



DISTRIBUTED OPTIMIZATION WITH UNBOUNDED DELAYS

Panayotis Mertikopoulos¹

joint with

Zhengyuan Zhou² Nick Bambos³ Peter Glynn³ Yinyu Ye³

¹CNRS / LIG / POLARIS

²NYU Stern

³Stanford University

WAX workshop – May 27, 2021



Distributed computing

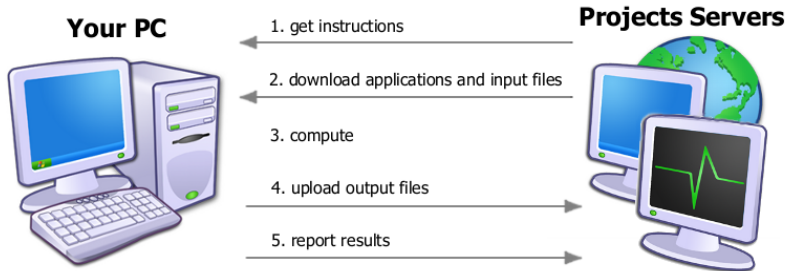
The screenshot displays a distributed computing interface for protein folding. The central element is a world map where numerous locations are marked with small spheres in various colors (grey, red, blue, yellow), representing the global distribution of computing resources. The interface includes several informational panels:

- Current Work Unit:**
 - Project: 9007 Run: 528 Clone: 3 Gen: 56
 - FahCore: GRO_A4 Uxa4
 - Progress:
 - Time Left: 1.08 days
- Donor:**
 - Name: J_M_Ward
 - Team: 026448
- Status:**
 - Snapshots: 1.1 of 2
 - Connection: Connected
 - Protein: Live
 - Slot: 1

Additional elements include a 'Folding@home' logo with a map of France, the Stanford School of Medicine logo, and small icons for a camera and user avatars on the right side of the map.



Distributed computing, cont'd





Stochastic/Distributed optimization

Stochastic optimization:

$$\begin{array}{ll} \text{minimize} & f(x) = \mathbb{E}[F(x; \omega)] \\ \text{subject to} & x \in \mathcal{X} \end{array} \quad (\text{Opt})$$

where:

- ▶ \mathcal{X} is a closed, convex subset of \mathbb{R}^d
- ▶ $F(x; \omega)$ is a random function (depending on ω)



Stochastic/Distributed optimization

Stochastic optimization:

$$\begin{array}{ll} \text{minimize} & f(x) = \mathbb{E}[F(x; \omega)] \\ \text{subject to} & x \in \mathcal{X} \end{array} \quad (\text{Opt})$$

where:

- ▶ \mathcal{X} is a closed, convex subset of \mathbb{R}^d
- ▶ $F(x; \omega)$ is a random function (depending on ω)

$$\text{Distributed optimization: } f(x) = \frac{1}{N} \sum_{i=1}^N f_i(x) \quad [F(x; \omega_i) = f_i(x)]$$



Stochastic/Distributed optimization

Stochastic optimization:

$$\begin{array}{ll} \text{minimize} & f(x) = \mathbb{E}[F(x; \omega)] \\ \text{subject to} & x \in \mathcal{X} \end{array} \quad (\text{Opt})$$

where:

- ▶ \mathcal{X} is a closed, convex subset of \mathbb{R}^d
- ▶ $F(x; \omega)$ is a random function (depending on ω)

$$\text{Distributed optimization: } f(x) = \frac{1}{N} \sum_{i=1}^N f_i(x) \quad [F(x; \omega_i) = f_i(x)]$$

Key challenges:

- ▶ Computation of expectation/sum impossible
- ▶ Gradients of f inaccessible but gradients of F cheap



The master-worker architecture

Master-Worker architecture

Require: Master and workers $i = 1, \dots, N$

1: **repeat**

2: **Master:**

 (a) Receive gradient update from worker i

 (b) Update state

 (c) Send updated state to worker i

3: **Workers:**

 (a) Receive state

 (b) Compute i.i.d. gradient

 (c) Send gradient update to master

4: **until** end



Stochastic gradient descent

Distributed asynchronous stochastic gradient descent (DASGD)

Require: Initial state $Y_0 \in \mathbb{R}^d$, step-size sequence γ_n

1: $n \leftarrow 0$

2: **repeat**

3: $X_n = \Pi(Y_n)$

4: $Y_{n+1} = Y_n - \gamma_{n+1} \nabla F(X_{s_n}; \omega_{s_{n+1}})$

5: $n \leftarrow n + 1$

6: **until** end

7: **return** solution candidate X_n

Glossary:

- ▶ n : global counter
- ▶ s_n : iteration from which gradient of round n originates
- ▶ $d_{s_n} = n - s_n$: delay of iteration s_n (could become huge over time)
- ▶ X_n : global state
- ▶ Π : projection to \mathcal{X}



State of the art

What is known:

- ▶ **Sublinear delays, convex, deterministic:** convergence if $d_n = o(n)$
[Bertsekas & Tsitsiklis, 1997]
- ▶ **Bounded delays, convex, stochastic:** mean convergence if $\sup_n d_n < \infty$
[Agarwal & Duchi, 2011; Recht et al., 2011; Mania et al., 2017;...]
- ▶ **Bounded delays, nonconvex, stochastic:** mean gradient gap if $\sup_n d_n < \infty$
[Lian et al., 2015]

Limitations:

- ▶ High probability / almost sure convergence?
- ▶ Large delays?
- ▶ Convergence in nonconvex objectives?



