CANITA: Faster Rates for Distributed Convex Optimization with Communication Compression

Zhize Li
https://zhizeli.github.io
King Abdullah University of Science and Technology (KAUST)

Joint work with **Peter Richtárik** (KAUST)

NeurIPS 2021

Overview

Problem

Training distributed/federated learning models is typically performed by solving an optimization problem

$$\min_{x\in\mathbb{R}^d} \left\{ f(x) \overset{\text{def}}{=} \frac{1}{n}\sum_{i=1}^{n} f_i(x) \right\},$$

x: model parameters

d: number of parameters (dimension)

n: number of devices/machines/nodes/workers

$f_i(x)$: loss function associated with data stored on device i

Examples

$$\min_{x\in\mathbb{R}^d}\left\{f(x)\stackrel{\text{def}}{=}\frac{1}{n}\sum_{i=1}^{n}f_i(x)\right\}$$

Each device i stores m data samples $\{a_{i,j}, b_{i,j}\}_{j=1}^m \in \mathbb{R}^{d+1}$ ($b_{i,j}$ is the label of data $a_{i,j}$)

- **Ordinary least squares**: $f_i(x) = \frac{1}{m}\sum_{j=1}^m (a_{i,j}^T x - b_{i,j})^2$
- **Logistic regression**: $f_i(x) = \frac{1}{m}\sum_{j=1}^m \log\left(1+\exp(-b_{i,j}a_{i,j}^T x)\right)$
- **SVM**: $f_i(x) = \frac{1}{m}\sum_{j=1}^m \max\left(0, 1-b_{i,j}a_{i,j}^T x\right) + \frac{\lambda}{2}\|x\|_2^2$

Goal

$$\min_{x \in \mathbb{R}^d} \left\{ f(x) \stackrel{\text{def}}{=} \frac{1}{n} \sum_{i=1}^{n} f_i(x) \right\}$$

Goal: find an ϵ-solution (parameters) $\hat{x}$, e.g., $f(\hat{x}) - f(x^*) \leq \epsilon$, where $x^* := \arg\min_{x \in \mathbb{R}^d} f(x)$.

Goal

$$\min_{x\in\mathbb{R}^d} \left\{ f(x) \overset{\text{def}}{=} \frac{1}{n}\sum_{i=1}^{n} f_i(x) \right\}$$

Goal: find an ϵ-solution (parameters) $\hat{x}$, e.g., $f(\hat{x}) - f(x^*) \leq \epsilon$, where $x^* := \arg\min_{x\in\mathbb{R}^d} f(x)$.

For distributed optimization methods:

Bottleneck: communication cost

Common strategy: **compress** the communicated messages (lower communication cost per communication round) and hope that this will not increase the total number of communication rounds.

Related Work

- Several recent work show that the total communication complexity can be improved via **compression**. See, e.g., QSGD (Alistarh et al., NIPS'17), DIANA (Mishchenko et al., arXiv'19), Natural compression (Horváth et al., arXiv'19), and MARINA (Gorbunov et al., ICML'21).

Related Work

- Several recent work show that the total communication complexity can be improved via **compression**. See, e.g., QSGD (Alistarh et al., NIPS'17), DIANA (Mishchenko et al., arXiv'19), Natural compression (Horváth et al., arXiv'19), and MARINA (Gorbunov et al., ICML'21).
- However previous work usually lead to this kind of improvement:

  Communication cost per round (- -) Rounds (+) $\Rightarrow$ Total (-)

  '-' denotes **decrease**, '+' denotes **increase**

Related Work

- Several recent work show that the total communication complexity can be improved via **compression**. See, e.g., QSGD (Alistarh et al., NIPS'17), DIANA (Mishchenko et al., arXiv'19), Natural compression (Horváth et al., arXiv'19), and MARINA (Gorbunov et al., ICML'21).
- However previous work usually lead to this kind of improvement:

  Communication cost per round (- -) Rounds (+) $\Rightarrow$ Total (-)

  '-' denotes **decrease**, '+' denotes **increase**
- **Acceleration/Momentum** of gradient-type methods is widely studied for achieving faster convergence rates (fewer iterations).

> *"Can distributed gradient-type methods theoretically benefit from the combination of* **compression** *and* **acceleration***?"*

Related Work

- Recently, Li et al. (ICML'20)[1] gave the first successful combination of compression and acceleration by proposing **ADIANA** method.

