{D}$, while the right hand side is linear. Thus, for every matrix $\boldsymbol{W} \in \mathbb{S}^d_{++}$, there exists $\gamma > 0$ such that

$$\gamma^2 \lambda_{\max}\left(\mathbb{E}\left[\boldsymbol{S}^k \boldsymbol{W} \boldsymbol{L} \boldsymbol{W} \boldsymbol{S}^k\right]\right) \leq \gamma \lambda_{\min}(\boldsymbol{W}).$$

Therefore, for $\gamma \boldsymbol{W}$ we deduce

$$\mathbb{E}\left[\boldsymbol{S}^k (\gamma \boldsymbol{W}) \boldsymbol{L} (\gamma \boldsymbol{W}) \boldsymbol{S}^k\right] \preceq \gamma^2 \lambda_{\max}\left(\mathbb{E}\left[\boldsymbol{S}^k \boldsymbol{W} \boldsymbol{L} \boldsymbol{W} \boldsymbol{S}^k\right]\right) \boldsymbol{I}_d \preceq \gamma \lambda_{\min}(\boldsymbol{W}) \boldsymbol{I}_d \preceq \gamma \boldsymbol{W}. \tag{15}$$

The following theorem is based on this simple fact applied to the corresponding blocks of the matrices $\boldsymbol{D}, \boldsymbol{L}, \boldsymbol{S}^k$ for det-CGD1.

Theorem 4 *Let $f : \mathbb{R}^d \to \mathbb{R}$ satisfy Assumptions 1 and 2, with $\boldsymbol{L}$ admitting the layer-separable structure $\boldsymbol{L} = \mathrm{Diag}(\boldsymbol{L}_1, \ldots, \boldsymbol{L}_\ell)$, where $\boldsymbol{L}_1, \ldots, \boldsymbol{L}_\ell \in \mathbb{S}^{d_i}_{++}$. Choose random matrices $\boldsymbol{S}^k_1, \ldots, \boldsymbol{S}^k_\ell \in$*

$\mathbb{S}_+^d$ *to satisfy Assumption 3 for all* $i \in [\ell]$*, and let* $\boldsymbol{S}^k := \mathrm{Diag}(\boldsymbol{S}_1^k, \ldots, \boldsymbol{S}_\ell^k)$*. Furthermore, choose matrices* $\boldsymbol{W}_1, \ldots, \boldsymbol{W}_\ell \in \mathbb{S}_{++}^d$ *and scalars* $\gamma_1, \ldots, \gamma_\ell > 0$ *such that*

$$\gamma_i \leq \lambda_{\max}^{-1}\left(\mathbb{E}\left[\boldsymbol{W}_i^{-1/2}\boldsymbol{S}_i^k\boldsymbol{W}_i\boldsymbol{L}_i\boldsymbol{W}_i\boldsymbol{S}_i^k\boldsymbol{W}_i^{-1/2}\right]\right) \qquad \forall i \in [\ell]. \tag{16}$$

Letting $\boldsymbol{W} := \mathrm{Diag}(\boldsymbol{W}_1, \ldots, \boldsymbol{W}_\ell)$*,* $\Gamma := \mathrm{Diag}(\gamma_1\boldsymbol{I}_{d_1}, \ldots, \gamma_\ell\boldsymbol{I}_{d_\ell})$ *and* $\boldsymbol{D} := \Gamma\boldsymbol{W}$*, we get*

$$\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}\left[\left\|\nabla f(x^k)\right\|^2_{\frac{\Gamma\boldsymbol{W}}{\det(\Gamma\boldsymbol{W})^{1/d}}}\right] \leq \frac{2(f(x^0)-f^{\inf})}{\det(\Gamma\boldsymbol{W})^{1/d}\,K}. \tag{17}$$

Table 1: Summary of communication complexities of det-CGD1 and det-CGD2 with different sketches and stepsize matrices. The $\boldsymbol{D}_i$ here for det-CGD1 is $\boldsymbol{W}_i$ with the optimal scaling determined using Theorem 4, for det-CGD2 it is the optimal stepsize matrix defined in (6). The constant $2(f(x^0)-f^{\inf})/\varepsilon^2$ is hidden, ℓ is the number of layers, k_i is the mini-batch size for the i-th layer if we use the rand-k sketch. The notation $\tilde{\boldsymbol{L}}_{i,k}$ is defined as $\frac{d-k}{d-1}\mathrm{diag}(\boldsymbol{L}_i) + \frac{k-1}{d-1}\boldsymbol{L}_i$.

No.	The method	$\left(\boldsymbol{S}_i^k, \boldsymbol{D}_i\right)$	$l \geq 1$, d_i, k_i, $\sum_{i=1}^{\ell} k_i = k$, layer structure	$l = 1$, $k_i = k$, general structure
1.	det-CGD1	$\left(\boldsymbol{I}_d, \gamma\boldsymbol{L}_i^{-1}\right)$	$d \cdot \det(\boldsymbol{L})^{1/d}$	$d \cdot \det(\boldsymbol{L})^{1/d}$
2.	det-CGD1	$\left(\boldsymbol{I}_d, \gamma\,\mathrm{diag}^{-1}(\boldsymbol{L}_i)\right)$	$d \cdot \det\left(\mathrm{diag}(\boldsymbol{L})\right)^{1/d}$	$d \cdot \det\left(\mathrm{diag}(\boldsymbol{L})\right)^{1/d}$
3.	det-CGD1	$\left(\boldsymbol{I}_d, \gamma\boldsymbol{I}_{d_i}\right)$	$d \cdot \left(\prod_{i=1}^{l}\lambda_{\max}(\boldsymbol{L}_i)\right)^{1/d}$	$d \cdot \lambda_{\max}(\boldsymbol{L})$
4.	det-CGD1	$\left(\text{rand-1}, \gamma\boldsymbol{I}_{d_i}\right)$	$\ell \cdot \left(\prod_{i=1}^{l} d_i^{d_i}\left(\max_j(\boldsymbol{L}_i)_{jj}\right)^{d_i}\right)^{1/d}$	$d \cdot \max_j(\boldsymbol{L}_{jj})$
5.	det-CGD1	$\left(\text{rand-1}, \gamma\boldsymbol{L}_i^{-1}\right)$	$\ell \cdot \left(\frac{\prod_{i=1}^{l} d_i^{d_i}\lambda_{\max}^{d_i}\left(\boldsymbol{L}_i^{\frac{1}{2}}\,\mathrm{diag}(\boldsymbol{L}_i^{-1})\boldsymbol{L}_i^{\frac{1}{2}}\right)}{\prod_{i=1}^{l}\det(\boldsymbol{L}_i^{-1})}\right)^{1/d}$	$\frac{d\lambda_{\max}\left(\boldsymbol{L}^{\frac{1}{2}}\,\mathrm{diag}\left(\boldsymbol{L}^{-1}\right)\boldsymbol{L}^{\frac{1}{2}}\right)}{\det\left(\boldsymbol{L}^{-1}\right)^{1/d}}$
6.	det-CGD1	$\left(\text{rand-1}, \gamma\boldsymbol{L}_i^{-1/2}\right)$	$\ell \cdot \left(\frac{\prod_{i=1}^{l} d_i^{d_i}\lambda_{\max}^{d_i}(\boldsymbol{L}_i^{1/2})}{\prod_{i=1}^{l}\det(\boldsymbol{L}_i^{-1/2})}\right)^{1/d}$	$d \cdot \lambda_{\max}^{1/2}(\boldsymbol{L})\det(\boldsymbol{L})^{1/(2d)}$
7.	det-CGD1	$\left(\text{rand-1}, \gamma\,\mathrm{diag}^{-1}(\boldsymbol{L}_i)\right)$	$\ell \cdot \left(\frac{\prod_{i=1}^{l} d_i^{d_i}}{\prod_{j=1}^{d}(\boldsymbol{L}_{jj}^{-1})}\right)^{1/d}$	$d \cdot \det\left(\mathrm{diag}(\boldsymbol{L})\right)^{1/d}$
8.	det-CGD1	$\left(\text{rand-}k_i, \gamma\,\mathrm{diag}^{-1}(\boldsymbol{L}_i)\right)$	$k \cdot \left(\prod_{i=1}^{l}\left(\frac{d_i}{k_i}\right)^{d_i}\det\left(\mathrm{diag}(\boldsymbol{L})\right)\right)^{1/d}$	$d \cdot \det\left(\mathrm{diag}(\boldsymbol{L})\right)^{1/d}$
9.	det-CGD2	$\left(\boldsymbol{I}_d, \boldsymbol{L}_i^{-1}\right)$	$d \cdot \det(\boldsymbol{L})^{1/d}$	$d \cdot \det(\boldsymbol{L})^{1/d}$
10.	det-CGD2	$\left(\text{rand-1}, \frac{\mathrm{diag}^{-1}(\boldsymbol{L}_i)}{d_i}\right)$	$\ell \cdot \left(\prod_{i=1}^{l} d_i^{d_i}\right)^{1/d}\det(\mathrm{diag}\,\boldsymbol{L})^{1/d}$	$d \cdot \det(\mathrm{diag}(\boldsymbol{L}))^{1/d}$
11.	det-CGD2	$\left(\text{rand-}k, \frac{k_i}{d_i}\tilde{\boldsymbol{L}}_{i,k_i}^{-1}\right)$	$k \cdot \left(\prod_{i=1}^{l}\left(\frac{d_i}{k_i}\right)^{\frac{d_i}{d}}\right)\left(\prod_{i=1}^{l}\det(\tilde{\boldsymbol{L}}_{i,k_i})\right)^{1/d}$	$d \cdot \det(\tilde{\boldsymbol{L}}_{1,k})$
12.	det-CGD2	$\left(\text{Bern-}q_i, q_i\boldsymbol{L}_i^{-1}\right)$	$\left(\sum_{i=1}^{l} q_i d_i\right) \cdot \prod_{i=1}^{l}\left(\frac{1}{q_i}\right)^{\frac{d_i}{d}}\det(\boldsymbol{L})^{1/d}$	$d \cdot \det(\boldsymbol{L})^{1/d}$
13.	GD	$\left(\boldsymbol{I}_d, \lambda_{\max}^{-1}(\boldsymbol{L})\boldsymbol{I}_d\right)$	N/A	$d \cdot \lambda_{\max}(\boldsymbol{L})$

In particular, if the scalars $\{\gamma_i\}$ are chosen to be equal to their maximum allowed values from (16), then the convergence factor of (17) is equal to

$$\det(\Gamma\boldsymbol{W})^{-\frac{1}{d}} = \left[\prod_{i=1}^{\ell}\lambda_{\max}^{d_i}\left(\mathbb{E}\left[\boldsymbol{W}_i^{-\frac{1}{2}}\boldsymbol{S}_i^k\boldsymbol{W}_i\boldsymbol{L}_i\boldsymbol{W}_i\boldsymbol{S}_i^k\boldsymbol{W}_i^{-\frac{1}{2}}\right]\right)\right]^{\frac{1}{d}}\det(\boldsymbol{W}^{-1})^{\frac{1}{d}}.$$

Table 1 contains the (expected) communication complexities of det-CGD1, det-CGD2 and GD for several choices of $\boldsymbol{W}, \boldsymbol{D}$ and $\boldsymbol{S}^k$. Here are a few comments about the table. We deduce that taking a matrix stepsize without compression (row 1) we improve GD (row 13). A careful analysis reveals that the result in row 5 is always worse than row 7 in terms of both communication and iteration complexity. However, the results in row 6 and row 7 are not comparable in general, meaning that neither of them is universally better. More discussion on this table can be found in the Appendix.

Compression for free. Now, let us focus on row 12, which corresponds to a sampling scheme where the i-th layer is independently selected with probability q_i. Mathematically, it goes as follows:

$$\boldsymbol{T}_i^k = \frac{\eta_i}{q_i}\boldsymbol{I}_{d_i}, \quad \text{where} \quad \eta_i \sim \text{Bernoulli}(q_i). \tag{18}$$

Jensen's inequality implies that

$$\left(\sum_{i=1}^{l} q_i d_i\right) \cdot \prod_{i=1}^{l} \left(\frac{1}{q_i}\right)^{\frac{d_i}{d}} \geq d. \tag{19}$$

The equality is attained when $q_i = q$ for all $i \in [\ell]$. The expected bits transferred per iteration of this algorithm is then equal to $k_{\text{exp}} = qd$ and the communication complexity equals $d\det(\boldsymbol{L})^{1/d}$. Comparing with the results for det-CGD2 with rand-k_{exp} on row 11 and using the fact that $\det(\boldsymbol{L}) \leq \det(\text{diag}(\boldsymbol{L}))$, we deduce that the Bernoulli scheme is better than the uniform sampling scheme. Notice also, the communication complexity matches the one for the uncompressed det-CGD2 displayed on row 9. This, in particular means that using the Bern-q sketches we can compress the gradients for free. The latter means that we reduce the number of bits broadcasted at each iteration without losing in the total communication complexity. In particular, when all the layers have the same width d_i, the number of broadcasted bits for each iteration is reduced by a factor of q.

In the rest of the section, we provide interpretations about some of the results and conclusions we had in Appendix D.

D.1. Proof of Theorem 4

Note that $\mathbb{E}\left[\boldsymbol{S}^k\boldsymbol{D}\boldsymbol{L}\boldsymbol{D}\boldsymbol{S}^k\right] = \text{Diag}\left(\boldsymbol{Q}_1^k, \ldots, \boldsymbol{Q}_\ell^k\right)$, where $\boldsymbol{Q}_i^k := \gamma_i^2\mathbb{E}\left[\boldsymbol{S}_i^k\boldsymbol{W}_i\boldsymbol{L}_i\boldsymbol{D}_i\boldsymbol{S}_i^k\right]$. In other words,

$$\mathbb{E}\left[\boldsymbol{S}^k\boldsymbol{D}\boldsymbol{L}\boldsymbol{D}\boldsymbol{S}^k\right] = \begin{pmatrix} \boldsymbol{Q}_1^k & 0 & \cdots & 0 \\ 0 & \boldsymbol{Q}_2^k & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{Q}_\ell^k \end{pmatrix},$$

which means that (3) holds if and only if $\boldsymbol{Q}_i^k \preceq \gamma_i\boldsymbol{W}_i$ for all $i \in [\ell]$, which holds if and only if (16) holds. Therefore, Theorem 1 applies, and we conclude that

$$\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}\left[\left\|\nabla f(x^k)\right\|_{\Gamma\boldsymbol{W}}^2\right] \leq \frac{2(f(x^0) - f^{\inf})}{K}. \tag{20}$$

To obtain (17), it remains to multiply both sides of (20) by $\frac{1}{\det(\Gamma\boldsymbol{W})^{1/d}}$.

D.2. Bernoulli-q sketch for det-CGD2

The following corollary of Theorem 4 computes the communication complexity of det-CGD2 in the block diagonal setting with Bernoulli-q.

