SARAH: StochAstic Recursive grAdient algoritHm

Pang-Chun Chung, Hung-Ruei Wu

Department of Computer Science, National Yang Ming Chiao Tung University, Hsinchu, Taiwan

Motivation

- **Limitation of SVRG:** In the inner loop, SVRG only uses the true gradient as a historical reference. This limited information leads to *unstable updates* within the inner loop.
- **Limitation of SAG/SAGA:** SAG and SAGA utilize all historical gradient information, resulting in *high storage requirements*, which can be impractical for large-scale problems.
- **Proposed Solution - SARAH:** To address these limitations, we propose a *recursive update scheme* in SARAH. Our method combines the advantages of SVRG and SAG/SAGA by:
  - ▹ Ensuring **stability** through recursive updates within the inner loop.
  - ▹ Avoiding excessive **storage costs** while maintaining efficiency.

Finite-Sum Optimization Problem

$$\min_{w \in \mathbb{R}^d} P(w) := \frac{1}{n} \sum_{i \in [n]} f_i(w),$$

where f_i is convex with a Lipschitz continuous gradient, $i \in [n] := \{1, \dots, n\}$, and we assume that the optimal solution w^* exists.

Remark: This optimization problem forms the foundation for many algorithms, including SVRG and our proposed SARAH.

SARAH: Stochastic Recursive Gradient Algorithm

Input: Learning rate $\eta > 0$, inner loop size m, initial point w_0
Output: Optimized solution w_s

Algorithm 1: SARAH

Initialize $\tilde{w}_0$
For $s = 1, 2, \dots$
  $w_0 \leftarrow \tilde{w}_{s-1}$
  $v_0 \leftarrow \frac{1}{n} \sum_{i=1}^{n} \nabla f_i(w_0)$
  $w_1 \leftarrow w_0 - \eta v_0$
  For $t = 1, \dots, m-1$
    Sample i_t uniformly at random from $[n]$
    $v_t \leftarrow \nabla f_{i_t}(w_t) - \nabla f_{i_t}(w_{t-1}) + v_{t-1}$
    $w_{t+1} \leftarrow w_t - \eta v_t$
  End For
  Set $\tilde{w}_s \leftarrow w_t$ with t chosen uniformly at random from $\{0, 1, \dots, m\}$
End For

Assumptions

- **Assumption 1 (L-smooth).** Each $f_i : \mathbb{R}^d \to \mathbb{R}$, $i \in [n]$, is L-smooth, i.e.,
$$\|\nabla f_i(w) - \nabla f_i(w')\| \le L\|w - w'\|, \quad \forall w, w' \in \mathbb{R}^d.$$
- **Assumption 2a (μ-strongly convex).** The function $P : \mathbb{R}^d \to \mathbb{R}$ is μ-strongly convex, i.e.,
$$P(w) \ge P(w') + \nabla P(w')^T (w - w') + \frac{\mu}{2}\|w - w'\|^2.$$
- **Assumption 2b.** Each $f_i : \mathbb{R}^d \to \mathbb{R}$, $i \in [n]$, is strongly convex with $\mu > 0$. Strong convexity implies:
$$2\mu[P(w) - P(w^*)] \le \|\nabla P(w)\|^2, \quad \forall w \in \mathbb{R}^d.$$
- **Assumption 3 (Convexity).** Each $f_i : \mathbb{R}^d \to \mathbb{R}$, $i \in [n]$, is convex, i.e.,
$$f_i(w) \ge f_i(w') + \nabla f_i(w')^T (w - w'), \quad \forall i \in [n].$$

Performance Metrics

- **ϵ-accurate Solution**: The solution w_T satisfies $\|\nabla P(w_T)\|^2 \le \epsilon$, ensuring the gradient norm meets the desired accuracy level.
- **Full Gradient Calculation**: The analysis is based on the total number of stochastic gradient evaluations, bounding the iterations required to achieve the specified accuracy.

