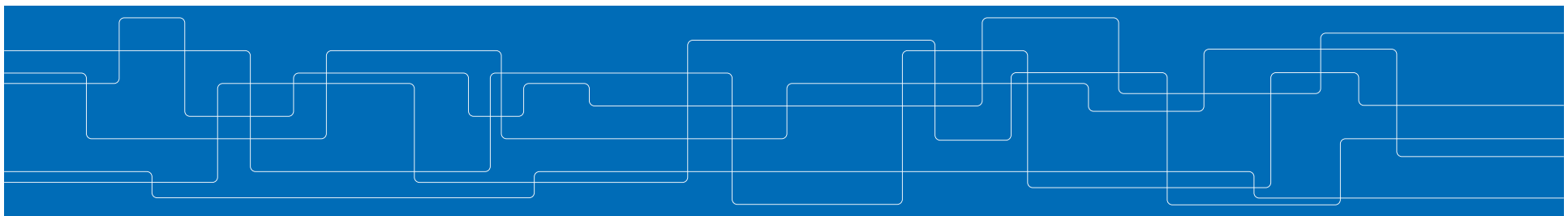
Analysis and Implementation of an Asynchronous Optimization Algorithm for the Parameter Server

Arda Aytekin [1] Hamid Reza Feyzmahdavian [2] Mikael Johansson [1]

[1]KTH Royal Institute of Technology

[2]ABB Corporate Research

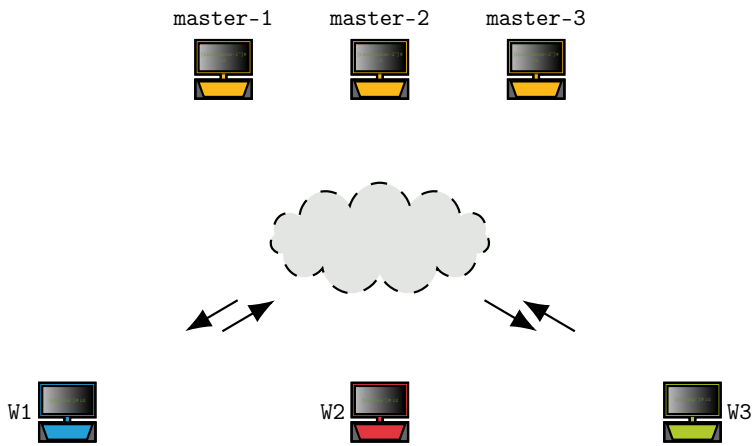
LCCC Focus Period on Large-Scale and Distributed Optimization



Topic of the Talk

Asynchronous, incremental aggregated gradient descent method

- Analysis
  - two important lemmas
  - **linear convergence** rate for **composite optimization** problems
- Implementation
  - representative real-world problems on **parameter server**



Emergence of "Big Data"

Big Data information set too large to handle

- Adoption of **cloud**-based technologies in engineering
  - collect, share and process **large** volumes of data
  - data **scattered** among different locations

Emergence of "Big Data"

Big Data information set too large to handle

- Adoption of **cloud**-based technologies in engineering
  - collect, share and process **large** volumes of data
  - data **scattered** among different locations
- Remarkable examples
  - `Google Books Ngrams`: 2 terabytes
  - `NASA NEX`: 10 terabytes
  - `1000 Genomes`: 300 terabytes

Emergence of "Big Data"

Big Data information set too large to handle

- Adoption of **cloud**-based technologies in engineering
  - collect, share and process **large** volumes of data
  - data **scattered** among different locations
- Remarkable examples
  - `Google Books Ngrams`: 2 terabytes
  - `NASA NEX`: **10 terabytes**
  - `1000 Genomes`: 300 terabytes

4 days to download, 18 hours to read

Solution?