Corollary 5 *Let $\boldsymbol{T}_i^k$ for the i-th layer in det-CGD2 be the Bern-q_i sketch which is defined as*

$$\boldsymbol{T}_i^k = \frac{\eta_i}{q_i}\boldsymbol{I}_{d_i}, \quad \text{where} \quad \eta_i \sim \text{Bernoulli}(q_i). \tag{21}$$

Then, the communication complexity of (det-CGD2) *is given by,*

$$\frac{2(f(x^0)-f^{\inf})}{\varepsilon^2}\left(\sum_{i=1}^{\ell} q_i d_i\right)\prod_{i=1}^{\ell}\left(\frac{1}{q_i}\right)^{\frac{d_i}{d}}\det(\boldsymbol{L})^{\frac{1}{d}}. \tag{22}$$

Furthermore, the communication complexity is minimized if the probabilities when $q_i = q, \quad \forall i \in [\ell]$ and the minimum value is equal to

$$\frac{2(f(x^0)-f^{\inf})\cdot dx\ \det(\boldsymbol{L})^{\frac{1}{d}}}{\varepsilon^2}. \tag{23}$$

Proof For det-CGD2, its convergence requires (6). We are using Bernoulli sketch here, so we deduce that

$$\begin{aligned}\mathbb{E}\left[\boldsymbol{T}^k\boldsymbol{L}\boldsymbol{T}^k\right] &= \mathbb{E}\left[\text{Diag}(\boldsymbol{T}_1^k\boldsymbol{L}_1\boldsymbol{T}_1^k, ..., \boldsymbol{T}_\ell^k\boldsymbol{L}_\ell\boldsymbol{T}_\ell^k)\right]\\ &= \text{Diag}\left(\mathbb{E}\left[\boldsymbol{T}_1^k\boldsymbol{L}_1\boldsymbol{T}_1^k\right], ..., \mathbb{E}\left[\boldsymbol{T}_\ell^k\boldsymbol{L}_\ell\boldsymbol{T}_\ell^k\right]\right).\end{aligned}$$

Using the fact that for each block, we have

$$\mathbb{E}\left[\boldsymbol{T}_i^k\boldsymbol{L}_i\boldsymbol{T}_i^k\right] = (1-q_i)\boldsymbol{O}_{d_i}\boldsymbol{L}_i\boldsymbol{O}_{d_i} + q_i\cdot\frac{1}{q_i^2}\boldsymbol{I}_{d_i}\boldsymbol{L}_i\boldsymbol{I}_{d_i} = \frac{\boldsymbol{L}_i}{q_i},$$

we obtain

$$\mathbb{E}\left[\boldsymbol{T}^k\boldsymbol{L}\boldsymbol{T}^k\right] = \text{Diag}\left(\frac{\boldsymbol{L}_1}{q_1}, ..., \frac{\boldsymbol{L}_\ell}{q_\ell}\right).$$

Recalling (6), the best stepsize possible is therefore given by

$$\begin{aligned}\boldsymbol{D} &= \left(\mathbb{E}\left[\boldsymbol{T}^k\boldsymbol{L}\boldsymbol{T}^k\right]\right)^{-1}\\ &= \text{Diag}^{-1}\left(\frac{\boldsymbol{L}_1}{q_1}, ..., \frac{\boldsymbol{L}_\ell}{q_\ell}\right)\\ &= \text{Diag}\left(q_1\boldsymbol{L}_1^{-1}, ..., q_\ell\boldsymbol{L}_\ell^{-1}\right).\end{aligned}$$

From (5), we know that in order for det-CGD2 to converge to ε^2 error level, we need

$$\frac{2(f(x^0)-f^{\inf})}{\det(\boldsymbol{D})^{\frac{1}{d}}\,K} \leq \varepsilon^2,$$

which means that we need

$$K \geq \frac{2(f(x^0) - f^{\inf})}{\det(\boldsymbol{D})^{\frac{1}{d}}\,\varepsilon^2} = \frac{1}{\det(\boldsymbol{D})^{\frac{1}{d}}} \cdot \frac{2(f(x^0) - f^{\inf})}{\varepsilon^2},$$

iterations. For each iteration, the number of bits sent in expectation is equal to $\sum_{i=1}^{\ell} q_i d_i$. As a result, the communication complexity is given by, if we leave out the constant factor $2(f(x^0) - f^{\inf})/\varepsilon^2$,

$$\begin{aligned}
\left(\sum_{i=1}^{\ell} q_i d_i\right) \cdot \frac{1}{\det(\boldsymbol{D})^{\frac{1}{d}}} &= \left(\sum_{i=1}^{\ell} q_i d_i\right) \cdot \det(\boldsymbol{D}^{-1})^{\frac{1}{d}} \\
&= \left(\sum_{i=1}^{\ell} q_i d_i\right) \cdot \left(\prod_{i=1}^{\ell} \det(\frac{\boldsymbol{L}_i}{q_i})\right)^{\frac{1}{d}} \\
&= \left(\sum_{i=1}^{\ell} q_i d_i\right) \cdot \prod_{i=1}^{\ell} \left(\frac{1}{q_i}\right)^{\frac{d_i}{d}} \left(\prod_{i=1}^{l} \det(\boldsymbol{L}_i)\right)^{\frac{1}{d}} \\
&= \left(\sum_{i=1}^{\ell} q_i d_i\right) \cdot \prod_{i=1}^{\ell} \left(\frac{1}{q_i}\right)^{\frac{d_i}{d}} \cdot \det(\boldsymbol{L})^{\frac{1}{d}}.
\end{aligned}$$

To obtain the optimal probability q_i, we can do the following transformation

$$\left(\sum_{i=1}^{\ell} q_i d_i\right) \cdot \frac{1}{\det(\boldsymbol{D})^{\frac{1}{d}}} = \left(\sum_{i=1}^{\ell} q_i \frac{d_i}{d}\right) \cdot \prod_{i=1}^{\ell} \left(\frac{1}{q_i}\right)^{\frac{d_i}{d}} \cdot d \det(\boldsymbol{L})^{\frac{1}{d}}.$$

Therefore, it is equivalent to minimizing the coefficient

$$\left(\sum_{i=1}^{\ell} q_i \frac{d_i}{d}\right) \cdot \prod_{i=1}^{\ell} \left(\frac{1}{q_i}\right)^{\frac{d_i}{d}}.$$

If we denote $\alpha_i = \frac{d_i}{d}$, then we know that $\alpha_i \in (0, 1]$ and $\sum_{i=1}^{\ell} \alpha_i = 1$, the above coefficient turns into

$$\left(\sum_{i=1}^{\ell} \alpha_i q_i\right) \prod_{i=1}^{\ell} \left(\frac{1}{q_i}\right)^{\alpha_i}.$$

From the strict log-concavity of the $\log(\cdot)$ function and Jensen's inequality we have

$$\left(\sum_{i=1}^{\ell} \alpha_i q_i\right) \geq \prod_{i=1}^{\ell} q_i^{\alpha_i}.$$

The identity is obtained if and only if $q_i = q_j$, for all $i \neq j$. Thus, we get

$$\left(\sum_{i=1}^{\ell} \alpha_i q_i\right) \prod_{i=1}^{\ell} \left(\frac{1}{q_i}\right)^{\alpha_i} \geq 1,$$

which in its turn implies that the minimum of expected communication complexity is equal to $d \cdot \det(\boldsymbol{L})^{\frac{1}{d}}$. The equality is achieved when the probabilities are equal. This concludes the proof. ■

By utilizing the block diagonal structure, we are able to design special sketches that allow us to compress for free. This can be seen from row 12, where the communication complexity of using Bernoulli compressor with equal probabilities for det-CGD2 in expectation is the same with GD, but the number of bits sent per iteration is reduced.

D.3. General cases for det-CGD1

The first part (row 1 to row 8) of Table 1 records the communication complexities of det-CGD1 in the block diagonal setting and in the general setting. Depending on the types of sketches $\boldsymbol{S}_i^k$ and matrices $\boldsymbol{W}_i$ we are using, we can calculate the optimal scaling factor γ_i using Theorem 4. According to (5), in order to reach an error level of ε^2, we need

$$K \geq \frac{1}{\det(\boldsymbol{D})^{\frac{1}{d}}} \cdot \frac{2(f(x^0) - f^{\inf})}{\varepsilon^2}, \tag{24}$$

where K is the number of iterations in total. We can then obtain the communication complexity taking into account the number of bits transferred in each iteration in the block diagonal case. The same applies to the general case which can be viewed as a special case of the block diagonal setting where there is only 1 block.

D.4. General cases for det-CGD2

The second part of Table 1 (row 9 to row 12) records the communication complexities of det-CGD2. Unlike det-CGD1, we can always obtain the best stepsize matrix $\boldsymbol{D}$ here if the sketch $\boldsymbol{S}^k$ is given. The communication complexity can then be obtained in the same way as in the previous case using (24) combined with the number of bits sent per iteration.

D.5. Interpretations of Table 1

The communication complexity of the Gradient Descent algorithm (row 13) in the general non-convex setting is equal to $d\lambda_{\max}(\boldsymbol{L})$, where $\lambda_{\max}(\boldsymbol{L})$ serves as the smoothness constant of the function. Compared to the GD, det-CGD1 and det-CGD2 that use matrix stepsize without compression (row 1 and 9) are better in terms of both iteration and communication complexity. There are some results in the table that need careful analysis and we them present below. In the remainder of the section we will omit the constant multiplier $2(f(x^0) - f^{\inf})/\varepsilon^2$ in communication complexity, as it appears for every setting in the table and thus is redundant for comparison purposes.

D.5.1 Comparison of row 5 and 7

Here we show that the communication complexity given in row 5 is always worse than that of row 7. This can be seen from the following proposition.

Proposition 6 *For any matrix $\boldsymbol{L} \in \mathbb{S}^d_{++}$, the following inequality holds*

$$\lambda_{\max}\left(\boldsymbol{L}^{\frac{1}{2}} \operatorname{diag}(\boldsymbol{L}^{-1})\boldsymbol{L}^{\frac{1}{2}}\right) \cdot \det(\boldsymbol{L})^{\frac{1}{d}} \geq \det(\operatorname{diag}(\boldsymbol{L}))^{\frac{1}{d}}.$$

Proof The inequality given in Proposition 6 can be reformulated as

$$\lambda_{\max}(\boldsymbol{L}\,\mathrm{diag}(\boldsymbol{L}^{-1})) \geq \det(\boldsymbol{L}^{-1}\,\mathrm{diag}(\boldsymbol{L}))^{\frac{1}{d}}.$$

We use the notation

$$\boldsymbol{M}_1 = \boldsymbol{L}\,\mathrm{diag}(\boldsymbol{L}^{-1}), \quad \boldsymbol{M}_2 = \boldsymbol{L}^{-1}\,\mathrm{diag}(\boldsymbol{L}),$$

and notice that for any $i \in [d]$, we have

$$(\boldsymbol{M}_1)_{ii} = (\boldsymbol{L})_{ii} \cdot (\boldsymbol{L}^{-1})_{ii} = (\boldsymbol{M}_2)_{ii}.$$

Here the notation $(\boldsymbol{A})_{ij}$ refers to the entry (i,j) of matrix $\boldsymbol{A}$. As a result

$$\lambda_{\max}(\boldsymbol{M}_1) \quad \geq \quad \left(\prod_{i=1}^{d}(\boldsymbol{M}_1)_{ii}\right)^{\frac{1}{d}} = \left(\prod_{i=1}^{d}(\boldsymbol{M}_2)_{ii}\right)^{\frac{1}{d}} \geq \det(\boldsymbol{M}_2)^{\frac{1}{d}},$$

where the first inequality is due to the fact that each diagonal element is upper-bounded by the maximum eigenvalue value, while the second one is obtained using the fact that the product of the diagonal elements is an upper bound of the determinant. ■

From Proposition 6, it immediately follows that the result in row 7 is better than row 5 in terms of both communication and iteration complexity.

D.5.2 Comparison of row 6 and 7

In this section we bring an examples of matrices $\boldsymbol{L}$ which show that rows 6 and 7 are not comparable in general. Let $d = 2$ and $\boldsymbol{L} \in \mathbb{S}^2_{++}$. If we pick

$$\boldsymbol{L} = \begin{pmatrix} 16 & 0 \\ 0 & 1 \end{pmatrix},$$

then

$$\begin{aligned} \det(\mathrm{diag}(\boldsymbol{L}))^{\frac{1}{d}} &= 4; \\ \lambda_{\max}^{\frac{1}{2}}(\boldsymbol{L})\det(\boldsymbol{L})^{\frac{1}{2d}} &= 8. \end{aligned}$$

However, if we pick

$$\boldsymbol{L} = \begin{pmatrix} 16 & 3.9 \\ 3.9 & 1 \end{pmatrix},$$

then

$$\begin{aligned} \det(\mathrm{diag}(\boldsymbol{L}))^{\frac{1}{d}} &= 4; \\ \lambda_{\max}^{\frac{1}{2}}(\boldsymbol{L})\det(\boldsymbol{L})^{\frac{1}{2d}} &\simeq 3.88. \end{aligned}$$

From this example, we can see that the relation between the results in row 6 and 7 may vary depending on the value of $\boldsymbol{L}$.

Appendix E. Distributed Case

E.1. Distributed det-CGD1

In this section we describe the distributed versions of our algorithms and present convergence guarantees for them. Let us consider an objective function that is sum decomposable:

$$f(x) := \frac{1}{n}\sum_{i=1}^{n} f_i(x), \tag{25}$$

where each $f_i : \mathbb{R}^d \to \mathbb{R}$ is a differentiable function. We assume that f satisfies Assumption 1 and the component functions satisfy the below condition.

Assumption 4 *Each component function f_i is $\boldsymbol{L}_i$-smooth and is bounded from below: $f_i(x) \geq f_i^{\inf}$ for all $x \in \mathbb{R}^d$.*

This assumption also implies that f is of matrix smoothness with $\bar{\boldsymbol{L}} \in \mathbb{S}^d_{++}$, where $\bar{\boldsymbol{L}} = \frac{1}{n}\sum_{i=1}^n \boldsymbol{L}_i$. Following the standard FL framework [27, 28, 33], we assume that the i-th component function f_i is stored on the i-th client. At each iteration, the clients in parallel compute and compress the local gradient ∇f_i and communicate it to the central server. The server, then aggregates the compressed gradients, computes the next iterate, and in parallel broadcasts it to the clients. See the pseudo-codes below for the details.

Algorithm 1 Distributed det-CGD1

Input: Starting point x_0, stepsize matrix $\boldsymbol{D}$, number of iterations K
for $k = 0, 1, 2, \ldots, K-1$ **do**
  The devices in parallel:
    sample $\boldsymbol{S}_i^k \sim \mathcal{S}$;
    compute $\boldsymbol{S}_i^k \nabla f_i(x_k)$;
    broadcast $\boldsymbol{S}_i^k \nabla f_i(x_k)$.
  The server:
    combines $g_k = \frac{\boldsymbol{D}}{n}\sum_i^n \boldsymbol{S}_i^k \nabla f_i(x_k)$;
    computes $x_{k+1} = x_k - g_k$;
    broadcasts x_{k+1}.
end for
Return: x_K

Algorithm 2 Distributed det-CGD2

Input: Starting point x_0, stepsize matrix $\boldsymbol{D}$, number of iterations K
for $k = 0, 1, 2, \ldots, K-1$ **do**
  The devices in parallel:
    sample $\boldsymbol{T}_i^k \sim \mathcal{T}$;
    compute $\boldsymbol{T}_i^k \boldsymbol{D} \nabla f_i(x_k)$;
    broadcast $\boldsymbol{T}_i^k \boldsymbol{D} \nabla f_i(x_k)$.
  The server:
    combines $g_k = \frac{1}{n}\sum_i^n \boldsymbol{T}_i^k \boldsymbol{D} \nabla f_i(x_k)$;
    computes $x_{k+1} = x_k - g_k$;
    broadcasts x_{k+1}.
end for
Return: x_K

Theorem 7 *Let $f_i : \mathbb{R}^d \to \mathbb{R}$ satisfy Assumption 4 and let f satisfy Assumption 1 and Assumption 2 with smoothness matrix $\boldsymbol{L}$. If the stepsize satisfies*

$$\boldsymbol{D}\boldsymbol{L}\boldsymbol{D} \preceq \boldsymbol{D}, \tag{26}$$

then the following convergence bound is true for the iterates of Algorithm 1:

$$\frac{1}{K}\mathbb{E}\left[\left\|\nabla f(x^k)\right\|^2_{\boldsymbol{D}/\det(\boldsymbol{D})^{1/d}}\right] \leq \frac{2(1+\frac{\lambda_{\boldsymbol{D}}}{n})^K\left(f(x^0) - f^{\inf}\right)}{\det(\boldsymbol{D})^{1/d}\,K} + \frac{2\lambda_{\boldsymbol{D}}\Delta^{\inf}}{\det(\boldsymbol{D})^{1/d}\,n},$$

where $\Delta^{\inf} := f^{\inf} - \frac{1}{n}\sum_{i=1}^{n} f_i^{\inf}$ *and*

$$\lambda_{\boldsymbol{D}} := \max_i \left\{ \lambda_{\max}\left(\mathbb{E}\left[\boldsymbol{L}_i^{\frac{1}{2}} \left(\boldsymbol{S}_i^k - \boldsymbol{I}_d \right) \boldsymbol{D}\boldsymbol{L}\boldsymbol{D} \left(\boldsymbol{S}_i^k - \boldsymbol{I}_d \right) \boldsymbol{L}_i^{\frac{1}{2}} \right] \right) \right\}.$$

The same result is true for Algorithm 2 with a different constant $\lambda_{\boldsymbol{D}}$, see Appendix E.4 for the details. The analysis is largely inspired by Theorem 1 of [26]. Now, let us examine the right-hand side of the bound from Theorem 7. We start by observing that the first term has exponential dependence in K. However, the term inside the brackets, $1 + \lambda_{\boldsymbol{D}}/n$, depends on the stepsize $\boldsymbol{D}$. Furthermore, it has a second-order dependence on $\boldsymbol{D}$, implying that $\lambda_{\alpha\boldsymbol{D}} = \alpha^2\lambda_{\boldsymbol{D}}$, as opposed to $\det(\alpha\boldsymbol{D})^{1/d}$, which is linear in α. Therefore, we can choose a small enough coefficient α to ensure that $\lambda_{\boldsymbol{D}}$ is of order n/K. This means that for a fixed number of iterations K, we choose the matrix stepsize to be "small enough" to guarantee that the denominator of the first term is bounded. The following corollary summarizes these arguments.

