Optimizing Nonconvex Finite Sums by a Proximal Primal-Dual Method

Davood Hajinezhad

Iowa State University
Co-Authors

Mingyi Hong  
Iowa State University  

Tuo Zhao  
Johns Hopkins University  

Zhaoran Wang  
Princeton University  

To appear: NIPS 2016
Outline

1. Introduction
   - Motivation
   - Related Works
   - Applications

2. The Proposed Approach
   - Convergence Analysis
   - Comments

3. Connections with Existing Works

4. Numerical Results
Outline

1. Introduction
   - Motivation
   - Related Works
   - Applications

2. The Proposed Approach
   - Convergence Analysis
   - Comments

3. Connections with Existing Works

4. Numerical Results
Problem Formulation

Consider:

\[
\min_{z \in Z} f(z) := \frac{1}{N} \sum_{i=1}^{N} g_i(z) + g_0(z) + p(z),
\]

(1)

- \( g_i, \ i = 1, \cdots N \) are cost functions; \( g_0 \) and \( p \) are regularizers.

- \( p(z) \): lower semi-continuous, convex, possibly nonsmooth.

- \( Z \): Close and convex set.

- \( g_i \): \( L_i \)-smooth, possibly nonconvex.
Problem Formulation

- Consider:

\[
\min_{z \in Z} f(z) := \frac{1}{N} \sum_{i=1}^{N} g_i(z) + g_0(z) + p(z),
\]

where \( g_i \), \( i = 1, \ldots, N \) are cost functions; \( g_0 \) and \( p \) are regularizers.

- \( p(z) \): lower semi-continuous, convex, possibly nonsmooth.

- \( Z \): Close and convex set.

- \( g_i \): \( L_i \)-smooth, possibly nonconvex.
Consider:

$$\min_{z \in Z} f(z) := \frac{1}{N} \sum_{i=1}^{N} g_i(z) + g_0(z) + p(z),$$  \hspace{1cm} (1)

- $g_i, \ i = 1, \cdots N$ are cost functions; $g_0$ and $p$ are regularizers.
- $p(z)$: lower semi-continuous, convex, possibly nonsmooth.
- $Z$: Close and convex set.
- $g_i$: $L_i$-smooth, possibly nonconvex.
Problem Formulation

Consider:

$$\min_{z \in Z} f(z) := \frac{1}{N} \sum_{i=1}^{N} g_i(z) + g_0(z) + p(z),$$  \hspace{1cm} (1)$$

- $g_i, \ i = 1, \cdots N$ are cost functions; $g_0$ and $p$ are regularizers.
- $p(z)$: lower semi-continuous, convex, possibly nonsmooth.
- $Z$: Close and convex set.
- $g_i$: $L_i$-smooth, possibly nonconvex.
Motivation

- When $N$ is very large, would like to optimize one function at a time
- We would like to decompose across the functions

**Question:**

1. Find fast algorithms to achieve decomposition for non-convex problems?
2. Rigorously characterize the convergence rates?

**Main Idea:** Utilize both primal and dual information.
Motivation

- When $N$ is very large, would like to optimize one function at a time.
- We would like to decompose across the functions.

**Question:**
1. Find fast algorithms to achieve decomposition for non-convex problems?
2. Rigorously characterize the convergence rates?

**Main Idea:** Utilize both primal and dual information.
Motivation

- When $N$ is very large, would like to optimize one function at a time
- We would like to decompose across the functions

**Question:**
1. Find fast algorithms to achieve decomposition for non-convex problems?
2. Rigorously characterize the convergence rates?

**Main Idea:** Utilize both primal and dual information.
Motivation

- When $N$ is very large, we would like to optimize one function at a time.
- We would like to decompose across the functions.

**Question:**

1. Find fast algorithms to achieve decomposition for non-convex problems?
2. Rigorously characterize the convergence rates?