Parallel Programming

Concepts

Split the problem into parts, solve them separately

- **faster** results
- **bigger** problems

Parallel Programming

Concepts

Split the problem into parts, solve them separately

- **faster** results
- **bigger** problems

Speedup	relative improvement in execution time (3x)
Efficiency	relative speedup per worker (3x / 4)
Scalability	maintainability of efficiency with increasing sizes (3x / 10)

Parallel Programming

Concepts

Split the problem into parts, solve them separately

- **faster** results
- **bigger** problems

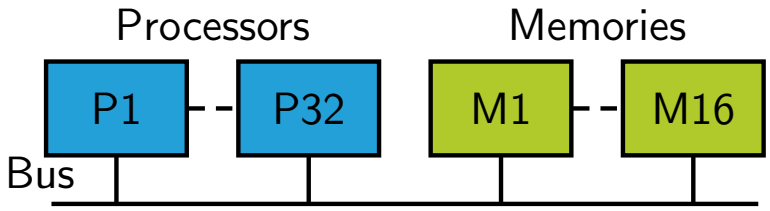
Speedup	relative improvement in execution time (3x)
Efficiency	relative speedup per worker (3x / 4)
Scalability	maintainability of efficiency with increasing sizes (3x / 10)

Parallel optimization

- rich history (D. Bertsekas and Tsitsiklis 1989; Censor and Zenios 1997)
- still important to tailor algorithms for **modern** architectures

Parallel Programming

Shared Memory



Advantages

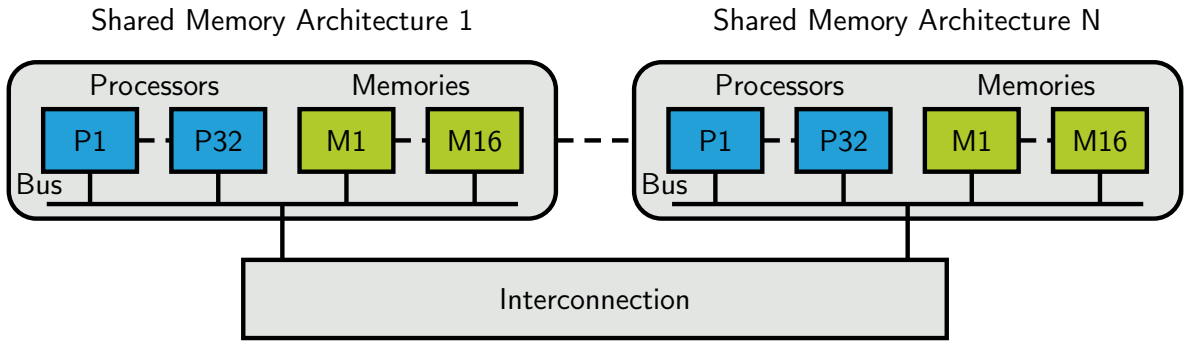
- Built-in support
- Relatively easy to program
- Cost effective when small-sized

Disadvantages

- Bus limitations
- Nonuniform memory access
- Cache coherency

Parallel Programming

Distributed Memory



Advantages

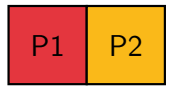
- Scalability
- Sometimes the only option

Disadvantages

- Significant delays
- Explicit coordination
- Third-party libraries

Parallel Programming

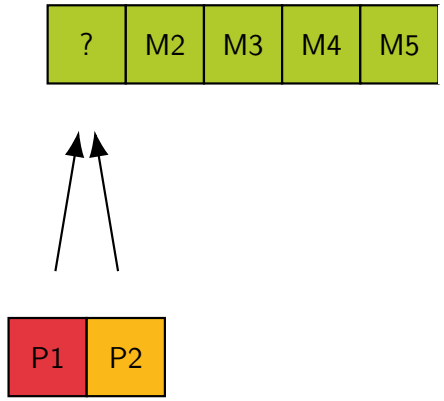
Synchronization



Some resource, *e.g.*, memory, is shared between P1 and P2.

Parallel Programming

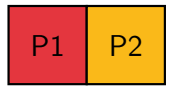
Synchronization



Both want to change; result unknown.

Parallel Programming

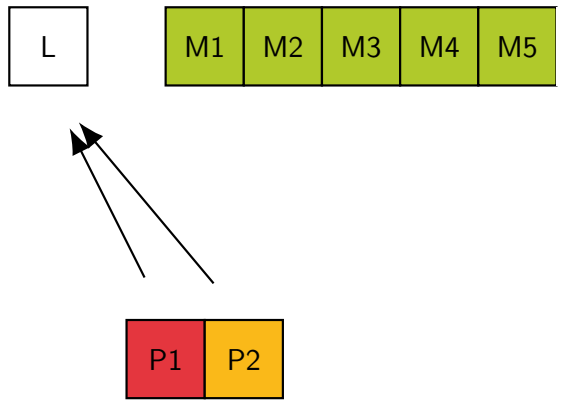
Synchronization



Define lock to control write access.