Comparison of SARAH and SVRG on Convex Problems

Method	Problem Type	Complexity
SARAH	Strongly Convex	$O((n + \kappa)\log(1/\epsilon))$
SVRG	Strongly Convex	$O((n + \kappa)\log(1/\epsilon))$
SARAH	General Convex	$O((n + 1/\epsilon)\log(1/\epsilon))$
SVRG	General Convex	$O(n + (n/\epsilon))$

SARAH+: A Practical Variant of SARAH

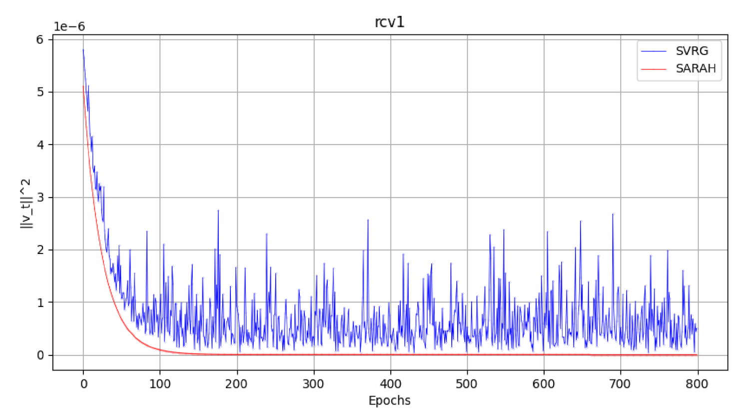
Input: Learning rate $\eta > 0$, $0 < \gamma \le 1$, and the maximum inner loop size m
Output: Optimized solution $\tilde{w}_s$

Algorithm 2: SARAH+

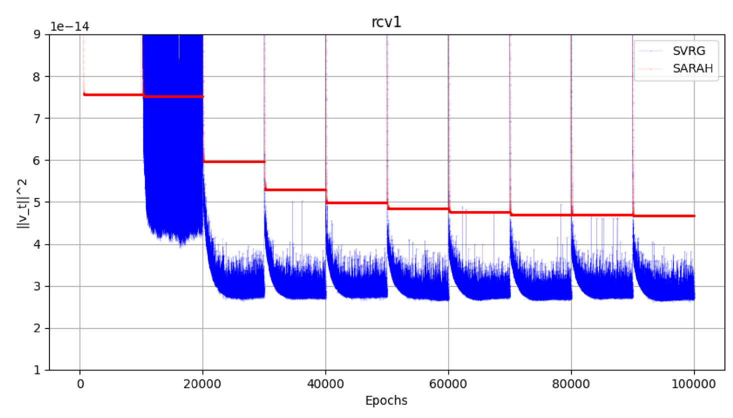
Initialize $\tilde{w}_0$
For $s = 1, 2, \dots$
  $w_0 \leftarrow \tilde{w}_{s-1}$
  $v_0 \leftarrow \frac{1}{n} \sum_{i=1}^{n} \nabla f_i(w_0)$
  $w_1 \leftarrow w_0 - \eta v_0$
  $t \leftarrow 1$
  While $\|v_{t-1}\|^2 > \gamma\|v_0\|^2$ and $t < m$
    Sample i_t uniformly at random from $[n]$
    $v_t \leftarrow \nabla f_{i_t}(w_t) - \nabla f_{i_t}(w_{t-1}) + v_{t-1}$
    $w_{t+1} \leftarrow w_t - \eta v_t$
    $t \leftarrow t + 1$
  End While
  Set $\tilde{w}_s \leftarrow w_t$
End For

Numerical Result I

The following figures show the stochastic gradient of SVRG and SARAH.