**Main Idea:** Utilize both primal and dual information.
This work

- Study the problem from a **primal-dual** perspective
- We develop a stochastic algorithm that achieves **sublinear** convergence rates to first-order stationary solutions
- Scaling constants up to $O(N)$ better than the (batch) gradient method
- For special quadratic problems, achieve **linear** convergence
- Identified some connections between the proposed primal-dual approach, with a few "primal only" methods such as IAG/SAG/SAGA
Related Works in Convex Case

- Extensive literature in convex case.
- Gradient Descent (GD):

\[
x^{r+1} = x^r - \alpha \frac{1}{N} \sum_{i=1}^{N} \nabla g_i(x^r_i)
\]

- Stochastic Gradient Descent (SG) [Robbins and Monro, 1951]:

Randomly pick an index \( i_r \in \{1, \cdots, N\} \)

\[
x^{r+1} = x^r - \frac{1}{\eta_{i_r}} \nabla g_{i_r}(x^r)
\]
Related Works in Convex Case

- Stochastic Average Gradient (SAG) [Le Roux 12]:
- Consider a special case of problem (1): \( p(x) = 0, g_0(x) = 0 \)

Randomly pick an index \( i_r \in \{1, \cdots, N\} \)

\[
y_{ir}^r = x^r, \quad y_j^r = y_j^{r-1}, \quad \forall \ j \neq i_r
\]

\[
x^{r+1} = x^r - \frac{1}{\eta N} \sum_{i=1}^{N} \nabla g_i(y_{ir}^r)
\]

average of past gradients

\[
= x^r - \frac{1}{\eta N} \sum_{i=1}^{N} \nabla g_i(y_{i}^{r-1}) + \frac{1}{\eta N} \left( \nabla g_{i_r}(y_{i-r}^{r-1}) - \nabla g_{i_r}(x^r) \right)
\]

- Only need to track the sum \( \sum_{i=1}^{N} \nabla g_i(y_{i}^{k-1}) \) and each of its components.
Related Works in Convex Case

- The SAGA algorithm [Defazio et al 14]:

Randomly pick an index $i_k \in \{1, \cdots, N\}$

\[
y_{i_k}^k = x^k, \quad y_{j}^k = y_{j}^{k-1}, \quad \forall \ j \neq i_k
\]

\[
w^{k+1} = x^k - \frac{1}{\eta N} \sum_{i=1}^{N} \nabla g_i(y_{i}^{k-1}) + \frac{1}{\eta} \left( \nabla g_{i_k}(y_{i_k}^{k-1}) - \nabla g_{i_k}(x^k) \right)
\]

\[
x^{k+1} = \text{prox}_{\frac{1}{\eta} h}(w^{k+1}) \quad \text{(Deal with nonsmooth part)}
\]

- It has better convergence rates.
- Again requires storage of past $N$ gradients.
Related Works in Convex Case

- Other related methods:
  - MISO [Mairal, 2013].
  - SDCA [Shalev-Schwartz and Zhang, 2013].

- But, these all introduce memory requirements

- Is it possible to achieve similar convergence rates as SAG, but without additional storage?

- Yes! Stochastic variance reduced gradient (SVRG) proposed in [Johnson-Zhang 13].
Related Works in Nonconvex Case

- **Classic algorithms**
  - Incremental Methods [Bertsekas 11]
  - SGD based algorithms [Ghadimi-Lan 13]

- **Recent fast algorithms**
  - Non-convex SVRG [Zhu et al 16, Reddi et al 16]
  - Non-convex SAGA [Reddi et al 16]

- The latter two algorithms only deal with smooth cases
Application: High Dimensional Regression

- $y \in \mathbb{R}^M$ observation vector generated: $y = X\nu + \epsilon$
  
  - $X \in \mathbb{R}^{M \times P}$: covariate matrix
  - $\nu \in \mathbb{R}^P$: ground truth
  - $\epsilon \in \mathbb{R}^M$: noise

