SEGA: Variance Reduction via Gradient Sketching

Peter Richtárik

Joint work with Filip Hanzely (KAUST) and Konstantin Mishchenko (KAUST)

Numerical Algorithms in Nonsmooth Optimization
Erwin Schrödinger International Institute for Mathematics and Physics
Vienna, February 26, 2019
SEGA: Variance Reduction via Gradient Sketching

Part of: Advances in Neural Information Processing Systems 31 (NIPS 2018)

Authors

- Filip Hanzely
- Konstantin Mishchenko
- Peter Richtarik
Optimization & Machine Learning
Group @ KAUST

Filip Hanzely

Konstantin Mishchenko
SEGA: Variance Reduction via Gradient Sketching

Filip Hanzely 1, Konstantin Mishchenko 1, Peter Richtárik 1, 2, 3

1KAUST  2University of Edinburgh  3Moscow Institute of Physics and Technology

Problem and Assumptions

Regularized Optimization

\[
\min_{x \in \mathbb{R}^d} F(x) = f(x) + R(x)
\]

- \( f: \mathbb{R} \to \mathbb{R} \) smooth & \( \mu \) strongly convex convex:
- \( f(x + h) \leq f(x) + \langle \nabla f(x), h \rangle + \frac{1}{2\mu} \|h\|^2 \)
- \( f(x) + \langle \nabla f(x), h \rangle + \frac{\mu}{2} \|h\|^2 \leq f(x + h) \)

Key Challenges:

- Design a proximal stochastic gradient-type method for solving
- How to deal with gradient sketches coming from any distribution
  \( S \): random \( n \times k \) matrix (small)
  \( D \): distribution from which \( S \) is drawn

Goal

Design a proximal stochastic gradient-type method for solving (1) using the gradient sketch oracle (2).

Simple Algorithmic Idea

\[
x^{k+1} = \text{prox}_{\alpha g^k} (x^k - \alpha g^k)
\]

How to design a good gradient estimator \( g^k \)?

Key Challenges:

- In the case when \( D \) is a distribution over standard basis vectors \( e_1, \ldots, e_n \) in \( \mathbb{R}^n \), i.e., if we have access to random partial derivatives of \( f \), then we can use
- \( g^k = \nabla f(x^k) \), and (3) reduces to proximal randomized coordinate descent (CD).

Convergence of SEGA

SEGA Estimator

- Ask oracle for a gradient sketch at \( x^k \);
- Define \( h^{k+1} \) as the closest (in some energy norm \( \|h\|_b \) \( \tilde{h} \) \( B \)), where \( B = 0 \) vector to \( h^k \) consistent with the gradient sketch:
- \( h^{k+1} = \text{arg min}_{h \in \mathbb{R}^n} \|h - h^k\|_b \), subject to \( S_i^k h = S_i^k \nabla f(x^k) \) \( k \in [1, \infty) \)
- Define the SEGA estimator:
- \( g^k = h^k + \theta_k (\nabla f(x^k) - h^k) \)

Key Property: As \( x_k \to x^* \), we get \( g^k \to 0 \), and hence SEGA estimator is variance-reduced.

Variants:

- \( \text{biasSEGA} \): use \( h^{k+1} \) instead of \( g^k \)
- \( \text{subspaceSEGA} \): If \( f(x) = \langle Ax \rangle \) for some matrix \( A \in \mathbb{R}^{n \times n} \), we can improve the SEGA estimator by exploiting the fact that \( \nabla f \) lies in range \( \langle A \rangle \).

SEGA (SkEtched GrAdient descent)

SEGA = Method (3) + SEGA estimator (5)

\( \text{biasSEGA} = \text{Method (3) + biasSEGA estimator (4)} \)

\( \text{subspaceSEGA} = \text{Method (3) + subspaceSEGA estimator} \)

SEGA with Coordinate Sketched

\[
\text{Iterates of SEGA (in 2D)}
\]

Bottom plot: \( R \) is the indicator function of the unit ball.

While CD does not converge, SEGA does!