Corollary 8 *We reach an error level of* ε^2 *in the bound from Theorem 7 if the following conditions are satisfied:*

$$\boldsymbol{D}\boldsymbol{L}\boldsymbol{D} \preceq \boldsymbol{D}, \quad \lambda_{\boldsymbol{D}} \leq \min\left\{ \frac{n}{K}, \frac{n\varepsilon^2}{4\Delta^{\inf}} \det(\boldsymbol{D})^{1/d} \right\}, K \geq \frac{12(f(x^0) - f^{\inf})}{\det(\boldsymbol{D})^{1/d}\,\varepsilon^2}. \tag{27}$$

Proposition 14 proves that these conditions with respect to $\boldsymbol{D}$ are convex. In order to minimize the iteration complexity for getting ε^2 error, one needs to solve the following optimization problem

$$\begin{array}{ll} \text{minimize} & \log\det(\boldsymbol{D}^{-1}) \\ \text{subject to} & \boldsymbol{D} \quad \text{satisfies} \quad (27). \end{array}$$

Choosing the optimal stepsize for Algorithm 1 is analogous to solving (8). One can formulate the distributed counterpart of Theorem 4 and attempt to solve it for different sketches. Furthermore, this leads to a convex matrix minimization problem involving $\boldsymbol{D}$. Similar to the single-node case, computational methods can be employed using the CVXPY package. However, some additional effort is required to transform (27) into the disciplined convex programming (DCP) format.

The second term in the bound from Theorem 7 corresponds to the convergence neighborhood of the algorithm. It does not depend on the number of iteration, thus it remains unchanged, after we choose the stepsize. Nevertheless, it depends on the number of clients n. In general, the term $\Delta^{\inf}/n$ can be unbounded, when $n \to +\infty$. However, per Corollary 8, we require $\lambda_{\boldsymbol{D}}$ to be upper-bounded by n/K. Thus, the neighborhood term will indeed converge to zero when $K \to +\infty$, if we choose the stepsize accordingly.

We compare our results with the existing results for DCGD. In particular we use the technique from [26] for the scalar smooth DCGD with scalar stepsizes. This means that the parameters of algorithms are $\boldsymbol{L}_i = L_i\boldsymbol{I}_d, \boldsymbol{L} = L\boldsymbol{I}_d, \boldsymbol{D} = \gamma\boldsymbol{I}_d, \omega = \lambda_{\max}\big(\mathbb{E}[(\boldsymbol{S}_i^k)^\top \boldsymbol{S}_i^k]\big) - 1$. One may check that (27) reduces to $\gamma \leq \min\left\{\frac{1}{L}, \sqrt{\frac{n}{KL_{\max}L\omega}}, \frac{n\varepsilon^2}{4\Delta^{\inf}L_{\max}L\omega}\right\}$ and $K\gamma \geq \frac{12(f(x^0)-f^{\inf})}{\varepsilon^2}$. As expected, this coincides with the results from [26]. Finally, we back up our theoretical findings with experiments which are deferred to Appendix G.

E.2. Proof of Theorem 7

We first present some simple technical lemmas whose proofs are deferred to Appendix F. Let us recall that $\boldsymbol{D} \in \mathbb{S}^d_{++}$ is the stepsize matrix, $\boldsymbol{L}, \boldsymbol{L}_i \in \mathbb{S}^d_{++}$ are the smoothness matrices for f and f_i, respectively.

Lemma 9 (Variance Decomposition) *For any random vector $x \in \mathbb{R}^d$, and any matrix $\boldsymbol{M} \in \mathbb{S}^d_{+}$, the following identity holds*

$$\mathbb{E}\left[\|x - \mathbb{E}[x]\|^2_{\boldsymbol{M}}\right] = \mathbb{E}\left[\|x\|^2_{\boldsymbol{M}}\right] - \|\mathbb{E}[x]\|^2_{\boldsymbol{M}}. \tag{28}$$

Lemma 10 *Assume $\{a_i\}_{i=1}^n$ is a set of independent random vectors in $\mathbb{R}^d$, which satisfy*

$$\mathbb{E}[a_i] = 0, \quad \forall i \in [n].$$

Then, for any $\boldsymbol{M} \in \mathbb{S}^d_{++}$, we have

$$\mathbb{E}\left[\left\|\frac{1}{n}\sum_{i=1}^n a_i\right\|^2_{\boldsymbol{M}}\right] = \frac{1}{n^2}\sum_{i=1}^n \mathbb{E}\left[\|a_i\|^2_{\boldsymbol{M}}\right]. \tag{29}$$

Lemma 11 *For any vector $x \in \mathbb{R}^d$, and sketch matrix $\boldsymbol{S} \in \mathbb{S}^d_{+}$ taken from some distribution $\mathcal{S}$ over $\boldsymbol{S}^d_{+}$, which satisfies*

$$\mathbb{E}[\boldsymbol{S}] = \boldsymbol{I}_d.$$

Then for any matrix $\boldsymbol{M} \in \boldsymbol{S}^d_{++}$, we have the following identity holds,

$$\mathbb{E}\left[\|\boldsymbol{S}x - x\|^2_{\boldsymbol{M}}\right] = \|x\|^2_{\mathbb{E}[\boldsymbol{S}\boldsymbol{M}\boldsymbol{S}]-\boldsymbol{M}}. \tag{30}$$

Lemma 12 *If we have a differentiable function $f : \mathbb{R}^d \to \mathbb{R}$, that is $\boldsymbol{L}$ matrix smooth and lower bounded by $f^{\inf}$, if we assume $\boldsymbol{L} \in \mathbb{S}^d_{++}$, then the following inequality holds*

$$\left\langle \nabla f(x), \boldsymbol{L}^{-1}\nabla f(x)\right\rangle \leq 2(f(x) - f^{\inf}). \tag{31}$$

Let the gradient estimator of our algorithm be defined as

$$g(x) := \frac{1}{n}\sum_{i=1}^n \boldsymbol{S}_i^k \nabla f_i(x), \tag{32}$$

as a result, det-CGD1 in the distributed case can then be written as

$$x^{k+1} = x^k - \boldsymbol{D}g(x^k).$$

Notice that we have

$$\mathbb{E}\left[g(x^k) \mid x^k\right] = \frac{1}{n}\sum_{i=1}^n \mathbb{E}\left[\boldsymbol{S}_i^k\right]\nabla f_i(x^k) = \nabla f(x^k). \tag{33}$$

We start with applying the $\boldsymbol{L}$-matrix smoothness of f:

$$\begin{aligned} f(x^{k+1}) &\leq f(x^k) + \left\langle \nabla f(x^k), x^{k+1} - x^k \right\rangle + \frac{1}{2}\left\langle \boldsymbol{L}(x^{k+1} - x^k), x^{k+1} - x^k \right\rangle \\ &= f(x^k) + \left\langle \nabla f(x^k), -\boldsymbol{D}g(x^k) \right\rangle + \frac{1}{2}\left\langle \boldsymbol{L}\left(-\boldsymbol{D}g(x^k)\right), -\boldsymbol{D}g(x^k) \right\rangle \\ &= f(x^k) - \left\langle \nabla f(x^k), \boldsymbol{D}g(x^k) \right\rangle + \frac{1}{2}\left\langle \boldsymbol{L}\boldsymbol{D}g(x^k), \boldsymbol{D}g(x^k) \right\rangle. \end{aligned}$$

Taking expectation conditioned on x^k, we get

$$\begin{aligned} \mathbb{E}\left[f(x^{k+1}) \mid x^k\right] &\leq f(x^k) - \left\langle \nabla f(x^k), \boldsymbol{D}\mathbb{E}\left[g(x^k) \mid x^k\right] \right\rangle + \frac{1}{2}\mathbb{E}\left[\left\langle \boldsymbol{L}\boldsymbol{D}g(x^k), \boldsymbol{D}g(x^k) \right\rangle \mid x^k\right] \\ &\overset{(33)}{=} f(x^k) - \left\langle \nabla f(x^k), \boldsymbol{D}\nabla f(x^k) \right\rangle + \frac{1}{2}\mathbb{E}\left[\left\langle \boldsymbol{L}\boldsymbol{D}g(x^k), \boldsymbol{D}g(x^k) \right\rangle \mid x^k\right] \\ &= f(x^k) - \left\|\nabla f(x^k)\right\|_{\boldsymbol{D}}^2 + \frac{1}{2}\underbrace{\mathbb{E}\left[\left\langle \boldsymbol{L}\boldsymbol{D}g(x^k), \boldsymbol{D}g(x^k) \right\rangle \mid x^k\right]}_{:=T}. \end{aligned} \tag{34}$$

Applying Lemma 9 to the term T we obtain

$$\begin{aligned} T &= \mathbb{E}\left[\left\|g(x^k)\right\|_{\boldsymbol{DLD}}^2 \mid x^k\right] \\ &\overset{(28)}{=} \mathbb{E}\left[\left\|g(x^k) - \mathbb{E}\left[g(x^k) \mid x^k\right]\right\|_{\boldsymbol{DLD}}^2 \mid x^k\right] + \left\|\mathbb{E}\left[g(x^k) \mid x^k\right]\right\|_{\boldsymbol{DLD}}^2. \end{aligned}$$

From the unbiasedness of the sketches, we have $\mathbb{E}\left[g(x^k) \mid x^k\right] = \nabla f(x^k)$, which yields

$$\begin{aligned} T &= \mathbb{E}\left[\left\|g(x^k) - \nabla f(x^k)\right\|_{\boldsymbol{DLD}}^2 \mid x^k\right] + \left\|\nabla f(x^k)\right\|_{\boldsymbol{DLD}}^2 \\ &= \mathbb{E}\left[\left\|\frac{1}{n}\sum_{i=1}^n \left(\boldsymbol{S}_i^k \nabla f_i(x^k) - \nabla f_i(x^k)\right)\right\|_{\boldsymbol{DLD}}^2 \mid x^k\right] + \left\|\nabla f(x^k)\right\|_{\boldsymbol{DLD}}^2. \end{aligned}$$

Using Lemma 10, we have

$$\begin{aligned} T &= \frac{1}{n^2}\sum_{i=1}^n \mathbb{E}\left[\left\|\boldsymbol{S}_i^k \nabla f_i(x^k) - \nabla f_i(x^k)\right\|_{\boldsymbol{DLD}}^2 \mid x^k\right] + \left\|\nabla f(x^k)\right\|_{\boldsymbol{DLD}}^2 \\ &\leq \frac{1}{n^2}\sum_{i=1}^n \mathbb{E}\left[\left\|\boldsymbol{S}_i^k \nabla f_i(x^k) - \nabla f_i(x^k)\right\|_{\boldsymbol{DLD}}^2 \mid x^k\right] + \left\|\nabla f(x^k)\right\|_{\boldsymbol{D}}^2, \end{aligned} \tag{35}$$

where the last inequality holds due to the inequality $\boldsymbol{DLD} \preceq \boldsymbol{D}$.

Lemma 13 *Let $\boldsymbol{S}$ be an unbiased ($\mathbb{E}\left[\boldsymbol{S}\right] = \boldsymbol{I}_d$) sketch drawn randomly from some distribution $\mathcal{S}$ over $\mathbb{S}_+^d$. The following bound holds for any $x \in \mathbb{R}^d$ and any matrix $\boldsymbol{A}$,*

$$\mathbb{E}\left[\left\|\boldsymbol{S}x - x\right\|_{\boldsymbol{DLD}}^2\right] \leq \lambda_{\max}\left(\boldsymbol{A}^{\frac{1}{2}}\mathbb{E}\left[\left(\boldsymbol{S} - \boldsymbol{I}_d\right)\boldsymbol{DLD}\left(\boldsymbol{S} - \boldsymbol{I}_d\right)\right]\boldsymbol{A}^{\frac{1}{2}}\right) \cdot \left\|x\right\|_{\boldsymbol{A}^{-1}}^2. \tag{36}$$

Plugging (35) into (34) and applying Lemmas 12 and 13 we deduce

$$\begin{aligned}
\mathbb{E}\left[f(x^{k+1}) \mid x^k\right] &\leq f(x^k) - \frac{1}{2}\left\|\nabla f(x^k)\right\|_{\boldsymbol{D}}^2 \\
&\quad + \frac{1}{2n^2}\sum_{i=1}^n \mathbb{E}\left[\left\|\boldsymbol{S}_i^k \nabla f_i(x^k) - \nabla f_i(x^k)\right\|_{\boldsymbol{DLD}}^2 \mid x^k\right]. \\
&\overset{(36)}{\leq} f(x^k) - \frac{1}{2}\left\|\nabla f(x^k)\right\|_{\boldsymbol{D}}^2 \\
&\quad + \frac{1}{2n^2}\sum_{i=1}^n \lambda_{\max}\left(\mathbb{E}\left[\boldsymbol{L}_i^{\frac{1}{2}}\left(\boldsymbol{S}_i^k - \boldsymbol{I}_d\right)\boldsymbol{DLD}\left(\boldsymbol{S}_i^k - \boldsymbol{I}_d\right)\boldsymbol{L}_i^{\frac{1}{2}}\right]\right)\left\|\nabla f_i(x^k)\right\|_{\boldsymbol{L}_i^{-1}}^2 \\
&\overset{(31)}{\leq} f(x^k) - \frac{1}{2}\left\|\nabla f(x^k)\right\|_{\boldsymbol{D}}^2 \\
&\quad + \frac{1}{n^2}\sum_{i=1}^n \lambda_{\max}\left(\mathbb{E}\left[\boldsymbol{L}_i^{\frac{1}{2}}\left(\boldsymbol{S}_i^k - \boldsymbol{I}_d\right)\boldsymbol{DLD}\left(\boldsymbol{S}_i^k - \boldsymbol{I}_d\right)\boldsymbol{L}_i^{\frac{1}{2}}\right]\right)\left(f_i(x^k) - f_i^{\inf}\right).
\end{aligned}$$

Recalling the definition of $\lambda_{\boldsymbol{D}}$, we bound $f(x^{k+1})$ by

$$\begin{aligned}
\mathbb{E}\left[f(x^{k+1}) \mid x^k\right] &\leq f(x^k) - \frac{1}{2}\left\|\nabla f(x^k)\right\|_{\boldsymbol{D}}^2 + \frac{1}{n^2}\sum_{i=1}^n \lambda_{\boldsymbol{D}}\left(f_i(x^k) - f_i^{\inf}\right) \\
&= f(x^k) - \frac{1}{2}\left\|\nabla f(x^k)\right\|_{\boldsymbol{D}}^2 + \frac{\lambda_{\boldsymbol{D}}}{n}\left(\frac{1}{n}\sum_{i=1}^n f_i(x^k) - \frac{1}{n}\sum_{i=1}^n f_i^{\inf}\right) \\
&= f(x^k) - \frac{1}{2}\left\|\nabla f(x^k)\right\|_{\boldsymbol{D}}^2 + \frac{\lambda_{\boldsymbol{D}}}{n}(f(x^k) - f^{\inf}) + \frac{\lambda_{\boldsymbol{D}}}{n}\left(f^{\inf} - \frac{1}{n}\sum_{i=1}^n f_i^{\inf}\right).
\end{aligned}$$