- Only know an estimate of $X$: $A = X + W$ where $W \in \mathbb{R}^{M \times P}$ is the noise

- **Goal**: To estimate the ground truth $\nu$

- Set $\hat{\Gamma} := 1/M(A^\top A) - \Sigma_W$, and $\hat{\gamma} := 1/M(A^\top y)$

- **Non-Convex Optimization** [Loh, Wainwright 12]:

  $$\min_z z^\top \hat{\Gamma} z - \hat{\gamma} z \quad \text{s.t.} \quad \|z\|_1 \leq R.$$
Outline

1. Introduction
   - Motivation
   - Related Works
   - Applications

2. The Proposed Approach
   - Convergence Analysis
   - Comments

3. Connections with Existing Works

4. Numerical Results
Outline

1. Introduction
   - Motivation
   - Related Works
   - Applications

2. The Proposed Approach
   - Convergence Analysis
   - Comments

3. Connections with Existing Works

4. Numerical Results
Let us start using a distributed optimization setting

Split $z$ into $N$ new variables

Consider the following reformulation:

$$\min_{x,z \in \mathbb{R}^d} \frac{1}{N} \sum_{i=1}^{N} g_i(x_i) + g_0(z) + h(z),$$

s.t. $x_i = z, i = 1, \ldots, N$

where $h(z) := \nu(z) + p(z), x := [x_1; \cdots; x_N]$.

$N$ distributed agents, one central controller
A Reformulation

- Let us start using a distributed optimization setting
- Split $z$ into $N$ new variables
- Consider the following reformulation:

$$\min_{x,z \in \mathbb{R}^d} \frac{1}{N} \sum_{i=1}^{N} g_i(x_i) + g_0(z) + h(z),$$

s.t. $x_i = z$, $i = 1, \ldots, N$

where $h(z) := \nu_Z(z) + p(z)$, $x := [x_1; \cdots; x_N]$.

- $N$ distributed agents, one central controller
The Proposed Approach

A Reformulation

- Let us start using a distributed optimization setting
- Split $z$ into $N$ new variables
- Consider the following reformulation:

$$\min_{x,z \in \mathbb{R}^d} \frac{1}{N} \sum_{i=1}^{N} g_i(x_i) + g_0(z) + h(z),$$

s.t. $x_i = z$, $i = 1, \ldots, N$

where $h(z) := \nu_Z(z) + p(z)$, $x := [x_1; \cdots; x_N]$.

- $N$ distributed agents, one central controller
A Reformulation

- Let us start using a distributed optimization setting.
- Split \( z \) into \( N \) new variables.
- Consider the following reformulation:

  \[
  \min_{x,z \in \mathbb{R}^d} \frac{1}{N} \sum_{i=1}^{N} g_i(x_i) + g_0(z) + h(z),
  \]

  s.t. \( x_i = z, \ i = 1, \cdots, N \)

  where \( h(z) := \nu_Z(z) + p(z), \ x := [x_1; \cdots; x_N] \).

- \( N \) distributed agents, one central controller.
A Reformulation

- Distributed Setting:
The Proposed Approach

Assumptions

A-(a) \( f(z) \) is bounded from below, i.e. \( f := \min_{z \in Z} f(z) > -\infty \).

A-(b) Let us define \( g(z) = \sum_{i=1}^{N} \frac{1}{N} g_i(z) \); Suppose \( g_i \)'s and \( g \) have Lipschitz continuous gradients, i.e.,

\[
\|\nabla g_i(y) - \nabla g_i(z)\| \leq L_i \|y - z\|, \quad \forall \ y, z \in \text{dom}(g_i), \ i = 0, \ldots, N
\]
\[
\|\nabla g(y) - \nabla g(z)\| \leq L \|y - z\|, \quad \forall \ y, z \in \text{dom}(g).
\]

- From assumption A-(b): \( L \leq 1/N \sum_{i=1}^{N} L_i \), and the equality can be achieved in the worst case.