Experiments

1. SEGA vs Random Direct Search (RDS) [2] (coordinate and Gaussian sketches) for derivative-free optimization
2. SEGA vs subspaceSEGA
3. SEGA vs Coordinate Descent (CD) [3] (left) and ADEGA vs Accelerated Coordinate Descent (ACD) [4, 5] (right) on ridge regression with \( R = 0 \)

References

Outline

1. Introduction
   a) The problem
   b) SEGA Oracle
   c) A “Gutless” Method
2. SEGA Estimator
   a) Sketch & Project
   b) Correcting for Bias
   c) Examples
3. SEGA Algorithm
   a) Variants
   b) Complexity
4. Experiments
1. Introduction
The Problem
Composite Minimization

\[ \min_{x \in \mathbb{R}^n} F(x) := f(x) + R(x) \]

Smoothness: \( f(x + h) \leq f(x) + \langle \nabla f(x), h \rangle + \frac{1}{2} \langle Lh, h \rangle \)

Strong convexity: \( f(x) + \langle \nabla f(x), h \rangle + \frac{1}{2} \langle \mu I h, h \rangle \leq f(x + h) \)

Dimension \( n \): very large

convex & closed (and not necessarily separable)
New Stochastic First-Order Oracle
New Stochastic First Order Oracle

**SkEtched GrAdient (SEGA) Oracle**

Access to a random linear transformation (i.e., “sketch”) of the gradient:

\[
S^\top \nabla f(x) = \begin{pmatrix} \langle \nabla f(x), s_1 \rangle \\ \langle \nabla f(x), s_2 \rangle \\ \vdots \\ \langle \nabla f(x), s_b \rangle \end{pmatrix} \in \mathbb{R}^b
\]

\[
S = [s_1, s_2, \ldots, s_b] \in \mathbb{R}^{n \times b}
\]

\[
S \sim \mathcal{D}
\]
Examples

1. Gaussian sketch
   \[ S = s \sim \mathcal{N}(0, \Omega) \]
   \[ S^\top \nabla f(x) = \langle \nabla f(x), s \rangle = \lim_{t \to 0} \frac{f(x + ts) - f(x)}{t} \]

2. Coordinate sketch
   \[ S = e_i \text{ with probability } p_i > 0 \]
   \[ S^\top \nabla f(x) = \langle \nabla f(x), e_i \rangle = (\nabla f(x))_i \]
A “Gutless” Method
**Proximal Stochastic Gradient Descent**

\[
\text{prox}_{\alpha R}(z) \overset{\text{def}}{=} \arg \min_{x \in \mathbb{R}^n} \left( \alpha R(x) + \frac{1}{2} \| x - z \|^2 \right)
\]

\[
x^{k+1} = \text{prox}_{\alpha R}(x^k - \alpha g^k)
\]

**Key question:**
How to construct a “good” estimator using the SkEtched GrAdient (SEGA) oracle?
2. SEGA: The Estimator
What Do We Want?
What is a “Good” Estimator?

1. Implementable given the information provided by the gradient sketch oracle

2. Unbiased

\[ \mathbb{E}_{S_k \sim \mathcal{D}} [g^k | x^k] = \nabla f(x^k) \]

3. Diminishing variance

\[ \mathbb{E} \left[ \| g^k - \nabla f(x^k) \|^2 \right] \rightarrow 0 \]
Sketch & Project
New estimator of the gradient

\[ h^{k+1} = \arg \min_{h \in \mathbb{R}^n} \| h - h^k \|^2 \]

subject to \( S_k^T h = S_k^T \nabla f(x^k) \)

Previous estimator of the gradient

Closed-form solution:

\[ h^{k+1} = h^k + Z_k (\nabla f(x^k) - h^k) \]

\[ Z_k \overset{\text{def}}{=} S_k (S_k^T S_k)^{\dagger} S_k^T \]
Sketch & Project: Visualization

The optimization problem is described by the following equations:

\[ h^{k+1} = \arg \min_{h \in \mathbb{R}^n} \| h - h^k \|^2 \]

subject to \( S_k^\top h = S_k^\top \nabla f(x^k) \)