Subtracting $f^{\inf}$ from both sides, we get

$$\begin{aligned}
\mathbb{E}\left[f(x^{k+1}) - f^{\inf} \mid x^k\right] \leq f(x^k) - f^{\inf} - \frac{1}{2}\left\|\nabla f(x^k)\right\|_{\boldsymbol{D}}^2 + \frac{\lambda_{\boldsymbol{D}}}{n}(f(x^k) - f^{\inf}) \\
+ \frac{\lambda_{\boldsymbol{D}}}{n}\left(f^{\inf} - \frac{1}{n}\sum_{i=1}^n f_i^{\inf}\right).
\end{aligned}$$

Taking expectation, applying tower property and rearranging terms, we get

$$\begin{aligned}
\mathbb{E}\left[f(x^{k+1}) - f^{\inf}\right] \leq \left(1 + \frac{\lambda_{\boldsymbol{D}}}{n}\right)\mathbb{E}\left[f(x^k) - f^{\inf}\right] - \frac{1}{2}\mathbb{E}\left[\left\|\nabla f(x^k)\right\|_{\boldsymbol{D}}^2\right] \\
+ \frac{\lambda_{\boldsymbol{D}}}{n}\left(f^{\inf} - \frac{1}{n}\sum_{i=1}^n f_i^{\inf}\right). \qquad (37)
\end{aligned}$$

If we denote

$$\delta^k = \mathbb{E}\left[f(x^k) - f^{\inf}\right], \quad r^k = \mathbb{E}\left[\left\|\nabla f(x^k)\right\|_{\boldsymbol{D}}^2\right], \quad \Delta^{\inf} = f^{\inf} - \frac{1}{n}\sum_{i=1}^n f_i^{\inf},$$

then (37) becomes

$$\frac{1}{2}r^k \leq \left(1+\frac{\lambda_{\boldsymbol{D}}}{n}\right)\delta^k - \delta^{k+1} + \frac{\lambda_{\boldsymbol{D}}\Delta^{\inf}}{n}. \tag{38}$$

In order to approach the final result, we now follow [43], [26] and define an exponentially decaying weighting sequence $\{w_k\}_{k=-1}^{K}$, where K is the total number of iterations. We fix $w_{-1} > 0$ and define

$$w_k = \frac{w_{k-1}}{1+\lambda_{\boldsymbol{D}}/n}, \qquad \text{for all} \quad k \geq 0.$$

By multiplying both sides of the recursion (38) by w_k, we get

$$\frac{1}{2}w_k r^k \leq w_{k-1}\delta^k - w_k\delta^{k+1} + \frac{\lambda_{\boldsymbol{D}}\Delta^{\inf}}{n}w_k.$$

Summing up the inequalities from $k = 0, ..., K-1$, we get

$$\frac{1}{2}\sum_{k=0}^{K-1} w_k r^k \leq w_{-1}\delta^0 - w_{K-1}\delta^K + \frac{\lambda_{\boldsymbol{D}}\Delta^{\inf}}{n}\sum_{k=0}^{K-1} w^k.$$

Define $W_K = \sum_{k=0}^{K-1} w_k$, and divide both sides by W_K, we get

$$\frac{1}{2}\min_{0\leq k\leq K-1} r^k \leq \frac{1}{2}\frac{\sum_{k=0}^{K-1} w_k r^k}{W_K} r^k \leq \frac{w_{-1}}{W_K}\delta^0 + \frac{\lambda_{\boldsymbol{D}}\Delta^{\inf}}{n}.$$

Notice that from the definition of w_k, we know that the following inequality holds,

$$\frac{w_{-1}}{W_K} \leq \frac{w_{-1}}{K w_{K-1}} = \frac{(1+\frac{\lambda_{\boldsymbol{D}}}{n})^K}{K}.$$

As a result, we have

$$\min_{0\leq k\leq K-1} r^k \leq \frac{2\left(1+\frac{\lambda_{\boldsymbol{D}}}{n}\right)^K}{K}\delta^0 + \frac{2\lambda_{\boldsymbol{D}}\Delta^{\inf}}{n}.$$

Recalling the definition for r^k and δ^k, we get the following result,

$$\min_{0\leq k\leq K-1} \mathbb{E}\left[\left\|\nabla f(x^k)\right\|_{\boldsymbol{D}}^2\right] \leq \frac{2(1+\frac{\lambda_{\boldsymbol{D}}}{n})^K\left(f(x^0)-f^{\inf}\right)}{K} + \frac{2\lambda_{\boldsymbol{D}}\Delta^{\inf}}{n}.$$

Finally, we apply determinant normalization and get

$$\min_{0\leq k\leq K-1} \mathbb{E}\left[\left\|\nabla f(x^k)\right\|_{\boldsymbol{D}/\det(\boldsymbol{D})^{1/d}}^2\right] \leq \frac{2(1+\frac{\lambda_{\boldsymbol{D}}}{n})^K\left(f(x^0)-f^{\inf}\right)}{\det(\boldsymbol{D})^{1/d}K} + \frac{2\lambda_{\boldsymbol{D}}\Delta^{\inf}}{\det(\boldsymbol{D})^{1/d}n}. \tag{39}$$

This concludes the proof.

E.3. Convexity of the constraints

Proposition 14 *The set of matrices $\boldsymbol{D}$ that satisfy (27) is convex.*

Proof The first inequality in (27) can be reformulated into

$$\boldsymbol{D} \preceq \boldsymbol{L}^{-1},$$

which is linear in $\boldsymbol{D}$ and, therefore, is convex. For the second constraint in (27),

$$\max_i \left\{ \lambda_{\max} \left(\mathbb{E} \left[\boldsymbol{L}_i^{\frac{1}{2}} \left(\boldsymbol{S}_i^k - \boldsymbol{I}_d \right) \boldsymbol{D}\boldsymbol{L}\boldsymbol{D} \left(\boldsymbol{S}_i^k - \boldsymbol{I}_d \right) \boldsymbol{L}_i^{\frac{1}{2}} \right] \right) \right\} \leq \frac{n}{K}, \tag{40}$$

we can reformulate it into n constraints, one for each client i:

$$\begin{aligned}
&\lambda_{\max} \left(\mathbb{E} \left[\boldsymbol{L}_i^{\frac{1}{2}} \left(\boldsymbol{S}_i^k - \boldsymbol{I}_d \right) \boldsymbol{D}\boldsymbol{L}\boldsymbol{D} \left(\boldsymbol{S}_i^k - \boldsymbol{I}_d \right) \boldsymbol{L}_i^{\frac{1}{2}} \right] \right) \leq \frac{n}{K}, \quad \forall i \in [\ell] \\
&\Leftrightarrow \mathbb{E} \left[\boldsymbol{L}_i^{\frac{1}{2}} \left(\boldsymbol{S}_i^k - \boldsymbol{I}_d \right) \boldsymbol{D}\boldsymbol{L}\boldsymbol{D} \left(\boldsymbol{S}_i^k - \boldsymbol{I}_d \right) \boldsymbol{L}_i^{\frac{1}{2}} \right] \preceq \frac{n}{K} \boldsymbol{I}_d, \quad \forall i \in [\ell] \\
&\Leftrightarrow \boldsymbol{L}_i^{\frac{1}{2}} \mathbb{E} \left[\left(\boldsymbol{S}_i^k - \boldsymbol{I}_d \right) \boldsymbol{D}\boldsymbol{L}\boldsymbol{D} \left(\boldsymbol{S}_i^k - \boldsymbol{I}_d \right) \right] \boldsymbol{L}_i^{\frac{1}{2}} \preceq \frac{n}{K} \boldsymbol{I}_d, \quad \forall i \in [\ell] \\
&\Leftrightarrow \mathbb{E} \left[\left(\boldsymbol{S}_i^k - \boldsymbol{I}_d \right) \boldsymbol{D}\boldsymbol{L}\boldsymbol{D} \left(\boldsymbol{S}_i^k - \boldsymbol{I}_d \right) \right] \preceq \frac{n}{K} \boldsymbol{L}_i^{-1}, \quad \forall i \in [\ell].
\end{aligned}$$

We then look at the individual condition for one client i,

$$\mathbb{E} \left[\left(\boldsymbol{S}_i^k - \boldsymbol{I}_d \right) \boldsymbol{D}\boldsymbol{L}\boldsymbol{D} \left(\boldsymbol{S}_i^k - \boldsymbol{I}_d \right) \right] \preceq \frac{n}{K} \boldsymbol{L}_i^{-1}, \tag{41}$$

that is for any vector $u \in \mathbb{R}^d$, we require

$$\begin{aligned}
&u^\top \mathbb{E} \left[\left(\boldsymbol{S}_i^k - \boldsymbol{I}_d \right) \boldsymbol{D}\boldsymbol{L}\boldsymbol{D} \left(\boldsymbol{S}_i^k - \boldsymbol{I}_d \right) \right] u \leq \frac{n}{K} u^\top \boldsymbol{L}_i^{-1} u, \\
&\quad \Leftrightarrow \operatorname{tr} \left(u^\top \mathbb{E} \left[\left(\boldsymbol{S}_i^k - \boldsymbol{I}_d \right) \boldsymbol{D}\boldsymbol{L}\boldsymbol{D} \left(\boldsymbol{S}_i^k - \boldsymbol{I}_d \right) \right] u \right) \leq \frac{n}{K} \operatorname{tr}(u^\top \boldsymbol{L}_i^{-1} u), \\
&\quad \Leftrightarrow \mathbb{E} \left[\operatorname{tr}(u^\top \left(\boldsymbol{S}_i^k - \boldsymbol{I}_d \right) \boldsymbol{D}\boldsymbol{L}\boldsymbol{D} \left(\boldsymbol{S}_i^k - \boldsymbol{I}_d \right) u) \right] \leq \operatorname{tr}(u^\top \boldsymbol{L}_i^{-1} u), \\
&\quad \Leftrightarrow \operatorname{tr}(\boldsymbol{L}^{\frac{1}{2}} \boldsymbol{D} \mathbb{E} \left[\left(\boldsymbol{S}_i^k - \boldsymbol{I}_d \right) u u^\top \left(\boldsymbol{S}_i^k - \boldsymbol{I}_d \right) \right] \boldsymbol{D} \boldsymbol{L}^{\frac{1}{2}}) \leq \operatorname{tr}(u^\top \boldsymbol{L}_i^{-1} u).
\end{aligned}$$

We now define function $g_u : \mathbb{S}^d_{++} \to \mathbb{R}$ for every fixed $u \neq 0$,

$$g_u(\boldsymbol{D}) = \operatorname{tr}(\boldsymbol{L}^{\frac{1}{2}} \boldsymbol{D} \mathbb{E} \left[\left(\boldsymbol{S}_i^k - \boldsymbol{I}_d \right) u u^\top \left(\boldsymbol{S}_i^k - \boldsymbol{I}_d \right) \right] \boldsymbol{D} \boldsymbol{L}^{\frac{1}{2}}), \tag{42}$$

notice that $uu^\top$ is a rank-1 matrix that is positive semi-definite, so for every $y \in \mathbb{R}^d$,

$$\left(\left(\boldsymbol{S}_i^k - \boldsymbol{I}_d \right) y \right)^\top u u^\top \left(\left(\boldsymbol{S}_i^k - \boldsymbol{I}_d \right) y \right) \geq 0,$$

which means that $\left(\boldsymbol{S}_i^k - \boldsymbol{I}_d\right) uu^\top \left(\boldsymbol{S}_i^k - \boldsymbol{I}_d\right) \in \mathbb{S}_+^d$, and thus $\boldsymbol{R} := \mathbb{E}\left[\left(\boldsymbol{S}_i^k - \boldsymbol{I}_d\right) uu^\top \left(\boldsymbol{S}_i^k - \boldsymbol{I}_d\right)\right] \in \mathbb{S}_+^d$ as well. Using Lemma 3, we know that $g_u(\boldsymbol{D})$ is a convex function for every $0 \neq u \in \mathbb{R}^d$, thus its sub-level set $\{\boldsymbol{D} \in \mathbb{S}_{++}^d \mid g_u(\boldsymbol{D}) \leq \operatorname{tr}(u^\top \boldsymbol{L}_i^{-1} u)\}$ is a convex set. The intersection of those convex sets corresponding to the individual constraint (41) of client i is convex. Again the intersection of those convex sets for each client i, which corresponds to (40), is still convex.

For the third constraint in (27), we can transform it using similar steps as we obtain (40) into

$$\mathbb{E}\left[\left(\boldsymbol{S}_i^k - \boldsymbol{I}_d\right) \boldsymbol{D}\boldsymbol{L}\boldsymbol{D} \left(\boldsymbol{S}_i^k - \boldsymbol{I}_d\right)\right] \preceq \frac{n\varepsilon^2}{4\Delta^{\inf}} \det(\boldsymbol{D})^{1/d} \boldsymbol{L}_i^{-1}, \quad \forall i. \tag{43}$$

If we look at each individual constraint, we can write in quadratic forms for any $0 \neq u \in \mathbb{R}^d$,

$$u^\top \mathbb{E}\left[\left(\boldsymbol{S}_i^k - \boldsymbol{I}_d\right) \boldsymbol{D}\boldsymbol{L}\boldsymbol{D} \left(\boldsymbol{S}_i^k - \boldsymbol{I}_d\right)\right] u \leq \frac{n\varepsilon^2}{4\Delta^{\inf}} \det(\boldsymbol{D})^{1/d} \cdot u^\top \boldsymbol{L}_i^{-1} u, \quad \forall u \neq 0.$$

Using the linearity of expectation and the trace operator with the trace trick, we can transform the above condition into,

$$\operatorname{tr}(\boldsymbol{L}^{\frac{1}{2}} \boldsymbol{D} \mathbb{E}\left[\left(\boldsymbol{S}_i^k - \boldsymbol{I}_d\right) uu^\top \left(\boldsymbol{S}_i^k - \boldsymbol{I}_d\right)\right] \boldsymbol{D}\boldsymbol{L}^{\frac{1}{2}}) \leq \frac{n\varepsilon^2}{4\Delta^{\inf}} \det(\boldsymbol{D})^{\frac{1}{d}} \operatorname{tr}(u^\top \boldsymbol{L}_i^{-1} u) \quad \forall u \neq 0.$$

notice that we have already shown that $\boldsymbol{R} = \mathbb{E}\left[\left(\boldsymbol{S}_i^k - \boldsymbol{I}_d\right) uu^\top \left(\boldsymbol{S}_i^k - \boldsymbol{I}_d\right)\right] \in \mathbb{S}_+^d$. Thus if we apply Lemma 3, we know that the left-hand side of the previous inequality is convex w.r.t. $\boldsymbol{D}$. On the other hand we know that $\det(\boldsymbol{D})^{\frac{1}{d}}$ is a concave function for symmetric positive definite matrices $\boldsymbol{D}$. So the set of $\boldsymbol{D}$ satisfying the constraint here for every $u \in \mathbb{R}^d$ is convex, thus their intersection is convex as well. Which means that the set of $\boldsymbol{D}$ satisfying the constraint for each client i is convex. Thus the intersection of those convex sets corresponding to different clients, which corresponds to (43), is still convex. Now we know that the set of $\boldsymbol{D}$ satisfying each of the three constraints in (27) is convex, thus the intersection of them is convex as well. This concludes the proof. ■

E.3.1 Proof of Corollary 8

For the first term in the RHS of the convergence bound from Theorem 7 under condition (27), we know that

$$2\left(1 + \frac{\lambda_{\boldsymbol{D}}}{n}\right)^K \leq 2 \cdot \exp\left(\lambda_{\boldsymbol{D}} \cdot \frac{K}{n}\right) \leq 2 \cdot \exp(1) \leq 6,$$

thus

$$\begin{aligned}
\frac{2(1 + \frac{\lambda_{\boldsymbol{D}}}{n})^K \left(f(x^0) - f^{\inf}\right)}{\det(\boldsymbol{D})^{1/d} K} &\leq \frac{6\left(f(x^0) - f^{\inf}\right)}{\det(\boldsymbol{D})^{1/d} K} \\
&\leq \frac{6\left(f(x^0) - f^{\inf}\right)}{\det(\boldsymbol{D})^{1/d}} \cdot \frac{\varepsilon^2 \det(\boldsymbol{D})^{\frac{1}{d}}}{12\left(f(x^0) - f^{\inf}\right)} \\
&= \frac{\varepsilon^2}{2}.
\end{aligned}$$

While for the second term of RHS in the bound from Theorem 7, we have

$$\frac{2\lambda_{\boldsymbol{D}}\Delta^{\inf}}{\det(\boldsymbol{D})^{1/d}\,n} \leq \frac{2\Delta^{\inf}}{\det(\boldsymbol{D})^{1/d}\,n} \cdot \frac{\varepsilon^2(\det(\boldsymbol{D}))^{1/d}\,n}{4\Delta^{\inf}} \leq \frac{\varepsilon^2}{2}.$$

Thus we know that the left hand side of the bound from Theorem 7 is upper bounded by

$$\min_{0\leq k\leq K-1} \mathbb{E}\left[\left\|\nabla f(x^k)\right\|^2_{\frac{\boldsymbol{D}}{\det(\boldsymbol{D})^{1/d}}}\right] \quad \leq \quad \frac{\varepsilon^2}{2} + \frac{\varepsilon^2}{2} = \varepsilon^2.$$

This concludes the proof.