- Central node is able to broadcast to every node, and each node is able to send messages to the central node.
Assumptions

A-(a) \( f(z) \) is bounded from below, i.e. \( f := \min_{z \in Z} f(z) > -\infty \).

A-(b) Let us define \( g(z) = \sum_{i=1}^{N} \frac{1}{N} g_i(z) \); Suppose \( g_i \)'s and \( g \) have Lipschitz continuous gradients, i.e.,

\[
\| \nabla g_i(y) - \nabla g_i(z) \| \leq L_i \| y - z \|, \quad \forall \ y, z \in \text{dom}(g_i), \ i = 0, \cdots, N
\]
\[
\| \nabla g(y) - \nabla g(z) \| \leq L \| y - z \|, \quad \forall \ y, z \in \text{dom}(g).
\]

- From assumption A-(b): \( L \leq 1/N \sum_{i=1}^{N} L_i \), and the equality can be achieved in the worst case.
- Central node is able to broadcast to every node, and each node is able to send messages to the central node.
**Assumptions**

A-(a) $f(z)$ is bounded from below, i.e. $f := \min_{z \in Z} f(z) > -\infty$.

A-(b) Let us define $g(z) = \sum_{i=1}^{N} \frac{1}{N} g_i(z)$; Suppose $g_i$’s and $g$ have Lipschitz continuous gradients, i.e.,

\[
\|\nabla g_i(y) - \nabla g_i(z)\| \leq L_i \|y - z\|, \quad \forall \ y, z \in \text{dom}(g_i), \ i = 0, \cdots, N
\]

\[
\|\nabla g(y) - \nabla g(z)\| \leq L \|y - z\|, \quad \forall \ y, z \in \text{dom}(g).
\]

- From assumption A-(b): $L \leq \frac{1}{N} \sum_{i=1}^{N} L_i$, and the equality can be achieved in the worst case.

- Central node is able to broadcast to every node, and each node is able to send messages to the central node.
A-(a) \( f(z) \) is bounded from below, i.e. \( f := \min_{z \in Z} f(z) > -\infty \).

A-(b) Let us define \( g(z) = \sum_{i=1}^{N} \frac{1}{N} g_i(z) \); Suppose \( g_i \)'s and \( g \) have Lipschitz continuous gradients, i.e.,

\[
\|\nabla g_i(y) - \nabla g_i(z)\| \leq L_i \|y - z\|, \quad \forall \ y, z \in \text{dom}(g_i), \ i = 0, \cdots, N \\
\|\nabla g(y) - \nabla g(z)\| \leq L \|y - z\|, \quad \forall \ y, z \in \text{dom}(g).
\]

- From assumption A-(b): \( L \leq 1/N \sum_{i=1}^{N} L_i \), and the equality can be achieved in the worst case.

- Central node is able to broadcast to every node, and each node is able to send messages to the central node.
The Proposed Algorithm: Preliminaries

- Augmented Lagrangian:

\[ L(x, z; \lambda) = \sum_{i=1}^{N} \left( \frac{1}{N} g_i(x_i) + \langle \lambda_i, x_i - z \rangle + \frac{\eta_i}{2} \| x_i - z \|^2 \right) + g_0(z) + h(z) \]

- \( \lambda := \{ \lambda_i \}_{i=1}^{N} \) is the set of dual variables.