Assumptions

Assumption (Regularity)

1. **Differentiability:** $F(x; \omega)$ is differentiable in x for \mathbb{P} -almost all ω
2. **Bounded second moments:** $\mathbb{E}[\|\nabla F(x; \omega)\|^2] < \infty$
3. **Lipschitz smoothness:** $\nabla f(x) = \mathbb{E}[\nabla F(x; \omega)]$ is Lipschitz on \mathcal{X}

Assumption (Variational Coherence)

F is variationally coherent in the mean:

$$\mathbb{E}[\langle \nabla f(x; \omega), x - x^* \rangle] > 0 \quad (\text{VC})$$

for all $x^* \in \arg \min f$ and all $x \notin \arg \min f$.



Our results

The method's step-size plays a crucial role:

1. **Bounded delays:** $\sup_n d_n < \infty \rightsquigarrow \sum_{n=1}^{\infty} \gamma_n^2 < \infty, \sum_{n=1}^{\infty} \gamma_n = \infty$
2. **Linearly growing delays:** $d_n = \mathcal{O}(n) \rightsquigarrow \gamma_n \propto 1/(n \log n)$
3. **Polynomially growing delays:** $d_n = \mathcal{O}(n^q) \rightsquigarrow \gamma_n \propto 1/(n \log n \log \log n)$



Our results

The method's step-size plays a crucial role:

1. **Bounded delays:** $\sup_n d_n < \infty \rightsquigarrow \sum_{n=1}^{\infty} \gamma_n^2 < \infty, \sum_{n=1}^{\infty} \gamma_n = \infty$
2. **Linearly growing delays:** $d_n = \mathcal{O}(n) \rightsquigarrow \gamma_n \propto 1/(n \log n)$
3. **Polynomially growing delays:** $d_n = \mathcal{O}(n^q) \rightsquigarrow \gamma_n \propto 1/(n \log n \log \log n)$

Theorem (Zhou, M, Bambos, Glynn & Ye, ICML 2018; coherent)

- ▶ **Assume:** **A1** (regularity); **A2** (coherence)
- ▶ **Then:** DASGD converges to a solution of (Opt) with probability 1



Our results

The method's step-size plays a crucial role:

1. **Bounded delays:** $\sup_n d_n < \infty \rightsquigarrow \sum_{n=1}^{\infty} \gamma_n^2 < \infty, \sum_{n=1}^{\infty} \gamma_n = \infty$
2. **Linearly growing delays:** $d_n = \mathcal{O}(n) \rightsquigarrow \gamma_n \propto 1/(n \log n)$
3. **Polynomially growing delays:** $d_n = \mathcal{O}(n^q) \rightsquigarrow \gamma_n \propto 1/(n \log n \log \log n)$

Theorem (Zhou, M, Bambos, Glynn & Ye, ICML 2018; coherent)

- ▶ **Assume:** **A1** (regularity); **A2** (coherence)
- ▶ **Then:** DASGD converges to a solution of (Opt) with probability 1

Theorem (Zhou, M, Bambos, Glynn & Ye, MOR 2021; non-convex)

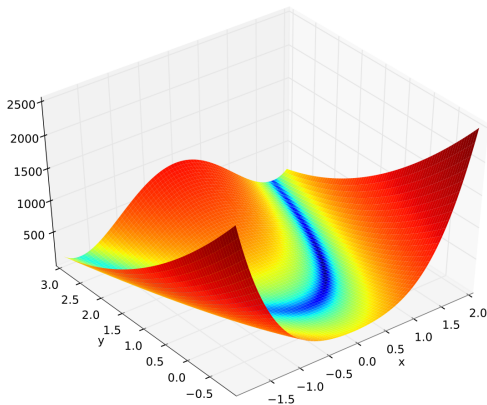
- ▶ **Assume:** **A1** (regularity)
- ▶ **Then:** DASGD converges to a solution of (Opt) with probability 1



Numerical experiments

Test: Rosenbrock benchmark with $d = 101$ degrees of freedom

$$f(x) = \sum_{i=1}^{100} [100(x_{i+1} - x_i^2)^2 + (1 - x_i)^2],$$

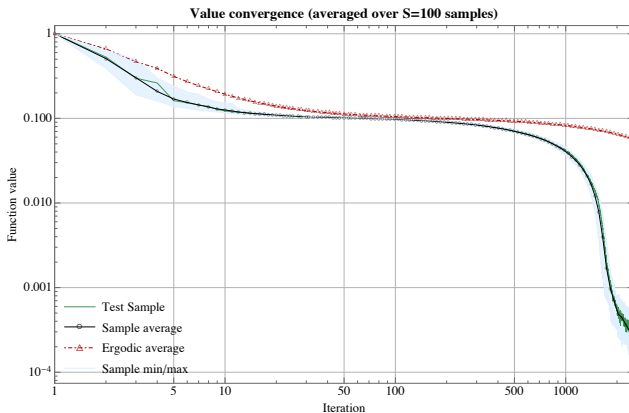




Numerical experiments

Test: Rosenbrock benchmark with $d = 101$ degrees of freedom

$$f(x) = \sum_{i=1}^{100} [100(x_{i+1} - x_i^2)^2 + (1 - x_i)^2],$$





Numerical experiments

Test: Rosenbrock benchmark with $d = 101$ degrees of freedom

$$f(x) = \sum_{i=1}^{100} [100(x_{i+1} - x_i^2)^2 + (1 - x_i)^2],$$