E.4. Distributed det-CGD2

We also extend det-CGD2 to the distributed case. Consider the method

$$x^{k+1} = x^k - \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{T}_i^k \boldsymbol{D}\nabla f_i(x^k), \tag{44}$$

where $\boldsymbol{D} \in \mathbb{S}^d_{++}$ is the stepsize matrix, and each $\boldsymbol{T}_i^k$ is a sequence of sketch matrices drawn randomly from some distribution $\mathcal{T}$ over $\mathbb{S}^d_+$ independent of each other, satisfying

$$\mathbb{E}\left[\boldsymbol{T}_i^k\right] = \boldsymbol{I}_d. \tag{45}$$

E.4.1 Analysis of distributed det-CGD2

In this section, we present the theory for Algorithm 2, which is an analogous to what we have seen for Algorithm 1. We first present the following lemma which is necessary for our analysis.

Lemma 15 *For any sketch $\boldsymbol{T}_i^k$ of client i drawn randomly from some distribution $\mathcal{T}$ over $\mathbb{S}^d_+$ which satisfies*

$$\mathbb{E}\left[\boldsymbol{T}_i^k\right] = \boldsymbol{I}_d,$$

the following inequality holds for any $x \in \mathbb{R}^d$ for each client i,

$$\mathbb{E}\left[\left\|\boldsymbol{T}_i^k\boldsymbol{D}x - \boldsymbol{D}x\right\|^2_{\boldsymbol{L}}\right] \leq \lambda_{\max}\left(\boldsymbol{L}_i^{\frac{1}{2}}\boldsymbol{D}\mathbb{E}\left[\left(\boldsymbol{T}_i^k - \boldsymbol{I}_d\right)\boldsymbol{L}\left(\boldsymbol{T}_i^k - \boldsymbol{I}_d\right)\right]\boldsymbol{D}\boldsymbol{L}_i^{\frac{1}{2}}\right) \cdot \|x\|^2_{\boldsymbol{L}_i^{-1}}. \tag{46}$$

Theorem 16 *Let $f_i : \mathbb{R}^d \to \mathbb{R}$ satisfy Assumption 4 and let f satisfy Assumptions 1 and 2 with a smoothness matrix $\boldsymbol{L}$. If the stepsize satisfies,*

$$\boldsymbol{D}\boldsymbol{L}\boldsymbol{D} \preceq \boldsymbol{D}, \tag{47}$$

then the following convergence bound is true for the iteration of Algorithm 2

$$\min_{0\leq k\leq K-1} \mathbb{E}\left[\left\|\nabla f(x^k)\right\|^2_{\frac{\boldsymbol{D}}{\det(\boldsymbol{D})^{1/d}}}\right] \leq \frac{2(1+\frac{\lambda'_{\boldsymbol{D}}}{n})^K\left(f(x^0) - f^{\inf}\right)}{\det(\boldsymbol{D})^{1/d}\,K} + \frac{2\lambda'_{\boldsymbol{D}}\Delta^{\inf}}{\det(\boldsymbol{D})^{1/d}\,n}, \tag{48}$$

where $\Delta^{\inf} := f^{\inf} - \frac{1}{n}\sum_{i=1}^n f_i^{\inf}$ and

$$\lambda'_{\boldsymbol{D}} \quad := \quad \max_i\left\{\lambda_{\max}\left(\mathbb{E}\left[\boldsymbol{L}_i^{\frac{1}{2}}\boldsymbol{D}\left(\boldsymbol{T}_i^k - \boldsymbol{I}_d\right)\boldsymbol{L}\left(\boldsymbol{T}_i^k - \boldsymbol{I}_d\right)\boldsymbol{D}\boldsymbol{L}_i^{\frac{1}{2}}\right]\right)\right\}.$$

Proof We first define function $g(x)$ as follows,

$$g(x) = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{T}_i^k \boldsymbol{D}\nabla f_i(x^k).$$

As a result, Algorithm 2 can be written as

$$x^{k+1} = x^k - g(x^k).$$

Notice that

$$\mathbb{E}\left[g(x)\right] = \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}\left[\boldsymbol{T}_i^k\right] \boldsymbol{D}\nabla f_i(x) = \boldsymbol{D}\nabla f(x). \tag{49}$$

We then start with the $\boldsymbol{L}$ matrix smoothness of function f,

$$\begin{aligned} f(x^{k+1}) &\leq f(x^k) + \left\langle \nabla f(x^k), x^{k+1} - x^k \right\rangle + \frac{1}{2}\left\langle \boldsymbol{L}(x^{k+1} - x^k), x^{k+1} - x^k \right\rangle \\ &= f(x^k) + \left\langle \nabla f(x^k), -g(x^k) \right\rangle + \frac{1}{2}\left\langle \boldsymbol{L}\left(-g(x^k)\right), -g(x^k) \right\rangle \\ &= f(x^k) - \left\langle \nabla f(x^k), g(x^k) \right\rangle + \frac{1}{2}\left\langle \boldsymbol{L}g(x^k), g(x^k) \right\rangle. \end{aligned}$$

We then take expectation conditioned on x^k,

$$\begin{aligned} \mathbb{E}\left[f(x^{k+1}) \mid x^k\right] &\leq f(x^k) - \left\langle \nabla f(x^k), \mathbb{E}\left[g(x^k) \mid x^k\right] \right\rangle + \frac{1}{2}\mathbb{E}\left[\left\langle \boldsymbol{L}g(x^k), g(x^k) \right\rangle \mid x^k\right] \\ &= f(x^k) - \left\langle \nabla f(x^k), \boldsymbol{D}\nabla f(x^k) \right\rangle + \frac{1}{2}\underbrace{\mathbb{E}\left[\left\langle \boldsymbol{L}g(x^k), g(x^k) \right\rangle \mid x^k\right]}_{:=T}. \end{aligned} \tag{50}$$

Lemma 9 yields

$$\begin{aligned} T &= \mathbb{E}\left[\left\|g(x^k)\right\|_{\boldsymbol{L}}^2 \mid x^k\right] \\ &\overset{(28)}{=} \mathbb{E}\left[\left\|g(x^k) - \mathbb{E}\left[g(x^k) \mid x^k\right]\right\|_{\boldsymbol{L}}^2 \mid x^k\right] + \left\|\mathbb{E}\left[g(x) \mid x^k\right]\right\|_{\boldsymbol{L}}^2. \end{aligned}$$

From (49) we deduce

$$\begin{aligned} T &= \mathbb{E}\left[\left\|g(x^k) - \boldsymbol{D}\nabla f(x^k)\right\|_{\boldsymbol{L}}^2 \mid x^k\right] + \left\|\boldsymbol{D}\nabla f(x^k)\right\|_{\boldsymbol{L}}^2 \\ &= \mathbb{E}\left[\left\|\frac{1}{n}\sum_{i=1}^{n} \boldsymbol{T}_i^k \boldsymbol{D}\nabla f_i(x^k) - \frac{\boldsymbol{D}}{n}\sum_{i=1}^{n}\nabla f_i(x^k)\right\|_{\boldsymbol{L}}^2 \mid x^k\right] + \left\|\nabla f(x^k)\right\|_{\boldsymbol{DLD}}^2 \\ &= \mathbb{E}\left[\left\|\frac{1}{n}\sum_{i=1}\left(\boldsymbol{T}_i^k\boldsymbol{D} - \boldsymbol{D}\right)\nabla f_i(x^k)\right\|_{\boldsymbol{L}}^2 \mid x^k\right] + \left\|\nabla f(x^k)\right\|_{\boldsymbol{DLD}}^2. \end{aligned}$$

Recalling Lemma 10 we obtain

$$
\begin{aligned}
T &\overset{(29)}{=} \frac{1}{n^2}\sum_{i=1}^n \mathbb{E}\left[\left\|\boldsymbol{T}_i^k \boldsymbol{D}\nabla f_i(x^k) - \boldsymbol{D}\nabla f_i(x^k)\right\|_{\boldsymbol{L}}^2 \mid x^k\right] + \left\|\nabla f(x^k)\right\|_{\boldsymbol{DLD}}^2 \\
&\overset{(47)}{\leq} \frac{1}{n^2}\sum_{i=1}^n \mathbb{E}\left[\left\|\boldsymbol{T}_i^k \boldsymbol{D}\nabla f_i(x^k) - \boldsymbol{D}\nabla f_i(x^k)\right\|_{\boldsymbol{L}}^2 \mid x^k\right] + \left\|\nabla f(x^k)\right\|_{\boldsymbol{D}}^2 .
\end{aligned}
$$

By applying Lemma 15, we get

$$
\begin{aligned}
T &\leq \frac{1}{n^2}\sum_{i=1}^n \lambda_{\max}\left(\boldsymbol{L}_i^{\frac{1}{2}}\boldsymbol{D}\mathbb{E}\left[(\boldsymbol{T}_i^k - \boldsymbol{I}_d)\boldsymbol{L}(\boldsymbol{T}_i^k - \boldsymbol{I}_d)\right]\boldsymbol{D}\boldsymbol{L}_i^{\frac{1}{2}}\right)\left\|\nabla f_i(x^k)\right\|_{\boldsymbol{L}_i^{-1}}^2 + \left\|\nabla f(x^k)\right\|_{\boldsymbol{D}}^2 \\
&\overset{(31)}{\leq} \lambda'_{\boldsymbol{D}} \cdot \frac{2}{n}\left(f(x^k) - \frac{1}{n}\sum_{i=1}^n f_i^{\inf}\right) + \left\|\nabla f(x^k)\right\|_{\boldsymbol{D}}^2 .
\end{aligned}
$$

Then we plug the upper bound of T back into (50), we get

$$
\begin{aligned}
&\mathbb{E}\left[f(x^{k+1}) \mid x^k\right] \\
&\quad \leq f(x^k) - \frac{1}{2}\left\|\nabla f(x^k)\right\|_{\boldsymbol{D}}^2 + \frac{\lambda'_{\boldsymbol{D}}}{n}\left(f(x^k) - f^{\inf}\right) + \frac{\lambda'_{\boldsymbol{D}}}{n}(f^{\inf} - \frac{1}{n}\sum_{i=1}^n f_i^{\inf}).
\end{aligned}
$$

Taking expectation, subtracting $f^{\inf}$ from both sides, and using tower property, we get

$$
\begin{aligned}
&\mathbb{E}\left[f(x^{k+1}) - f^{\inf}\right] \\
&\quad \leq \mathbb{E}\left[f(x^k) - f^{\inf}\right] - \frac{1}{2}\mathbb{E}\left[\left\|\nabla f(x^k)\right\|_{\boldsymbol{D}}^2\right] + \frac{\lambda'_{\boldsymbol{D}}}{n}\mathbb{E}\left[f(x^k) - f^{\inf}\right] + \frac{\lambda'_{\boldsymbol{D}}}{n}\Delta^{\inf}.
\end{aligned}
$$

Then following similar steps as in the proof of Theorem 7, we are able to get

$$
\min_{0\leq k\leq K-1} \mathbb{E}\left[\left\|\nabla f(x^k)\right\|_{\frac{\boldsymbol{D}}{\det(\boldsymbol{D})^{1/d}}}^2\right] \leq \frac{2(1+\frac{\lambda'_{\boldsymbol{D}}}{n})^K\left(f(x^0) - f^{\inf}\right)}{\det(\boldsymbol{D})^{1/d}\,K} + \frac{2\lambda'_{\boldsymbol{D}}\Delta^{\inf}}{\det(\boldsymbol{D})^{1/d}\,n}.
$$

This concludes the proof. ■

Similar to Algorithm 1, we can choose the parameters of the algorithm to avoid the exponential blow-up in convergence bound (48). The following corollary sums up the convergence conditions for Algorithm 2.

Corollary 17 *We reach an error level of ε^2 in (48) if the following conditions are satisfied:*

$$
\boldsymbol{DLD} \preceq \boldsymbol{D}, \quad \lambda'_{\boldsymbol{D}} \leq \min\left\{\frac{n}{K}, \frac{n\varepsilon^2}{4\Delta^{\inf}}\det(\boldsymbol{D})^{1/d}\right\}, \quad K \geq \frac{12(f(x^0) - f^{\inf})}{\det(\boldsymbol{D})^{1/d}\varepsilon^2}. \tag{51}
$$

The proof of this corollary is exactly the same as for Corollary 8.

E.4.2 Optimal stepsize

In order to minimize the iteration complexity for Algorithm 2, the following optimization problem needs to be solved

$$\begin{aligned} \min \quad & \log\det(\boldsymbol{D}^{-1}) \\ \text{subject to} \quad & \boldsymbol{D} \quad \text{satisfies} \quad (51) \end{aligned}$$

Following similar techniques in the proof of Proposition 14, we are able to prove that the above optimization problem is still a convex optimization problem. One simple way to find stepsize matrices is to follow the scheme suggested for solving (8). That is we first fix $\boldsymbol{W} \in \mathbb{S}^d_{++}$ and we find the optimal $0 < \gamma \in \mathbb{R}$, such that $\boldsymbol{D} = \gamma \boldsymbol{W}$ satisfies (51).