- \( \eta := \{ \eta_i > 0 \}_{i=1}^{N} \) are penalty parameters.
The Proposed Approach

The Proposed Algorithm: Preliminaries

- Consider $L(x, z, \lambda)$:

$$L(x, z; \lambda) = \sum_{i=1}^{N} \left( \frac{1}{N} g_i(x_i) + \langle \lambda_i, x_i - z \rangle + \frac{\eta_i}{2} \| x_i - z \|^2 \right) + g_0(z) + h(z)$$

- For variable $x_i$, let us define:

$$V_i(x_i, z; \lambda_i) = \frac{1}{N} g_i(z) + \frac{1}{N} \langle \nabla g_i(z), x_i - z \rangle + \langle \lambda_i, x_i - z \rangle + \frac{\alpha_i \eta_i}{2} \| x_i - z \|^2.$$  

an approximation of $\frac{1}{N} g_i(x_i)$

- Remarks:

  1. We have approximated $\frac{1}{N} g_i(x_i)$ use its linear approximation
  2. Parameter $\alpha_i$ gives some freedom to the algorithm.
The Proposed Approach

NESTT Algorithm

We propose a NonconvEx primal-dual SpliTTing algorithm (NESTT)

<table>
<thead>
<tr>
<th>The NESTT-G Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>From iteration $r = 1, \cdots, R$, do:</td>
</tr>
<tr>
<td>Pick $i_r \in {1, 2, \cdots N}$ with probability $p_{i_r}$ and update $(x_j, \lambda_j)$:</td>
</tr>
<tr>
<td>$x_{i_r}^{r+1} = \arg \min_{x_{i_r}} V_{i_r} (x_{i_r}, z^r, \lambda_{i_r}^r)$;</td>
</tr>
<tr>
<td>$\lambda_{i_r}^{r+1} = \lambda_{i_r}^r + \alpha_{i_r} \eta_{i_r} (x_{i_r}^{r+1} - z^r)$;</td>
</tr>
<tr>
<td>$\lambda_j^{r+1} = \lambda_j^r, \quad x_j^{r+1} = z^r, \quad \forall j \neq i_r$;</td>
</tr>
<tr>
<td>Update $z$: $z^{r+1} = \arg \min_{z \in Z} L({x_{i_r}^{r+1}}, z; \lambda^r)$.</td>
</tr>
<tr>
<td>Output: $(z^m, x^m, \lambda^m)$ where $m$ randomly picked from ${1, 2, \cdots, R}$.</td>
</tr>
</tbody>
</table>
NSETT Algorithm

- First, each node $i$ stores $(x_i, \lambda_i)$
- When the algorithm starts, the central node broadcast to everyone $z^r$
The Proposed Approach

**NSETT Algorithm**

- Randomly pick $i_r$ and update $x_{i_r}$, and related $\lambda_{i_r}$:

\[
x_{i_r}^{r+1} = \arg \min_{x_{i_r}} V_{i_r} (x_{i_r}, z^r, \lambda_{i_r}^r);
\]

\[
\lambda_{i_r}^{r+1} = \lambda_{i_r}^r + \alpha_{i_r} \eta_{i_r} (x_{i_r}^{r+1} - z^r)
\]
The Proposed Approach

NSETT Algorithm

- Aggregate \((x, y)\) to node \(z\); Update \(z\):

\[
z^{r+1} = \arg\min_{z \in Z} L(\{x_i^{r+1}\}, z; \lambda^r)
\]

![Diagram showing the aggregation process in the NSETT Algorithm](image)
Main Steps of Convergence Analysis of NESTT-G

- **Step 1)** Build a Potential Function and show that it is decreasing?

$$Q^r := \sum_{i=1}^{N} \frac{1}{N} g_i(z^r) + g_0(z^r) + h(z^r) + \sum_{i=1}^{N} \frac{3p_i\eta_i}{(\alpha_i\eta_i)^2} \left\| \frac{1}{N}(\nabla g_i(y_i^{r-1}) - \nabla g_i(z^r)) \right\|^2$$

- **Step 2)** Show that certain optimality gap is decreasing in a sublinear manner
Main Steps of Convergence Analysis of NESTT-G

- **Step 1)** Build a Potential Function and show that it is decreasing?

\[
Q^r := \sum_{i=1}^{N} \frac{1}{N} g_i(z^r) + g_0(z^r) + h(z^r) \\
+ \sum_{i=1}^{N} \frac{3p_i \eta_i}{(\alpha_i \eta_i)^2} \|1/N(\nabla g_i(y_{i}^{r-1}) - \nabla g_i(z^r))\|^2
\]