[1] Zhize Li, Dmitry Kovalev, Xun Qian, and Peter Richtárik. Acceleration for compressed gradient descent in distributed and federated optimization. In *ICML*, 2020.

Related Work

- Recently, Li et al. (ICML'20)[1] gave the first successful combination of compression and acceleration by proposing **ADIANA** method.
- Some drawbacks:
  - They only provide theoretical results for **strongly convex** problems. (e.g., logistic regression is convex but not strongly convex)
  - The ADIANA method is also not applicable to general convex case.
  - Even if a problem is strongly convex, the modulus of strong convexity is typically not known, or hard to estimate properly.

[1] Zhize Li, Dmitry Kovalev, Xun Qian, and Peter Richtárik. Acceleration for compressed gradient descent in distributed and federated optimization. In *ICML*, 2020.

Related Work

- Recently, Li et al. (ICML'20)[1] gave the first successful combination of compression and acceleration by proposing **ADIANA** method.
- Some drawbacks:
  - They only provide theoretical results for **strongly convex** problems. (e.g., logistic regression is convex but not strongly convex)
  - The ADIANA method is also not applicable to general convex case.
  - Even if a problem is strongly convex, the modulus of strong convexity is typically not known, or hard to estimate properly.
- Hence, one needs to design **new methods and analyses** to push forward this line of research (compression + acceleration).

[1] Zhize Li, Dmitry Kovalev, Xun Qian, and Peter Richtárik. Acceleration for compressed gradient descent in distributed and federated optimization. In *ICML*, 2020.

Our Contributions

- In this work, we propose and analyze a new **CANITA** method.
- CANITA is the **first** work provably combining the benefits of **compression** and **acceleration** in the **general convex** setting.

Our Contributions

- In this work, we propose and analyze a new **CANITA** method.
- CANITA is the **first** work provably combining the benefits of **compression** and **acceleration** in the **general convex** setting.

- Previous work (compression **without acceleration**):
  Communication cost per round (- -) Rounds (+) $\Rightarrow$ Total (-)
  '-' denotes **decrease**, '+' denotes **increase**

Our Contributions

- In this work, we propose and analyze a new **CANITA** method.
- CANITA is the **first** work provably combining the benefits of **compression** and **acceleration** in the **general convex** setting.
- Previous work (compression **without acceleration**):

  Communication cost per round (- -) Rounds (+) ⇒ Total (-)

  '-' denotes **decrease**, '+' denotes **increase**
- Our CANITA (compression **and acceleration**):

  Communication cost per round (- -) Rounds (- -) ⇒ Total (- - - -)

Our Contributions

Table: Communication rounds for finding an ϵ-solution $f(x^T) - f(x^*) \le \epsilon$

Algorithm	General convex	Remark
DIANA (Mishchenko et al., 2019)	$O\left(\left(1+\frac{\omega}{n}\right)\frac{L}{\epsilon}+\frac{\omega}{\epsilon}\right)$	✓compression ✗ acceleration
CANITA (this paper)	$O\left(\sqrt{\left(1+\sqrt{\frac{\omega^3}{n}}\right)\frac{L}{\epsilon}}+\omega\left(\frac{1}{\epsilon}\right)^{\frac{1}{3}}\right)$	✓compression ✓acceleration

***L*:** L-smooth parameter ($\|\nabla f(x) - \nabla f(y)\| \le L\|x-y\|$)

ω: compression parameter (no compression implies $\omega = 0$)

***n*:** number of devices/machines/nodes/workers

Our Contributions

Table: Communication rounds for finding an ϵ-solution $f(x^T) - f(x^*) \le \epsilon$

Algorithm	General convex	Remark
DIANA (Mishchenko et al., 2019)	$O\Big(\big(1+\frac{\omega}{n}\big)\frac{L}{\epsilon}+\frac{\omega}{\epsilon}\Big)$	✓ compression ✗ acceleration
CANITA (this paper)	$O\Big(\sqrt{\big(1+\sqrt{\frac{\omega^3}{n}}\big)\frac{L}{\epsilon}}+\omega\big(\frac{1}{\epsilon}\big)^{\frac{1}{3}}\Big)$	✓ compression ✓ acceleration

***L*:** L-smooth parameter $(\|\nabla f(x) - \nabla f(y)\| \le L\|x-y\|)$

ω: compression parameter (no compression implies $\omega = 0$)

***n*:** number of devices/machines/nodes/workers

- For example, if compression ratio is 0.1, then $\omega \approx 10$ (e.g. random sparsification). Further if $n = 10^6$ and $\epsilon = 10^{-6}$, then the result of our CANITA is $O(10^3)$, while the previous state-of-the-art DIANA is $O(10^6)$, i.e., $O(\sqrt{\frac{L}{\epsilon}})$ **vs.** $O(\frac{L}{\epsilon})$.