E.5. DCGD with constant stepsize

In this section we describe the convergence result for DCGD from [26]. We assume that the component functions f_i satisfy Assumption 4 with $\boldsymbol{L}_i = L_i \boldsymbol{I}_d$ and f satisfies Assumption 1 and 2 with $\boldsymbol{L} = L\boldsymbol{I}_d$. [26] proposed a unified analysis for non-convex optimization algorithms based on a generic upper bound on the second moment of the gradient estimator $g(x^k)$:

$$\mathbb{E}\left[\left\|g(x^k)\right\|^2\right] \leq 2A\left(f(x^k) - f^{\inf}\right) + B\left\|\nabla f(x^k)\right\|^2 + C, \tag{52}$$

In our case the gradient estimator is defined as follows

$$g_{\mathsf{DCGD}}(x^k) = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{S}_i^k \nabla f_i(x^k). \tag{53}$$

Here each $\boldsymbol{S}_i^k$ is the sketch matrix on the i-th client at the k-th iteration. One may check that g_{DCGD} satisfies (52) with the following constants:

$$A = \frac{\omega L_{\max}}{n}, \quad B = 1, \quad C = \frac{2\omega L_{\max}}{n}\Delta^{\inf}. \tag{54}$$

The constant $L_{\max}$ is defined as the maximum of all L_i and $\omega = \lambda_{\max}\left(\mathbb{E}\left[\left(\boldsymbol{S}_i^k\right)^\top \boldsymbol{S}_i^k\right]\right) - 1$. Applying Corollary 1 from [26], we deduce the following. If

$$\gamma \leq \min\left\{\frac{1}{L}, \frac{\sqrt{n}}{\sqrt{\omega L L_{\max} K}}, \frac{n\varepsilon^2}{4LL_{\max}\omega\Delta^{\inf}}\right\} \text{ and } \gamma K \geq \frac{12\left(f(x^0) - f^{\inf}\right)}{\varepsilon^2}, \tag{55}$$

then

$$\min_{k=0,\ldots,K-1} \mathbb{E}\left[\left\|\nabla f(x^k)\right\|^2\right] \leq \varepsilon^2. \tag{56}$$

Appendix F. Proofs of technical lemmas

F.1. Proof of Lemma 3

Let us pick any two matrices $\boldsymbol{D}_1, \boldsymbol{D}_2 \in \mathbb{S}^d_{++}$, scalar α satisfying $0 \leq \alpha \leq 1$ and show that the following inequality holds regardless of the choice of $\boldsymbol{R}$,

$$f(\alpha \boldsymbol{D}_1 + (1-\alpha)\boldsymbol{D}_2) \leq \alpha f(\boldsymbol{D}_1) + (1-\alpha) f(\boldsymbol{D}_2). \tag{57}$$

For the LHS, we have

$$\begin{aligned}
&f(\alpha \boldsymbol{D}_1 + (1-\alpha)\boldsymbol{D}_2) \\
&\quad = \operatorname{tr}(\boldsymbol{L}^{\frac{1}{2}} \left(\alpha \boldsymbol{D}_1 + (1-\alpha)\boldsymbol{D}_2\right) \boldsymbol{R} (\alpha \boldsymbol{D}_1 + (1-\alpha)\boldsymbol{D}_2) \boldsymbol{L}^{\frac{1}{2}}) \\
&\quad = \alpha^2 \operatorname{tr}(\boldsymbol{L}^{\frac{1}{2}} \boldsymbol{D}_1 \boldsymbol{R} \boldsymbol{D}_1 \boldsymbol{L}^{\frac{1}{2}}) + (1-\alpha)^2 \operatorname{tr}(\boldsymbol{L}^{\frac{1}{2}} \boldsymbol{D}_2 \boldsymbol{R} \boldsymbol{D}_2 \boldsymbol{L}^{\frac{1}{2}}) \\
&\qquad + \alpha(1-\alpha) \operatorname{tr}(\boldsymbol{L}^{\frac{1}{2}} \boldsymbol{D}_1 \boldsymbol{R} \boldsymbol{D}_2 \boldsymbol{L}^{\frac{1}{2}}) + \alpha(1-\alpha) \operatorname{tr}(\boldsymbol{L}^{\frac{1}{2}} \boldsymbol{D}_2 \boldsymbol{R} \boldsymbol{D}_1 \boldsymbol{L}^{\frac{1}{2}}).
\end{aligned}$$

and for the RHS, we have

$$\alpha f(\boldsymbol{D}_1) + (1-\alpha) f(\boldsymbol{D}_2) = \alpha \operatorname{tr}(\boldsymbol{L}^{\frac{1}{2}} \boldsymbol{D}_1 \boldsymbol{R} \boldsymbol{D}_1 \boldsymbol{L}^{\frac{1}{2}}) + (1-\alpha) \operatorname{tr}(\boldsymbol{L}^{\frac{1}{2}} \boldsymbol{D}_2 \boldsymbol{R} \boldsymbol{D}_2 \boldsymbol{L}^{\frac{1}{2}}).$$

Thus (57) can be simplified to the following inequality after rearranging terms

$$\begin{aligned}
&\alpha(1-\alpha) \operatorname{tr}(\boldsymbol{L}^{\frac{1}{2}} \boldsymbol{D}_1 \boldsymbol{R} \boldsymbol{D}_2 \boldsymbol{L}^{\frac{1}{2}}) + \alpha(1-\alpha) \operatorname{tr}(\boldsymbol{L}^{\frac{1}{2}} \boldsymbol{D}_2 \boldsymbol{R} \boldsymbol{D}_1 \boldsymbol{L}^{\frac{1}{2}}) \\
&\quad \leq \alpha(1-\alpha) \operatorname{tr}(\boldsymbol{L}^{\frac{1}{2}} \boldsymbol{D}_1 \boldsymbol{R} \boldsymbol{D}_1 \boldsymbol{L}^{\frac{1}{2}}) + \alpha(1-\alpha) \operatorname{tr}(\boldsymbol{L}^{\frac{1}{2}} \boldsymbol{D}_2 \boldsymbol{R} \boldsymbol{D}_2 \boldsymbol{L}^{\frac{1}{2}}).
\end{aligned}$$

This is equivalent to

$$\operatorname{tr}(\boldsymbol{L}^{\frac{1}{2}} \boldsymbol{D}_1 \boldsymbol{R} \boldsymbol{D}_1 \boldsymbol{L}^{\frac{1}{2}}) + \operatorname{tr}(\boldsymbol{L}^{\frac{1}{2}} \boldsymbol{D}_2 \boldsymbol{R} \boldsymbol{D}_2 \boldsymbol{L}^{\frac{1}{2}}) - \operatorname{tr}(\boldsymbol{L}^{\frac{1}{2}} \boldsymbol{D}_1 \boldsymbol{R} \boldsymbol{D}_2 \boldsymbol{L}^{\frac{1}{2}}) - \operatorname{tr}(\boldsymbol{L}^{\frac{1}{2}} \boldsymbol{D}_2 \boldsymbol{R} \boldsymbol{D}_1 \boldsymbol{L}^{\frac{1}{2}}) \geq 0.$$

To show that the above inequality holds, we do the following transformation for the LHS

$$\begin{aligned}
&\operatorname{tr}(\boldsymbol{L}^{\frac{1}{2}} \boldsymbol{D}_1 \boldsymbol{R} \boldsymbol{D}_1 \boldsymbol{L}^{\frac{1}{2}}) + \operatorname{tr}(\boldsymbol{L}^{\frac{1}{2}} \boldsymbol{D}_2 \boldsymbol{R} \boldsymbol{D}_2 \boldsymbol{L}^{\frac{1}{2}}) - \operatorname{tr}(\boldsymbol{L}^{\frac{1}{2}} \boldsymbol{D}_1 \boldsymbol{R} \boldsymbol{D}_2 \boldsymbol{L}^{\frac{1}{2}}) - \operatorname{tr}(\boldsymbol{L}^{\frac{1}{2}} \boldsymbol{D}_2 \boldsymbol{R} \boldsymbol{D}_1 \boldsymbol{L}^{\frac{1}{2}}) \\
&\quad = \operatorname{tr}(\boldsymbol{L}^{\frac{1}{2}} \boldsymbol{D}_1 \boldsymbol{R} (\boldsymbol{D}_1 - \boldsymbol{D}_2) \boldsymbol{L}^{\frac{1}{2}}) + \operatorname{tr}(\boldsymbol{L}^{\frac{1}{2}} \boldsymbol{D}_2 \boldsymbol{R} (\boldsymbol{D}_2 - \boldsymbol{D}_1) \boldsymbol{L}^{\frac{1}{2}}) \\
&\quad = \operatorname{tr}(\boldsymbol{L}^{\frac{1}{2}} (\boldsymbol{D}_1 - \boldsymbol{D}_2) \boldsymbol{R} (\boldsymbol{D}_1 - \boldsymbol{D}_2) \boldsymbol{L}^{\frac{1}{2}}).
\end{aligned}$$

Since $\boldsymbol{R} \in \mathbb{S}^d_+$ and $\boldsymbol{D}_1 - \boldsymbol{D}_2, \boldsymbol{L}$ are symmetric, for any vector $u \in \mathbb{R}^d$

$$u^\top \boldsymbol{L}^{\frac{1}{2}} (\boldsymbol{D}_1 - \boldsymbol{D}_2) \boldsymbol{R} (\boldsymbol{D}_1 - \boldsymbol{D}_2) \boldsymbol{L}^{\frac{1}{2}} u = \left((\boldsymbol{D}_1 - \boldsymbol{D}_2) \boldsymbol{L}^{\frac{1}{2}} u\right)^\top \boldsymbol{R} \left((\boldsymbol{D}_1 - \boldsymbol{D}_2) \boldsymbol{L}^{\frac{1}{2}} u\right) \geq 0. \tag{58}$$

Thus, $\boldsymbol{L}^{\frac{1}{2}} (\boldsymbol{D}_1 - \boldsymbol{D}_2) \boldsymbol{R} (\boldsymbol{D}_1 - \boldsymbol{D}_2) \boldsymbol{L}^{\frac{1}{2}} \in \mathbb{S}^d_+$, which yields the positivity of its trace. Therefore, (57) holds, thus $f(D)$ is a convex function. This concludes the proof.

F.2. Proof of Lemma 9

We have

$$\begin{aligned}
\mathbb{E}\left[\|x-\mathbb{E}[x]\|_{\boldsymbol{M}}^2\right] &= \mathbb{E}\left[\langle x-\mathbb{E}[x], \boldsymbol{M}(x-\mathbb{E}[x])\rangle\right] \\
&= \mathbb{E}\left[(x-\mathbb{E}[x])^\top \boldsymbol{M}(x-\mathbb{E}[x])\right] \\
&= \mathbb{E}\left[x^\top \boldsymbol{M}x - \mathbb{E}[x]^\top \boldsymbol{M}x - x^\top \boldsymbol{M}\mathbb{E}[x] + \mathbb{E}[x]^\top \boldsymbol{M}\mathbb{E}[x]\right] \\
&= \mathbb{E}\left[x^\top \boldsymbol{M}x\right] - 2\mathbb{E}[x]^\top \boldsymbol{M}\mathbb{E}[x] + \mathbb{E}[x]^\top \boldsymbol{M}\mathbb{E}[x] \\
&= \mathbb{E}\left[x^\top \boldsymbol{M}x\right] - \mathbb{E}[x]^\top \boldsymbol{M}\mathbb{E}[x] \\
&= \mathbb{E}\left[\|x\|_{\boldsymbol{M}}^2\right] - \|\mathbb{E}[x]\|_{\boldsymbol{M}}^2,
\end{aligned}$$

which concludes the proof.

F.3. Proof of Lemma 10

Proof We have

$$\begin{aligned}
\mathbb{E}\left[\left\|\frac{1}{n}\sum_{i=1}^n a_i\right\|_{\boldsymbol{M}}^2\right] &= \frac{1}{n^2}\sum_{i=1}^n \mathbb{E}\left[\langle a_i, \boldsymbol{M}a_i\rangle\right] + \frac{1}{n^2}\sum_{i\neq j}\mathbb{E}\left[\langle a_i, \boldsymbol{M}a_j\rangle\right] \\
&= \frac{1}{n^2}\sum_{i=1}^n \mathbb{E}\left[\|a_i\|_{\boldsymbol{M}}^2\right] + \frac{1}{n}\sum_{i\neq j}\langle \mathbb{E}[a_i], \boldsymbol{M}\mathbb{E}[a_j]\rangle \\
&= \frac{1}{n^2}\sum_{i=1}^n \mathbb{E}\left[\|a_i\|_{\boldsymbol{M}}^2\right].
\end{aligned}$$

This concludes the proof. ■

F.4. Proof of Lemma 11

Notice that

$$\mathbb{E}[\boldsymbol{S}x] = \mathbb{E}[\boldsymbol{S}]x = x.$$

We start with variance decomposition in the matrix norm,

$$\begin{aligned}
\mathbb{E}\left[\|\boldsymbol{S}x - x\|_{\boldsymbol{M}}^2\right] &\overset{(28)}{=} \mathbb{E}\left[\|\boldsymbol{S}x\|_{\boldsymbol{M}}^2\right] - \|x\|_{\boldsymbol{M}}^2 \\
&= \mathbb{E}\left[\langle \boldsymbol{S}x, \boldsymbol{M}\boldsymbol{S}x\rangle\right] - \langle x, \boldsymbol{M}x\rangle \\
&= \langle x, \mathbb{E}[\boldsymbol{S}\boldsymbol{M}\boldsymbol{S}]x\rangle - \langle x, \boldsymbol{M}x\rangle \\
&= \langle x, (\mathbb{E}[\boldsymbol{S}\boldsymbol{M}\boldsymbol{S}] - \boldsymbol{M})x\rangle \\
&= \|x\|_{\mathbb{E}[\boldsymbol{S}\boldsymbol{M}\boldsymbol{S}]-\boldsymbol{M}}^2.
\end{aligned}$$

This concludes the proof.

F.5. Proof of Lemma 12

We follow the definition of $\boldsymbol{L}$ matrix smoothness of function f, that for any $x^+, x \in \mathbb{R}^d$, we have

$$f(x^+) \leq f(x) + \left\langle \nabla f(x), x^+ - x \right\rangle + \frac{1}{2} \left\langle x^+ - x, \boldsymbol{L}(x^+ - x) \right\rangle .$$

We plug in $x^+ = x - \boldsymbol{L}^{-1}\nabla f(x)$, and get

$$f^{\inf} \leq f(x^+) \leq f(x) - \left\langle \nabla f(x), \boldsymbol{L}^{-1}\nabla f(x) \right\rangle + \frac{1}{2} \left\langle \nabla f(x), \boldsymbol{L}^{-1}\nabla f(x) \right\rangle .$$

Rearranging terms we get

$$\|\nabla f(x)\|^2_{\boldsymbol{L}^{-1}} \leq 2\left(f(x) - f^{\inf}\right), \tag{59}$$

which completes the proof.

F.6. Proof of Lemma 13

$$\begin{aligned}
\mathbb{E}\left[\|\boldsymbol{S} - x\|^2_{\boldsymbol{DLD}}\right] &= \mathbb{E}\left[\langle (\boldsymbol{S} - \boldsymbol{I}_d)x, \boldsymbol{DLD}(\boldsymbol{S} - \boldsymbol{I}_d)x\rangle\right] \\
&= \mathbb{E}\left[x^\top(\boldsymbol{S} - \boldsymbol{I}_d)\boldsymbol{DLD}(\boldsymbol{S} - \boldsymbol{I}_d)x\right] \\
&= x^\top \mathbb{E}\left[(\boldsymbol{S} - \boldsymbol{I}_d)\boldsymbol{DLD}(\boldsymbol{S} - \boldsymbol{I}_d)\right] x \\
&= x^\top \boldsymbol{A}^{-\frac{1}{2}}\left(\boldsymbol{A}^{\frac{1}{2}}\mathbb{E}\left[(\boldsymbol{S} - \boldsymbol{I}_d)\boldsymbol{DLD}(\boldsymbol{S} - \boldsymbol{I}_d)\right]\boldsymbol{A}^{\frac{1}{2}}\right)\boldsymbol{A}^{-\frac{1}{2}}x \\
&\leq \lambda_{\max}\left(\boldsymbol{A}^{\frac{1}{2}}\mathbb{E}\left[(\boldsymbol{S} - \boldsymbol{I}_d)\boldsymbol{DLD}(\boldsymbol{S} - \boldsymbol{I}_d)\right]\boldsymbol{A}^{\frac{1}{2}}\right)\left\|\boldsymbol{A}^{-\frac{1}{2}}x\right\|^2 \\
&= \lambda_{\max}\left(\boldsymbol{A}^{\frac{1}{2}}\mathbb{E}\left[(\boldsymbol{S} - \boldsymbol{I}_d)\boldsymbol{DLD}(\boldsymbol{S} - \boldsymbol{I}_d)\right]\boldsymbol{A}^{\frac{1}{2}}\right)\|x\|^2_{\boldsymbol{A}^{-1}} .
\end{aligned}$$

This completes the proof.