The set \( \mathcal{L}_k \) is defined as:

\[ \mathcal{L}_k \overset{\text{def}}{=} \left\{ h \in \mathbb{R}^n \mid S_k^\top h = S_k^\top \nabla f(x^k) \right\} = \nabla f(x^k) + \text{Null}(S_k^\top) \]
**Lemma**  For any $v \in \mathbb{R}^n$

$$
\mathbb{E}_D \left[ \| h^{k+1} - v \|_I^2 \right] = \| h^k - v \|^2_{I - \mathbb{E}_D[Z]} + \| \nabla f(x^k) - v \|^2_{\mathbb{E}_D[Z]}
$$
Original sketch and project

- Robert Mansel Gower and P.R. Mansel Gower
  *Randomized Iterative Methods for Linear Systems*

Removal of full rank assumption + duality

- Robert Mansel Gower and P.R.
  *Stochastic Dual Ascent for Solving Linear Systems*

Inverting matrices & connection to quasi-Newton updates

- Robert Mansel Gower and P.R.
  *Randomized Quasi-Newton Methods are Linearly Convergent Matrix Inversion Algorithms*
  *SIAM J. on Matrix Analysis and Applications* 38(4), 1380-1409, 2017

Computing the pseudoinverse

- Robert Mansel Gower and P.R.
  *Linearly Convergent Randomized Iterative Methods for Computing the Pseudoinverse*

Application to machine learning

- Robert Mansel Gower, Donald Goldfarb and P.R.
  *Stochastic Block BFGS: Squeezing More Curvature out of Data*
  *ICML* 2016

Sketch and project revisited: stochastic reformulations of linear systems

- P.R. and Martin Takáč
  *Stochastic Reformulations of Linear Systems: Algorithms and Convergence Theory*
  *arXiv:1706.01108*, 2017

New understanding of Quasi-Newton Rules

- 2017 IMA Fox Prize (2nd Prize) in Numerical Analysis
- Most downloaded SIMAX paper (2017)

My course from last week
Sketch and Project II

Linear convergence of the stochastic heavy ball method

Nicolas Loizou and P.R.
*Momentum and Stochastic Momentum for Stochastic Gradient, Newton, Proximal Point and Subspace Descent Methods*
*arXiv:1712.09677*, 2017

Stochastic projection methods for convex feasibility

Ion Necoara, Andrei Patrascu and P.R.
*Randomized Projection Methods for Convex Feasibility Problems: Conditioning and Convergence Rates*
*arXiv:1801.04873*, 2018

Stochastic spectral & conjugate descent

Dmitry Kovalev, Eduard Gorbunov, Elnur Gasanov and P.R.
*Stochastic Spectral and Conjugate Descent Methods*
*NeurIPS 2018*

Accelerated stochastic matrix inversion

Robert Mansel Gower, Filip Hanzely, P.R. and Sebastian Stich
*Accelerated Stochastic Matrix Inversion: General Theory and Speeding up BFGS Rules for Faster Second-Order Optimization*
*NeurIPS 2018*

SAGD: a “strange” special case of JacSketch

Adel Bibi, Alibek Sailanbayev, Bernard Ghanem, Robert Mansel Gower and P.R.
*Improving SAGA via a Probabilistic Interpolation with Gradient Descent*
*arXiv:1806.05633*, 2018

Extension to Convex Feasibility

Acceleration
Unbiasedness: SEGA for Coordinate Sketches
2D Example

\[
S = \begin{cases} 
  e_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} & \text{with probability } p_1 \in (0, 1) \\
  e_2 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} & \text{with probability } p_2 = 1 - p_1
\end{cases}
\]

\[
S_k = e_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \implies \mathcal{L}_k = \{ h \in \mathbb{R}^2 \mid h_1 = (\nabla f(x^k))_1 \}
\]

\[
S_k = e_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \implies \mathcal{L}_k = \{ h \in \mathbb{R}^2 \mid h_2 = (\nabla f(x^k))_2 \}
\]
Case 1

\[ p_1 = \frac{1}{2} \quad p_2 = \frac{1}{2} \]
\begin{align*}
p_1 &= \frac{1}{2}, \quad p_2 = \frac{1}{2} \\
g^k &= h^k + 2(h^{k+1} - h^k) \\
\mathcal{L}_k &= \{h \in \mathbb{R}^2 \mid h_1 = (\nabla f(x^k))_1 \} \\
\mathcal{L}_k &= \{h \in \mathbb{R}^2 \mid h_2 = (\nabla f(x^k))_2 \} \\
g^k &= h^k + 2(h^{k+1} - h^k)
\end{align*}
Case 2