Our CANITA Algorithm

Our CANITA algorithm is based on the accelerated gradient method **ANITA** (Li, 2021)[2] which achieves the current state-of-the-art convergence result for **general convex** problems.

[2] Zhize Li. ANITA: An optimal loopless accelerated variance-reduced gradient method. *arXiv:2103.11333*, 2021.

ANITA vs. CANITA

ANITA (simplified)

1: **for** $t = 0, 1, 2, \ldots$ **do**
2: $y^t = \theta_t x^t + (1 - \theta_t) w^t$
3: Randomly pick $i \in \{1, 2, \ldots, n\}$
4: $g^t = \nabla f_i(y^t) - \nabla f_i(w^t) + \nabla f(w^t)$
5: $x^{t+1} = x^t - \frac{\eta_t}{\theta_t} g^t$
6: $z^{t+1} = \theta_t x^{t+1} + (1 - \theta_t) w^t$
7: $w^{t+1} = \begin{cases} z^{t+1}, & \text{with probability } p_t \\ w^t, & \text{with probability } 1 - p_t \end{cases}$
8: **end for**

ANITA vs. CANITA

ANITA (simplified)

for $t = 0, 1, 2, \ldots$ **do**
  $y^t = \theta_t x^t + (1 - \theta_t) w^t$
  Randomly pick $i \in \{1, 2, \ldots, n\}$
  $g^t = \nabla f_i(y^t) - \nabla f_i(w^t) + \nabla f(w^t)$
  $x^{t+1} = x^t - \frac{\eta_t}{\theta_t} g^t$
  $z^{t+1} = \theta_t x^{t+1} + (1 - \theta_t) w^t$
  $$w^{t+1} = \begin{cases} z^{t+1}, & \text{with probability } p_t \\ w^t, & \text{with probability } 1 - p_t \end{cases}$$
end for

- Compared with ANITA, our **CANITA** needs to deal with the **extra compression of shifted local gradients** in the distributed network.
- Hence, the obtained gradient estimator g^t is substantially different and more complicated, **which necessitates a novel proof technique**.

Our CANITA

for $t = 0, 1, 2, \ldots$ **do**
  $y^t = \theta_t x^t + (1 - \theta_t) w^t$
  for all nodes $i = 1, 2, \ldots, n$ **do in parallel**
    Compress the shifted local gradient $\mathcal{C}(\nabla f_i(y^t) - h_i^t)$ and send to the server
    Update the local shift
    $h_i^{t+1} = h_i^t + \alpha_t \mathcal{C}(\nabla f_i(w^t) - h_i^t)$
  end for
  Aggregate received compressed local gradient information:
  $$g^t = h^t + \frac{1}{n} \sum_{i=1}^{n} \mathcal{C}(\nabla f_i(y^t) - h_i^t)$$
  $$h^{t+1} = h^t + \alpha_t \frac{1}{n} \sum_{i=1}^{n} \mathcal{C}(\nabla f_i(w^t) - h_i^t)$$
  $x^{t+1} = x^t - \frac{\eta_t}{\theta_t} g^t$
  $z^{t+1} = \theta_t x^{t+1} + (1 - \theta_t) w^t$
  $$w^{t+1} = \begin{cases} z^{t+1}, & \text{with probability } p_t \\ w^t, & \text{with probability } 1 - p_t \end{cases}$$
end for

Conclusion

- We propose the **first compressed and accelerated** gradient method **CANITA** for distributed **general convex** optimization.

- We show that CANITA provably enjoys the benefits of both **compression** (compressed communication in each round) and **acceleration** (much fewer communication rounds).

- Previous work (compression **without acceleration**):
  Communication cost per round (- -) Rounds (+) $\Rightarrow$ Total (-)

- Our CANITA (compression **and acceleration**):
  Communication cost per round (- -) Rounds (- -) $\Rightarrow$ Total (- - - -)

Thanks!

Zhize Li