Parallel Programming

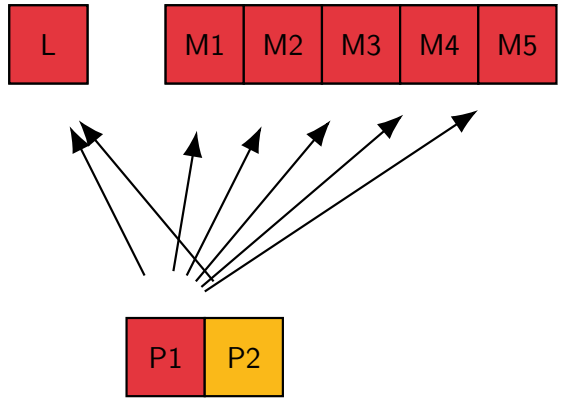
Synchronization



Both want to obtain lock.

Parallel Programming

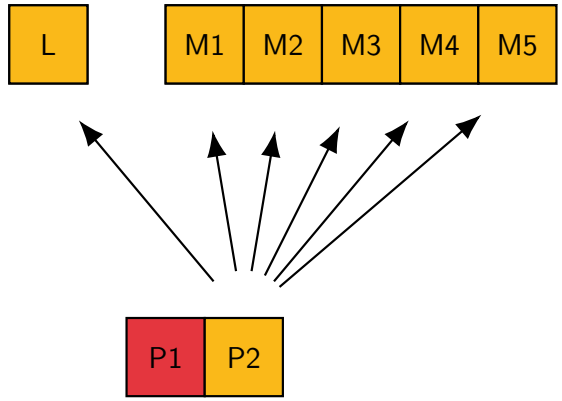
Synchronization



P1 wins, P1 has write access.

Parallel Programming

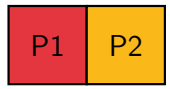
Synchronization



P1 finishes, P2 can write.

Parallel Programming

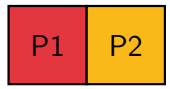
Synchronization



Memory available for another round.

Parallel Programming

Synchronization

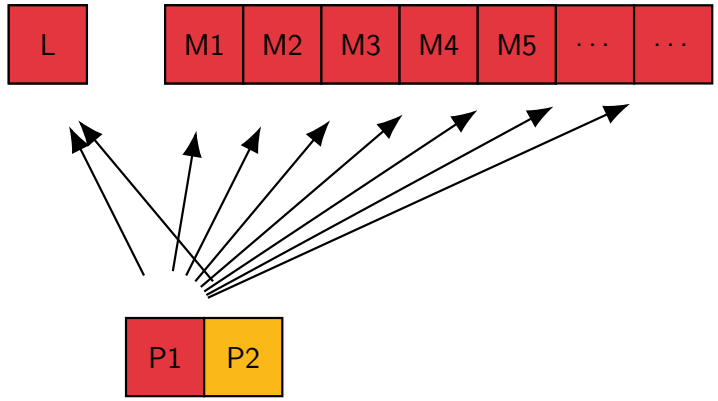


Optimization algorithms traditionally synchronous

- running at the pace of **slowest** processor

Parallel Programming

Synchronization

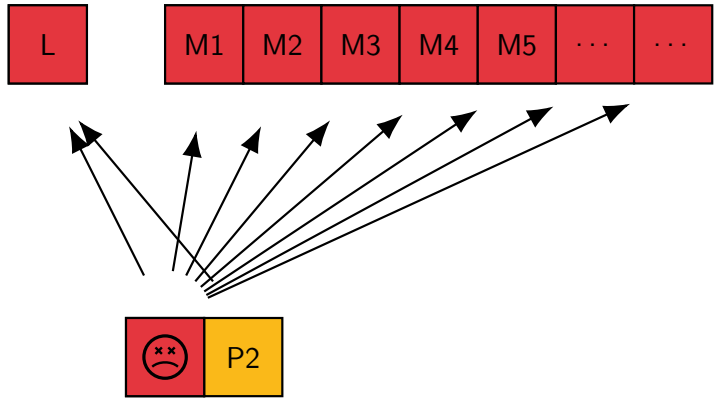


Optimization algorithms traditionally synchronous

- running at the pace of **slowest** processor
- what if memory is **large**?