(a) Single Inner Loop



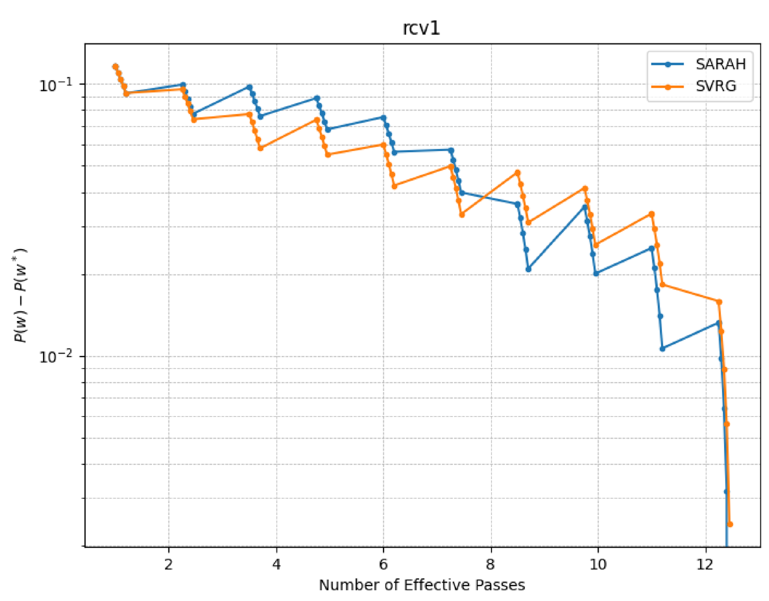
(b) Multiple Inner Loop

Key Observations:

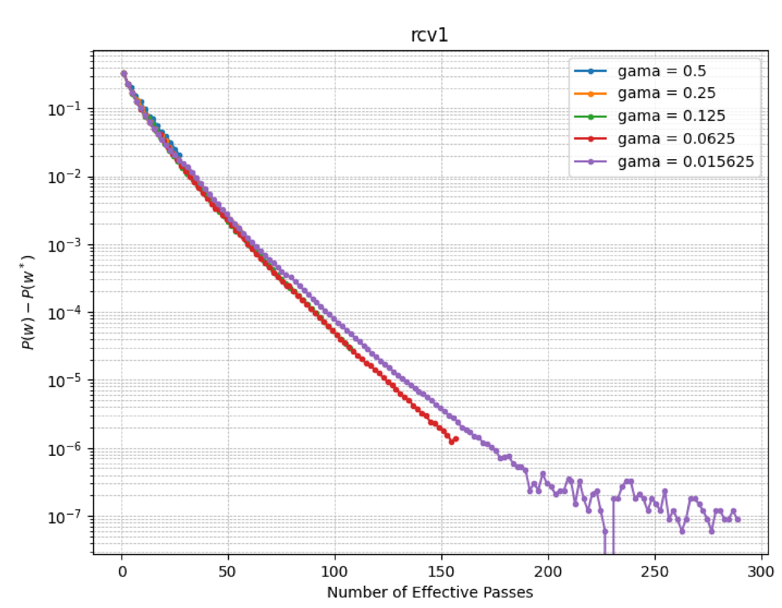
- SVRG shows oscillations due to large fluctuations in stochastic gradients.
- SARAH is more stable and converges smoothly.

Numerical Result II

The figure below compares the performance of SVRG and SARAH in terms of loss residuals.



(a) Comparison of SVRG and SARAH



(b) Comparison SARAH+ with different γ

Key Observations:

- **Left Figure (SVRG and SARAH):** As mentioned in the paper, SARAH initially does not perform as well as SVRG. However, with sufficient effective passes, SARAH slightly outperforms SVRG in terms of convergence stability and final accuracy.
- **Right Figure (SARAH+ with gamma):** The performance of SARAH+ depends on the choice of gamma values. The most appropriate range for γ appears to be approximately between $\frac{1}{8}$ and $\frac{1}{16}$.

Conclusion

- A new **variance reducing** algorithm has been proposed.
- The algorithm achieves a **linear convergence** rate for the inner loop.
- The convergence rate's **constant is smaller**, improving efficiency.
- SARAH+ provides an explicit **stopping criterion** for practical implementations.

References

1. Johnson, Rie, and Tong Zhang, *"Accelerating Stochastic Gradient Descent Using Predictive Variance Reduction,"* NeurIPS, 2013.
2. Nguyen, Lam M., Jie Liu, Katya Scheinberg, and Martin Takáč, *"SARAH: A Novel Method for Machine Learning Problems Using Stochastic Recursive Gradient,"* ICML, 2017.

{x85432, caco}.sc11@nycu.edu.tw