- **Step 2)** Show that certain optimality gap is decreasing in a sublinear manner
Sufficient Descent:

Lemma

Suppose Assumption A holds, and pick

\[ \alpha_i = p_i = \beta \eta_i, \quad \text{where} \quad \beta := \frac{1}{\sum_{i=1}^{N} \eta_i}, \quad \text{and} \quad \eta_i \geq \frac{9L_i}{Np_i} \]  

(3)

Then the following descent estimate holds true for NESTT

\[
\mathbb{E}[Q^r - Q^{r-1} | \mathcal{F}^{r-1}] \leq - \sum_{i=1}^{N} \frac{\eta_i}{8} \mathbb{E}_{z^r} \| z^r - z^{r-1} \|^2 \\
- \sum_{i=1}^{N} \frac{1}{\eta_i} \left\| \frac{1}{N} \left( \nabla g_i(z^{r-1}) - \nabla g_i(y_i^{r-2}) \right) \right\|^2.
\]  

(4)
The Proposed Approach

Convergence Analysis

Convergence Analysis of NESTT

- **Sublinear Convergence of NESTT:**

**Definition**

The proximal gradient of problem (1) is given by (for any $\gamma > 0$)

$$
\tilde{\nabla} f_\gamma(z) := \gamma \left( z - \text{prox}_{p+\nu Z}^\gamma \left[ z - 1/\gamma \nabla (g(z) + g_0(z)) \right] \right)
$$

where $\text{prox}_{p+\nu Z}^\gamma [u] := \arg\min_{u \in Z} p(u) + \frac{\gamma}{2} \|z - u\|^2$.

- **Optimality Gap:**

$$
\mathbb{E}[G^r] := \mathbb{E} \left[ \|\tilde{\nabla} 1/\beta f(z^r)\|^2 \right] = \frac{1}{\beta^2} \mathbb{E} \left[ \|z^r - \text{prox}_{h}^{1/\beta} [z^r - \beta \nabla (g(z^r) + g_0(z^r))]\|^2 \right].
$$

$G = 0$ if and only if a first-order stationary solution is obtained.
Convergence Rate Analysis of NESTT

Theorem (Main Theorem)

Suppose Assumption A holds, and pick (for $i = 1, \cdots, N$)

$$\alpha_i = p_i = \frac{\sqrt{L_i/N}}{\sum_{j=1}^{N} \sqrt{L_j/N}}, \eta_i = 3 \left( \sum_{j=1}^{N} \sqrt{L_j/N} \right) \sqrt{L_i/N},$$

1) Every limit point of NESTT is stationary solution of problem (2) w.p. 1.

2) $\mathbb{E}[G^m] \leq \frac{80}{3} \left( \sum_{i=1}^{N} \sqrt{L_i/N} \right)^2 \frac{\mathbb{E}[Q^0 - Q^R]}{R}$. 

Davood Hajinezhad

Optimizing Nonconvex Finite Sums by a Prox
Linear Convergence of NESTT

- Assumption B:
  B-(a) Each function $g_i(z)$ is a **quadratic function** of the form
  \[
g_i(z) = \frac{1}{2}z^T A_i z + \langle b, z \rangle
  \]
  where $A_i$ is a symmetric matrix but not necessarily positive semidefinite;
  
  B-(b) The feasible set $Z$ is a **closed compact polyhedral** set;
  
  B-(c) The nonsmooth function $p(z) = \mu \|z\|_1$, for some $\mu \geq 0$. 
The Proposed Approach

Convergence Analysis

Linear Convergence of NESTT

Theorem (Linear Convergence of NESTT)

Suppose that Assumptions A, B are satisfied. Then the sequence \( \{ \mathbb{E}[Q^{r+1}] \}_{r=1}^{\infty} \) converges \( Q \)-linearly to some \( Q^* = f(z^*) \), where \( z^* \) is a stationary solution for problem (1).