\[ p_1 = \frac{2}{3} \quad p_2 = \frac{1}{3} \]
\[ p_1 = \frac{2}{3}, \quad p_2 = \frac{1}{3} \]

\[ g^k = h^k + \frac{3}{2}(h^{k+1} - h^k) \]

\[ \mathcal{L}_k = \{ h \in \mathbb{R}^2 \mid h_1 = (\nabla f(x^k))_1 \} \]

\[ \mathcal{L}_k = \{ h \in \mathbb{R}^2 \mid h_2 = (\nabla f(x^k))_2 \} \]

\[ g^k = h^k + 3(h^{k+1} - h^k) \]
SEGA for General Sketches
SEGA Estimator

SEGA estimator

\( g^k \) \( \overset{\text{def}}{=} h^k + \theta_k (h^{k+1} - h^k) \)

\( = h^k + \theta_k Z_k (\nabla f(x^k) - h^k) \)

Bias correcting random variable

\( \mathbb{E} [\theta_k Z_k] = I \)

\[ \mathbb{E}_{D} [g^k] = \nabla f(x^k) \]
3. SEGA: The Algorithm
The Algorithm
The SEGA Algorithm

**Step 0** Choose \( x^0, h^0 \in \text{dom}F \)

For \( k \geq 0 \) **REPEAT**

**Step 1** Ask SEGA Oracle for \( S_k^T \nabla f(x^k) \)
Perform Sketch & Project
\[
    h^{k+1} = \arg \min_{h \in \mathbb{R}^n} \| h - h^k \|^2 \\
    \text{subject to } S_k^T h = S_k^T \nabla f(x^k)
\]

**Step 2** Compute the SEGA Estimator
\[
g^k = h^k + \theta_k(h^{k+1} - h^k)
\]

**Step 3** Perform **Proximal SGD** step
\[
    x^{k+1} = \text{prox}_{\alpha R}(x^k - \alpha g^k)
\]

\[\min_{x \in \mathbb{R}^n} F(x) := f(x) + R(x)\]
Variants of SEGA

1. SEGA

\[ g^k = h^k + \theta_k (h^{k+1} - h^k) \]

2. Biased SEGA

Use \( \theta_k \equiv 1 \)

\[ g^k = h^{k+1} \]

3. Subspace SEGA

\[ f(x) = \phi(Ax) \quad \nabla f(x) \in \text{Range}(A^\top) \]

\[ h^{k+1} = \arg \min_{h \in \mathbb{R}^n} \| h - h^k \|^2 \]

subject to \( S_k^\top h = S_k^\top \nabla f(x^k) \)

\[ h \in \text{Range}(A^\top) \]

4. Accelerated SEGA

\[ x^{k+1} = \text{prox}_{\alpha R}(x^k - \alpha g^k) \]

\[ \mathbb{E}_D[\theta_k Z_k] = \mathbf{I} \]
Complexity:
General Sketch
Complexity for General Sketches

Theorem

\[ \mathbb{E} \left[ \Phi^k \right] \leq (1 - \alpha \mu)^k \Phi^0 \]

Lyapunov function:

\[ x^0, h^0 \in \text{dom} \mathcal{F} \]

\[ \Phi^k \overset{\text{def}}{=} \| x^k - x^* \|^2 + \sigma \alpha \| h^k - \nabla f(x^*) \|^2 \]

Strong convexity:

\[ f(x) + \langle \nabla f(x), h \rangle + \frac{\mu}{2} \| h \|^2 \leq f(x + h) \]

Stepsize can’t be too large:

\[ \alpha (2(C - I) + \sigma \mu I) \leq \sigma \mathbb{E}_{S \sim \mathcal{D}} [Z] \]

\[ 2\alpha C + \sigma \mathbb{E}_{S \sim \mathcal{D}} [Z] \leq L^{-1} \]

\[ C \overset{\text{def}}{=} \mathbb{E}_{S \sim \mathcal{D}} [\theta^2 Z] \]
Complexity:
Coordinate Sketch
Coordinate Sketch: Arbitrary Sampling Setup