F.7. Proof of Lemma 15

$$\begin{aligned}
\mathbb{E}\left[\left\|\boldsymbol{T}_i^k\boldsymbol{D}x - \boldsymbol{D}x\right\|^2_{\boldsymbol{L}}\right] &= \mathbb{E}\left[\left\langle (\boldsymbol{T}_i^k - \boldsymbol{I}_d)\boldsymbol{D}x, \boldsymbol{L}(\boldsymbol{T}_i^k - \boldsymbol{I}_d)\boldsymbol{D}x\right\rangle\right] \\
&= \mathbb{E}\left[x^\top\boldsymbol{D}(\boldsymbol{T}_i^k - \boldsymbol{I}_d)\boldsymbol{L}(\boldsymbol{T}_i^k - \boldsymbol{I}_d)\boldsymbol{D}x\right] \\
&= x^\top\boldsymbol{D}\mathbb{E}\left[(\boldsymbol{T}_i^k - \boldsymbol{I}_d)\boldsymbol{L}(\boldsymbol{T}_i^k - \boldsymbol{I}_d)\right]\boldsymbol{D}x \\
&= x^\top\boldsymbol{L}_i^{-\frac{1}{2}}\left(\boldsymbol{L}_i^{\frac{1}{2}}\boldsymbol{D}\mathbb{E}\left[(\boldsymbol{T}_i^k - \boldsymbol{I}_d)\boldsymbol{L}(\boldsymbol{T}_i^k - \boldsymbol{I}_d)\right]\boldsymbol{D}\boldsymbol{L}_i^{\frac{1}{2}}\right)\boldsymbol{L}_i^{-\frac{1}{2}}x \\
&\leq \lambda_{\max}\left(\boldsymbol{L}_i^{\frac{1}{2}}\boldsymbol{D}\mathbb{E}\left[(\boldsymbol{T}_i^k - \boldsymbol{I}_d)\boldsymbol{L}(\boldsymbol{T}_i^k - \boldsymbol{I}_d)\right]\boldsymbol{D}\boldsymbol{L}_i^{\frac{1}{2}}\right)\left\|\boldsymbol{L}_i^{-\frac{1}{2}}x\right\|^2 \\
&= \lambda_{\max}\left(\boldsymbol{L}_i^{\frac{1}{2}}\boldsymbol{D}\mathbb{E}\left[(\boldsymbol{T}_i^k - \boldsymbol{I}_d)\boldsymbol{L}(\boldsymbol{T}_i^k - \boldsymbol{I}_d)\right]\boldsymbol{D}\boldsymbol{L}_i^{\frac{1}{2}}\right)\|x\|^2_{\boldsymbol{L}_i^{-1}} .
\end{aligned}$$

This completes the proof.

Appendix G. Experiments

In this section, we describe the settings and results of numerical experiments to demonstrate the effectiveness of our method. We perform several experiments under single node case and distributed case. The code is available at `https://anonymous.4open.science/r/detCGD_Code-A87D/`.

G.1. Single node case

For the single node case, we study the logistic regression problem with a non-convex regularizer. The objective is given as

$$f(x) = \frac{1}{n}\sum_{i=1}^{n} \log\left(1 + e^{-b_i \cdot \langle a_i, x\rangle}\right) + \lambda \cdot \sum_{j=1}^{d} \frac{x_j^2}{1 + x_j^2},$$

where $x \in \mathbb{R}^d$ is the model, $(a_i, b_i) \in \mathbb{R}^d \times \{-1, +1\}$ is one data point in the dataset whose size is n. The constant $\lambda > 0$ is a tunable hyperparameter associated with the regularizer. We conduct numerical experiments using several datasets from the LibSVM repository [6]. We estimate the smoothness matrix of function f here as

$$\boldsymbol{L} = \frac{1}{n}\sum_{i=1}^{n} \frac{a_i a_i^\top}{4} + 2\lambda \cdot \boldsymbol{I}_d.$$

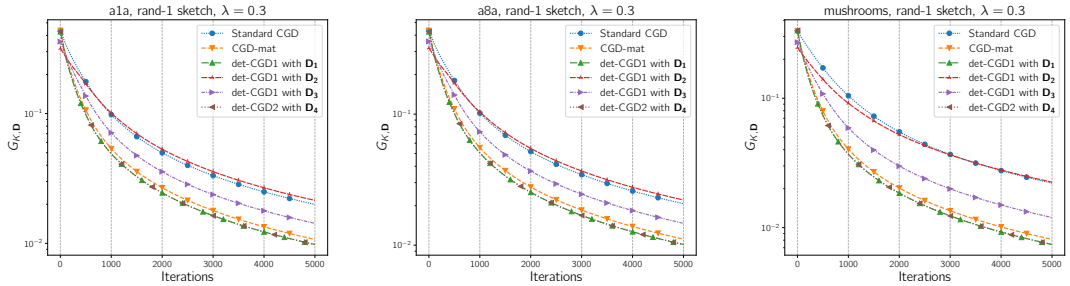
G.1.1 Comparison to CGD with scalar stepsize, scalar smoothness constant

The purpose of the first experiment is to show that by using matrix stepsize, det-CGD1 and det-CGD2 will have better iteration and communication complexities compared to standard CGD. We run a CGD with scalar stepsize γ and a scalar smoothness constant $L = \lambda_{\max}(\boldsymbol{L})$ and CGD with scalar stepsize $\gamma \cdot \boldsymbol{I}_d$ and smoothness matrix $\boldsymbol{L}$. We use standard CGD to refer to the CGD with scalar stepsize, scalar smoothness constant, and CGD-mat to refer to CGD with scalar stepsize, smoothness matrix in Figure 1, 2 and 3. The notation $G_{K,\boldsymbol{D}}$ appears in the label of y axis is defined as

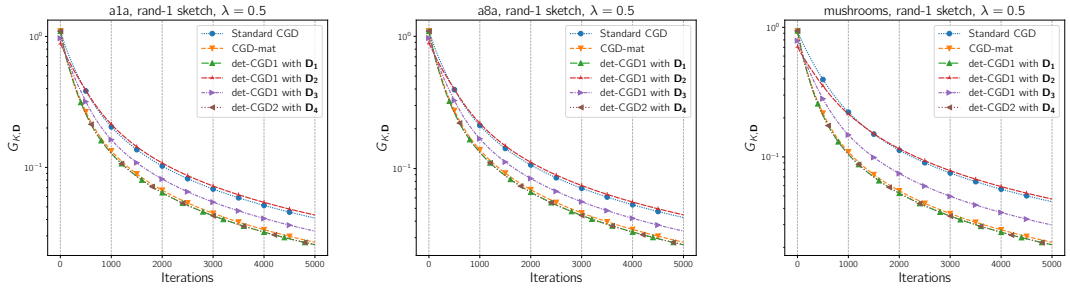
$$G_{K,\boldsymbol{D}} := \frac{1}{K}\left(\sum_{k=0}^{K-1} \left\|\nabla f(x^k)\right\|^2_{\frac{\boldsymbol{D}}{\det(\boldsymbol{D})^{1/d}}}\right), \tag{60}$$

it is the average matrix norm of the gradient of f over the first $K-1$ iterations in log scale. The weight matrix here has determinant 1, and thus it is comparable to the standard Euclidean norm. The result is meaningful in this sense.

The result presented in Figure 1 and Figure 2 suggest that compared to the standard CGD [27], CGD-mat performs better in terms of both iteration complexity and communication complexity. Furthermore, det-CGD1 and det-CGD2 with the best diagonal matrix stepsizes outperform both CGD and CGD-mat which confirms our theory. The scaling factors $\gamma_1, \gamma_2, \gamma_3$ here for det-CGD1 are determined using Theorem 4 with $\ell = 1$. The matrix stepsize for det-CGD2 is determined through (7). det-CGD1 and det-CGD2 with diagonal matrix stepsizes perform very similarly in the experiment, this is expected since we are using rand-1 sketch, which means that the stepsize matrix and the sketch matrix are commutable since they are both diagonal. We also notice that det-CGD1

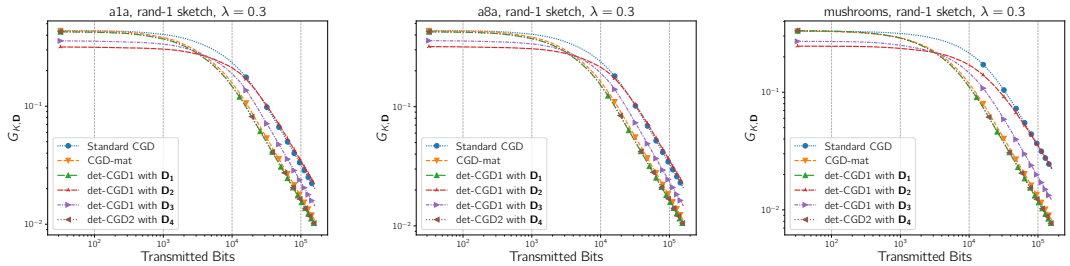


(*a*)

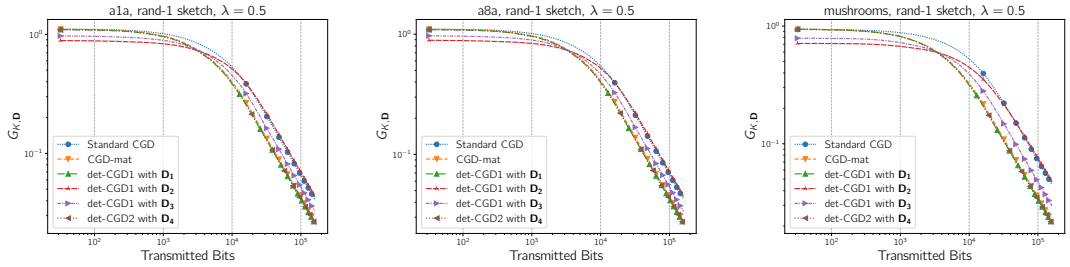


(*b*)

Figure 1: Comparison, in terms of iteration complexity, of standard CGD, CGD-mat, det-CGD1 with $\boldsymbol{D}_1 = \gamma_1 \cdot \mathrm{diag}^{-1}(\boldsymbol{L})$, det-CGD1 with $\boldsymbol{D}_2 = \gamma_2 \cdot \boldsymbol{L}^{-1}$, det-CGD1 with $\boldsymbol{D}_3 = \gamma_3 \cdot \boldsymbol{L}^{-1/2}$ and det-CGD2 with $\boldsymbol{D}_4 = \gamma_4 \cdot \mathrm{diag}^{-1}(\boldsymbol{L})$, where $\gamma_1, \gamma_2, \gamma_3$ are the optimal scaling factors for det-CGD1 in that case, $\boldsymbol{D}_4$ is the optimal matrix stepsize for det-CGD2. Rand-1 sketch is used in all the methods through out the experiments. The x-axis is the number of iterations performed. The notation $G_{K,\boldsymbol{D}}$ in the y-axis is defined in (60).



(*a*)



(*b*)

Figure 2: Comparison, in terms of communication complexity, of standard CGD, CGD-mat, det-CGD1 with $\boldsymbol{D}_1 = \gamma_1 \cdot \mathrm{diag}^{-1}(\boldsymbol{L})$, det-CGD1 with $\boldsymbol{D}_2 = \gamma_2 \cdot \boldsymbol{L}^{-1}$, det-CGD1 with $\boldsymbol{D}_3 = \gamma_3 \cdot \boldsymbol{L}^{-1/2}$ and det-CGD2 with $\boldsymbol{D}_4 = \gamma_4 \cdot \mathrm{diag}^{-1}(\boldsymbol{L})$, where $\gamma_1, \gamma_2, \gamma_3$ are the optimal scaling factors for det-CGD1 in that case, $\boldsymbol{D}_4$ is the optimal matrix stepsize for det-CGD2. Rand-1 sketch is used in all the methods through out the experiments. The x-axis is the number of bits transmitted. The notation $G_{K,\boldsymbol{D}}$ in the y-axis is defined in (60).

with $\boldsymbol{D}_2 = \gamma_2 \cdot \boldsymbol{L}^{-1}$ is always worse than $\boldsymbol{D}_4 = \gamma_4 \cdot \mathrm{diag}^{-1}(\boldsymbol{L})$, this is also expected since we mentioned in Appendix D.5.1 that the result row 5 (corresponding to $\boldsymbol{D}_2$) in Table 1 is always worse than row 7 (corresponding to $\boldsymbol{D}_4$).

G.1.2 Comparison of the two algorithms under the same stepsize

The purpose of the second experiment is to compare the performance of det-CGD1 and det-CGD2 in terms of iteration complexity and communication complexity. We know the conditions for det-CGD1 and det-CGD2 to converge are given by (3) and (4) respectively. As a result, we are able to obtain the optimal matrix stepsize for det-CGD2 if we are using rand-τ sparsification. It is given by

$$\boldsymbol{D}_2^* = \frac{\tau}{d}\left(\frac{d-\tau}{d-1}\,\mathrm{diag}(\boldsymbol{L}) + \frac{\tau-1}{d-1}\boldsymbol{L}\right)^{-1},$$

according to (6). The definition of $G_{K,\boldsymbol{D}}$ is given in (60). Parameter τ here for random sparsification is set to be an the integer part $\{\frac{d}{4}, \frac{d}{2}, \frac{3d}{4}\}$, where d is the dimension of the model.

It can be observed from the result presented in Figure 3, that in almost all cases in this experiment, 2 with $\boldsymbol{D} = \boldsymbol{D}_2^*$ outperforms the other methods. Compared to standard CGD and CGD with matrix stepsize, det-CGD1 and det-CGD2 are always better. This provides numerical evidence in support of our theory. In this case, the stepsize matrix is not diagonal for det-CGD1 and det-CGD2, so we do not expect them to perform similarly. Notice that in dataset `phishing`, the four algorithms behave very similarly, this is because the smoothness matrix $\boldsymbol{L}$ here has a concentrated spectrum.

G.2. Distributed case

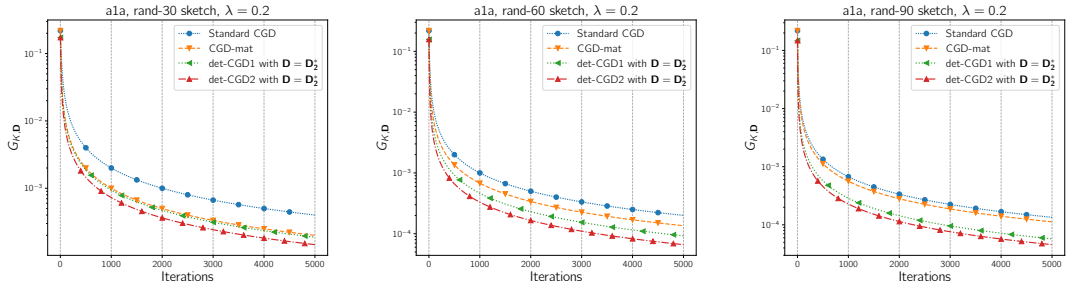
For the distributed case, we again use the logistic regression problem with a non-convex regularizer as our experiment setting. The objective is given similarly as

$$f(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(x); \quad f_i(x) = \frac{1}{m_i}\sum_{j=1}^{m_i} \log\left(1 + e^{-b_{i,j}\cdot\langle a_{i,j},x\rangle}\right) + \lambda \cdot \sum_{t=1}^{d} \frac{x_t^2}{1+x_t^2},$$

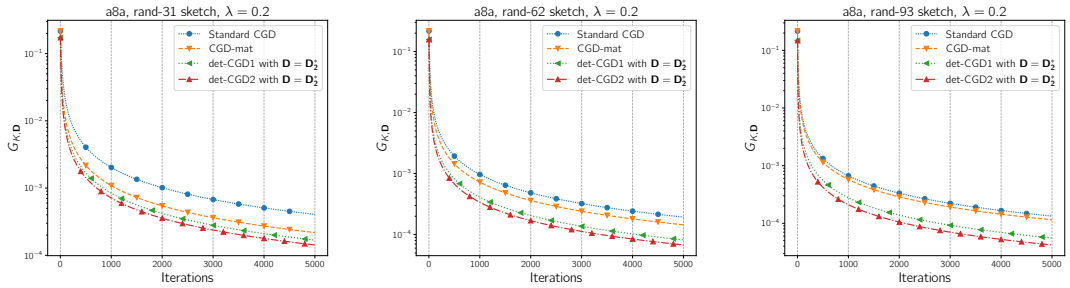
where $x \in \mathbb{R}^d$ is the model, $(a_{i,j}, b_{i,j}) \in \mathbb{R}^d \times \{-1,+1\}$ is one data point in the dataset of client i whose size is m_i. $\lambda > 0$ is a constant associated with the regularizer. For each dataset used in the distributed setting, we randomly reshuffled the dataset before splitting it equally to each client. We estimate the smoothness matrices of function f and each individual function f_i here as

$$\begin{aligned}\boldsymbol{L}_i &= \frac{1}{m_i}\sum_{i=1}^{m_i} \frac{a_i a_i^\top}{4} + 2\lambda \cdot \boldsymbol{I}_d;\\ \boldsymbol{L} &= \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{L}_i.\end{aligned}$$

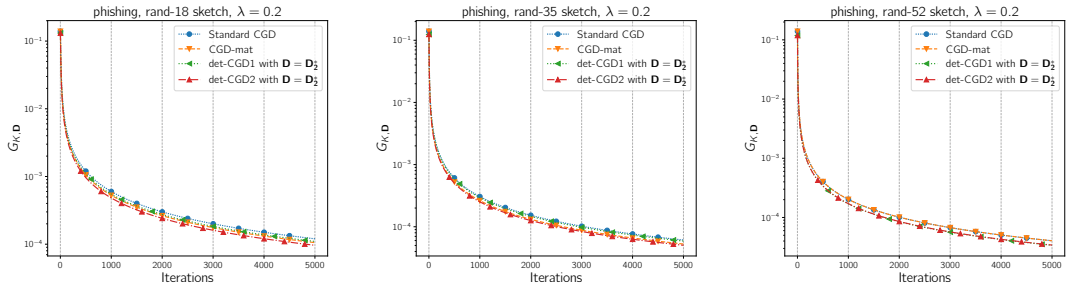
The value of $\Delta^{\inf}$ here is determined in the following way, we first perform gradient descent on f and record the minimum value in the entire run, $f^{\inf}$, as the estimate of its global minimum, then we do the same procedure for each f_i to obtain the estimate of its global minimum $f_i^{\inf}$. After that we estimate $\Delta^{\inf}$ using its definition.