That is, there exists a finite \( \bar{r} > 0 \), \( \rho \in (0, 1) \) such that for all \( r \geq \bar{r} \),

\[
\mathbb{E}[Q^{r+1} - Q^*] < \rho \mathbb{E}[Q^r - Q^*]
\]
Connections and Comparisons with Existing Works

The NESTT can be written into the following compact form:

\[
\begin{align*}
    u^{r+1} &:= z^r - \beta \left( \frac{1}{N} \sum_{i=1}^{N} \nabla g_i(y_i^{r-1}) \right) + \frac{1}{N\alpha_i} \left( \nabla g_i(z^r) - \nabla g_i(y_i^{r-1}) \right) \\
    z^{r+1} &= \arg \min_z h(z) + g_0(z) + \frac{1}{2\beta} \| z - u^{r+1} \|^2
\end{align*}
\]

<table>
<thead>
<tr>
<th></th>
<th>$h(z)$</th>
<th>$g_0(z)$</th>
<th>$\alpha_i$</th>
<th>$p_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>IAG</strong> [Blatt 07]</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>cyclic</td>
</tr>
<tr>
<td><strong>SAG</strong> [Schmidt 12]</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1/N</td>
</tr>
<tr>
<td><strong>SAGA</strong> [Defazio 14]</td>
<td>✓</td>
<td>0</td>
<td>1/N</td>
<td>1/N</td>
</tr>
</tbody>
</table>
Comparison in terms of number of gradient evaluations

- Based on the classical theory of GD to get $\epsilon$-stationary solution GD should be run in the order of $O(L/\epsilon)$.

- Number of gradient evaluations in each step for GD is $N$.

- In the worse case ($\sum_{i=1}^{N} L_i/N = L$) number of gradient evaluations is

$$O\left(\sum_{i=1}^{N} L_i/\epsilon\right)$$
Comparison in terms of number of gradient evaluations

<table>
<thead>
<tr>
<th>Case</th>
<th>NESTT</th>
<th>GD</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( \mathcal{O}\left(\sum_{i=1}^{N} \sqrt{L_i/N}\right)^2/\epsilon )</td>
<td>( \mathcal{O}\left(\sum_{i=1}^{N} L_i/\epsilon \right) )</td>
</tr>
<tr>
<td>Case I:</td>
<td>( \mathcal{O}(N/\epsilon) )</td>
<td>( \mathcal{O}(N/\epsilon) )</td>
</tr>
<tr>
<td>( L_i = 1, \forall i )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Case II:</td>
<td>( \mathcal{O}(N^{2/3}) ): ( L_i = N^{2/3} )</td>
<td>( \mathcal{O}(N/\epsilon) )</td>
</tr>
<tr>
<td>rest: ( L_i = 1 )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Case III:</td>
<td>( \mathcal{O}(\sqrt{N}) ): ( L_i = N )</td>
<td>( \mathcal{O}(N/\epsilon) )</td>
</tr>
<tr>
<td>rest: ( L_i = 1 )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Case IV:</td>
<td>( \mathcal{O}(1) ): ( L_i = N^2 )</td>
<td>( \mathcal{O}(N/\epsilon) )</td>
</tr>
<tr>
<td>rest: ( L_i = 1 )</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
1 Introduction
   • Motivation
   • Related Works
   • Applications

2 The Proposed Approach
   • Convergence Analysis
   • Comments

3 Connections with Existing Works

4 Numerical Results
High Dimensional Regression

- Comparison of NESTT, SAGA, SGD. The $x$-axis denotes the number of passes of the dataset.
- Left: Uniform Sampling ($p_i = 1/N$) with $N_i = N_j$; Right: Non-uniform Sampling ($p_i = \frac{\sqrt{L_i/N}}{\sum_{j=1}^{N} \sqrt{L_j/N}}$) while $N_i \neq N_j$. 