Random subset of \{1, \ldots, n\}

- \( S = I_C \) (random column submatrix of the identity matrix)
- Probability vector \( p \in \mathbb{R}^n: p_i \overset{\text{def}}{=} \text{Prob}(i \in C) \)
- Probability matrix \( P \in \mathbb{R}^{n \times n}: P_{ij} \overset{\text{def}}{=} \text{Prob}(i \in C \& j \in C) \)
- ESO vector \( v \in \mathbb{R}^n \) (for mini-batching) defined by:

\[
P \cdot M \preceq \text{Diag}(p \cdot v)
\]
### Complexity Results

\[ R \equiv 0 \]

<table>
<thead>
<tr>
<th>Method</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>SEGA</td>
<td>[ 8.55 \cdot \frac{\text{Tr}(L)}{\mu} \log \frac{1}{\epsilon} ]</td>
</tr>
<tr>
<td>importance sampling</td>
<td></td>
</tr>
<tr>
<td>SEGA</td>
<td>[ 8.55 \cdot \left( \max_i \frac{v_i}{p_i \mu} \right) \log \frac{1}{\epsilon} ]</td>
</tr>
<tr>
<td>arbitrary sampling</td>
<td></td>
</tr>
<tr>
<td>ASEGA</td>
<td>[ 9.8 \cdot \frac{\sum_i \sqrt{L_{ii}}}{\sqrt{\mu}} \log \frac{1}{\epsilon} ]</td>
</tr>
<tr>
<td>importance sampling</td>
<td></td>
</tr>
<tr>
<td>ASEGA</td>
<td>[ 9.8 \cdot \sqrt{\max_i \frac{v_i}{p_i^2 \mu}} \log \frac{1}{\epsilon} ]</td>
</tr>
<tr>
<td>arbitrary sampling</td>
<td></td>
</tr>
</tbody>
</table>

Up to the constant factors 8.55 and 9.5, these rates are exactly the same as the rates of CD [R. & Takáč ‘16] and accelerated CD [Allen-Zhu et al ‘16, Hanzely & R. ‘19].
Coordinate Descent

P.R. and Martin Takáč
On optimal probabilities in stochastic coordinate descent methods
*Optimization Letters* 10(6), 1223-1243, 2016

Zeyuan Allen-Zhu, Zheng Qu, P.R. and Yang Yuan
Even faster accelerated coordinate descent using non-uniform sampling
*ICML* 2016

Filip Hanzely and P.R.
Accelerated coordinate descent with arbitrary sampling and best rates for minibatches
*AISTATS* 2019
4. Experiments
Illustration in 2D
SEGA vs
Projected Gradient Descent
Gaussian Sketch, Ball Constraint

\[ S = \text{Gaussian vector} \quad R(x) = 1_{\mathcal{B}(0, 1)}(x) \]
SEGA vs Subspace SEGA
SEGA vs Subspace SEGA

\[ f(x) = \phi(Ax) \quad \nabla f(x) \in \text{Range}(A^\top) \]

\[ n = 1,000 \]
SEGA vs Random Direct Search

El Houcine Bergou, Eduard Gorbunov and P.R.

Stochastic three points method for unconstrained smooth minimization

arXiv:1902.03591, 2019
SEGA vs Coordinate Descent
SEGA vs CD

![Graph showing the comparison between CD and SEGA](image-url)
Accelerated SEGA vs Accelerated CD

![Graph comparing Accelerated SEGA (ASEGA) and Accelerated CD (ACD)]
5. Summary
Summary

• New Stochastic First-Order Oracle: SkEtched GrAdient (SEGA)

• New Stochastic Proximal SGD method. Comes in several variants:
  – SEGA (based on the SEGA Estimator)
  – Biased SEGA
  – Subspace SEGA
  – Accelerated SEGA

• Coordinate sketches:
  – Same complexity as state-of-the art CD methods
  – Can handle non-separable regularizer $R$
The End