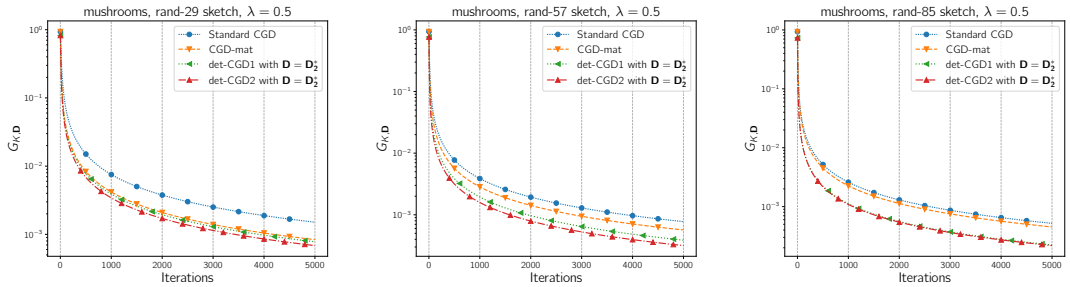
(*a*)



(*b*)



(*c*)



(*d*)

Figure 3: Comparison of standard CGD, CGD-mat det-CGD1 with stepsize $\boldsymbol{D} = \boldsymbol{D}_2^*$ and det-CGD2 with stepsize $\boldsymbol{D} = \boldsymbol{D}_2^*$, where $\boldsymbol{D}_2^*$ is the optimal stepsize matrix for det-CGD2 and the optimal diagonal stepsize matrix for det-CGD1. Rand-τ sketch is used in all the algorithms throughout the experiments. The notation $G_{K,\boldsymbol{D}}$ in the y-axis is defined in (60).

G.2.1 Comparison to standard DCGD in the distributed case

To ease the reading of this section we use D-det-CGD1 (resp. D-det-CGD2) to refer to Algorithm 1 (resp. Algorithm 2). This experiment is designed to show that D-det-CGD1 and D-det-CGD2 will have better iteration and communication complexity compared to standard DCGD [27] and DCGD with scalar stepsize, smoothness matrix. We will use the standard DCGD here to refer to DCGD with a scalar stepsize and a scalar smoothness constant, and DCGD-mat to refer to the DCGD with a scalar stepsize with smoothness. The Rand-1 sparsifier is used in all the algorithms throughout the experiment. The error level is fixed as $\varepsilon^2 = 0.0001$, the conditions for the standard DCGD to converge can be deduced using Proposition 4 in [26], we use the largest possible scalar stepsize here for standard DCGD. The optimal scalar stepsize for DCGD-mat, optimal diagonal matrix stepsize $\boldsymbol{D}_1$ for D-det-CGD1 and $\boldsymbol{D}_2$ for D-det-CGD2 can be determined using Corollary 8.

From the result of Figure 4 and Figure 5, we are able to see that both D-det-CGD1 and D-det-CGD2 outperform standard DCGD and DCGD-mat in terms of iteration complexity and communication complexity, which confirms our theory. Notice that D-det-CGD1, D-det-CGD2 are expected to perform very similarly because the stepsize matrix and sketches are diagonal which means that they are commutable. We also plot the corresponding standard Euclidean norm of iterates of D-det-CGD1 and D-det-CGD2 in Figure 6, the E_K here appears in the y-axis is defined as,

$$E_K := \frac{1}{K} \sum_{k=0}^{K-1} \left\| \nabla f(x^k) \right\|^2 . \tag{61}$$

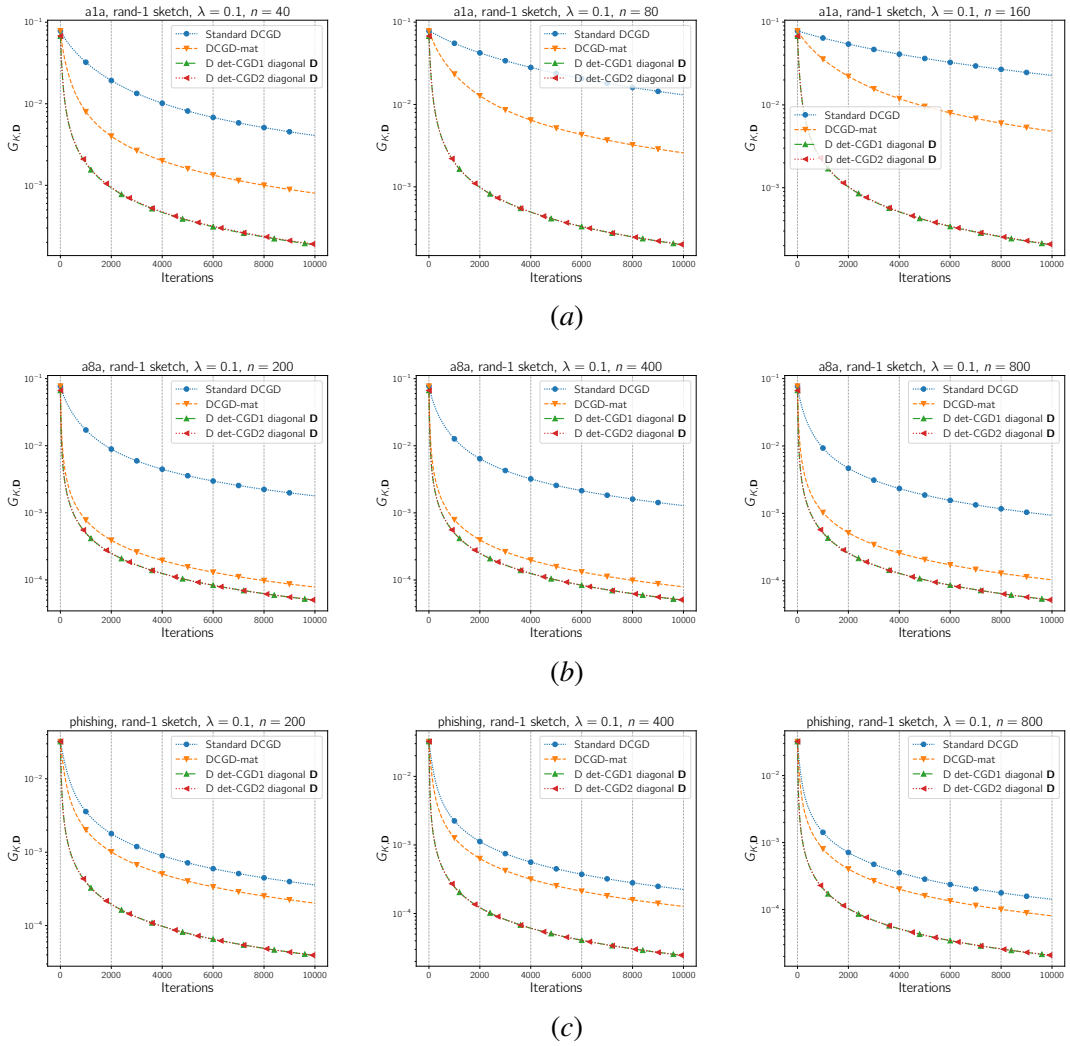
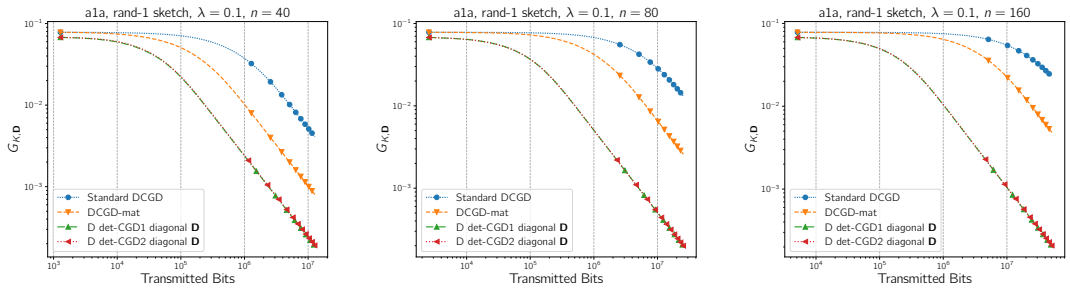
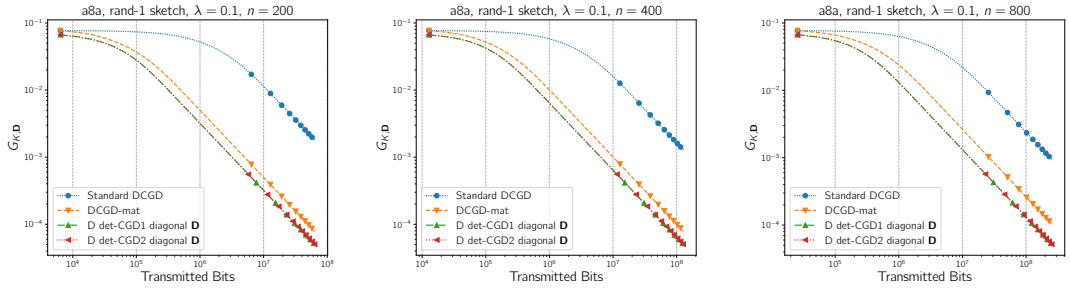


Figure 4: Comparison, in terms of iteration complexity, of standard DCGD, DCGD-mat, D-det-CGD1 with matrix stepsize $\boldsymbol{D}_1$ and D-det-CGD2 with matrix stepsize $\boldsymbol{D}_2$, where $\boldsymbol{D}_1, \boldsymbol{D}_2$ are the optimal diagonal matrix stepsizes for D-det-CGD1 and D-det-CGD2 respectively. Rand-1 sketch is used in all the algorithms throughout the experiment. The x-axis is the number of iterations performed. The notation $G_{K,\boldsymbol{D}}$ in the y-axis is defined in (60).



(*a*)



(*b*)

Figure 5: Comparison, in terms of communication complexity, of standard DCGD, DCGD-mat, D-det-CGD1 with matrix stepsize $\boldsymbol{D}_1$ and D-det-CGD2 with matrix stepsize $\boldsymbol{D}_2$, where $\boldsymbol{D}_1, \boldsymbol{D}_2$ are the optimal diagonal matrix stepsizes for D-det-CGD1 and D-det-CGD2 respectively. Rand-1 sketch is used in all the algorithms throughout the experiment. The x-axis is the number of bits transmitted. The notation $G_{K,\boldsymbol{D}}$ in the y-axis is defined in (60).

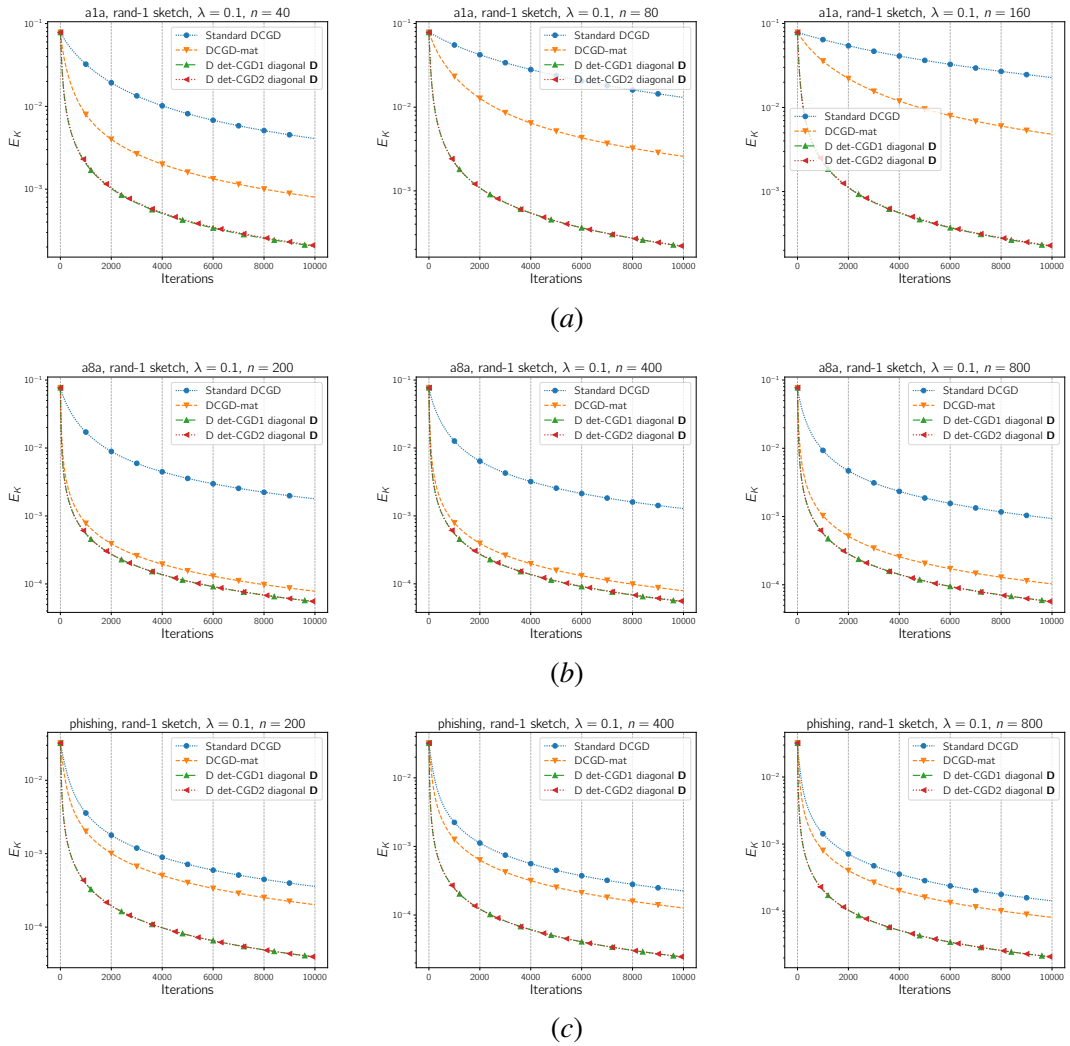


Figure 6: Comparison, in terms of iteration complexity, of standard DCGD, DCGD-mat, D-det-CGD1 with matrix stepsize $\boldsymbol{D}_1$ and D-det-CGD2 with matrix stepsize $\boldsymbol{D}_2$, where $\boldsymbol{D}_1, \boldsymbol{D}_2$ are the optimal diagonal matrix stepsizes for D-det-CGD1 and D-det-CGD2 respectively. Rand-1 sketch is used in all the algorithms throughout the experiment. The x-axis is the number of iterations. The y-axis is now standard Euclidean norm defined in (61).