![Uniform Sampling](image1.png)

![Non-Uniform Sampling](image2.png)
High Dimensional Regression

- Optimality gap $\|\tilde{\nabla}_{1/\beta} f(z^r)\|^2$ for different algorithms, with 100 passes of the datasets.

<table>
<thead>
<tr>
<th>N</th>
<th>SGD Uniform</th>
<th>SGD Non-Uni</th>
<th>NESTT-E ($\alpha = 10$) Uniform</th>
<th>NESTT-E ($\alpha = 10$) Non-Uni</th>
<th>NESTT Uniform</th>
<th>NESTT Non-Uni</th>
<th>SAGA Uniform</th>
<th>SAGA Non-Uni</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>3.4054</td>
<td>0.2265</td>
<td>2.6E-16</td>
<td>6.16E-19</td>
<td>2.3E-21</td>
<td>6.1E-24</td>
<td>2.7E-17</td>
<td>2.8022</td>
</tr>
<tr>
<td>20</td>
<td>0.6370</td>
<td>6.9087</td>
<td>2.4E-9</td>
<td>5.9E-9</td>
<td>1.2E-10</td>
<td>2.9E-11</td>
<td>7.7E-7</td>
<td>11.3435</td>
</tr>
<tr>
<td>30</td>
<td>0.2260</td>
<td>0.1639</td>
<td>3.2E-6</td>
<td>2.7E-6</td>
<td>4.5E-7</td>
<td>1.4E-7</td>
<td>2.5E-5</td>
<td>0.1253</td>
</tr>
<tr>
<td>40</td>
<td>0.0574</td>
<td>0.3193</td>
<td>5.8E-4</td>
<td>8.1E-5</td>
<td>1.8E-5</td>
<td>3.1E-5</td>
<td>4.1E-5</td>
<td>0.7385</td>
</tr>
<tr>
<td>50</td>
<td>0.0154</td>
<td>0.0409</td>
<td>8.3E-4</td>
<td>7.1E-4</td>
<td>1.2E-4</td>
<td>2.7E-4</td>
<td>2.5E-4</td>
<td>3.3187</td>
</tr>
</tbody>
</table>

- In both uniform and non-uniform case NESTT outperforms other methods.
Comparison with GD (worst case)

- Consider distributed SPCA problem
- Split a matrix into diagonal sub-matrices:
- Comparison of NESTT and GD
Numerical Results

Conclusion:

- A Primal-Dual based algorithms for minimizing nonconvex finite sum

- It converges sublinearly in general, and Q-Linearly for certain nonconvex $\ell_1$ penalized quadratic problems

- IAG/SAG/SAGA has closed connection with NESTT

- Non-uniform sampling helps to improve the convergence rates

- **Key insight**: Primal-Dual based algorithm can offer faster convergence compared with primal-only algorithms
Thank You!
Comments: Linear Convergence

- The linear convergence results utilizes a key error bound condition from [Luo-Tseng 92]

- Linear convergence to stationary solutions

- This is different from a few recent works showing linear convergence for "non-convex" problems [Zhu-Hazan 16] [Reddi et al 16] [Karimi-Schmidt 16]
  1. Based on certain quadratic growth condition
  2. Smooth unconstrained problems, every stationary point is a global minimum
  3. Do not cover our nonconvex QP
Connections and Comparisons with Existing Works

- We can verify that the $x_i$ update is given by

$$x_i^{r+1} = z^r - \frac{1}{\alpha_i \eta_i} \left( \lambda_i^r + \frac{1}{N} \nabla g_i(y_i^r) \right).$$  \hspace{1cm} (5)

- The dual update is given by:

$$\lambda_i^{r+1} = - \frac{1}{N} \nabla g_i(y_i^r)$$  \hspace{1cm} (6)

- **Observation:** The dual variables remembers the past gradients of $g_i$'s.