Parallel Programming

Synchronization



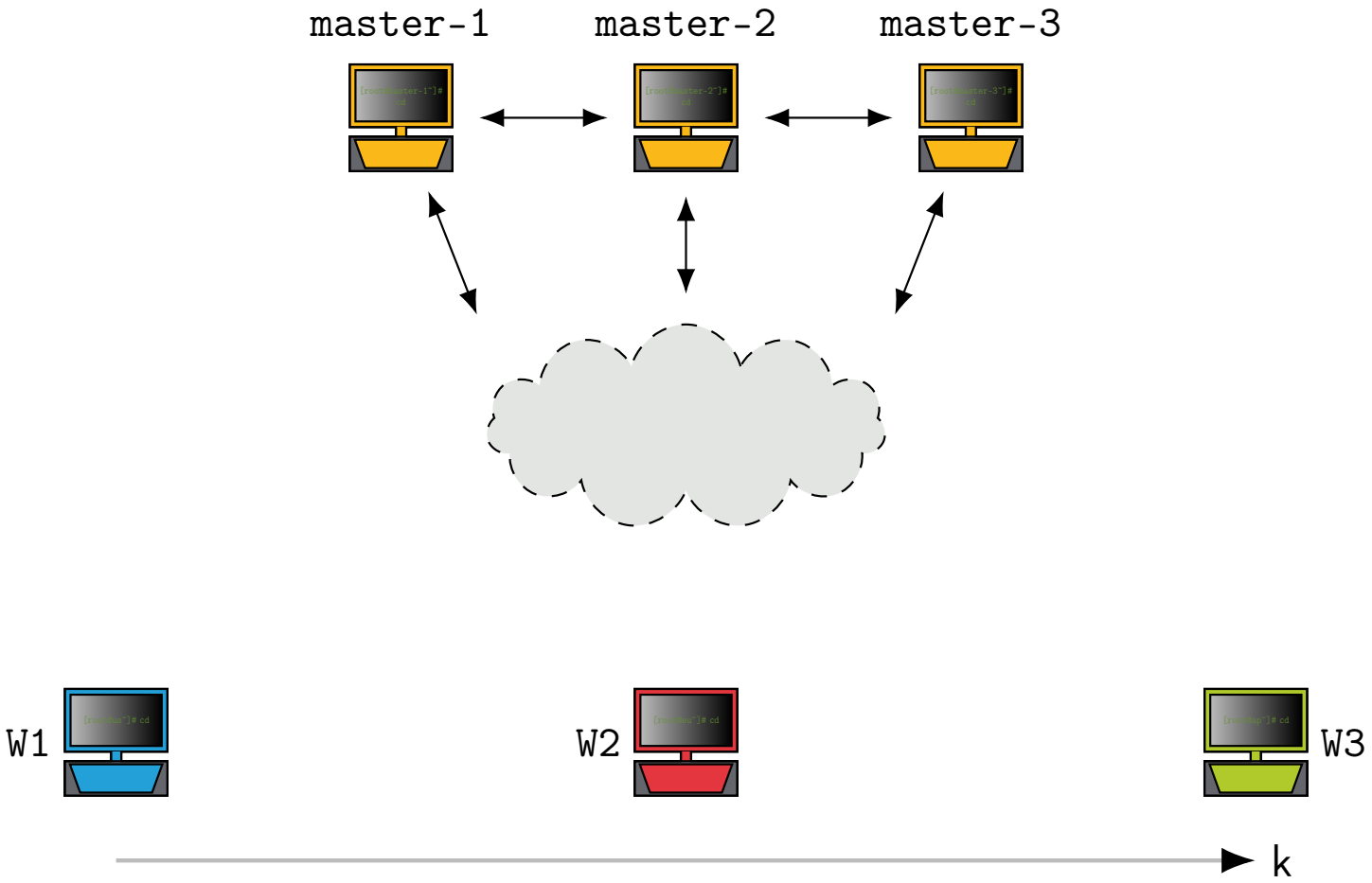
Optimization algorithms traditionally synchronous

- running at the pace of **slowest** processor
- what if memory is **large**?
- what if P1 **fails**?

Solution?

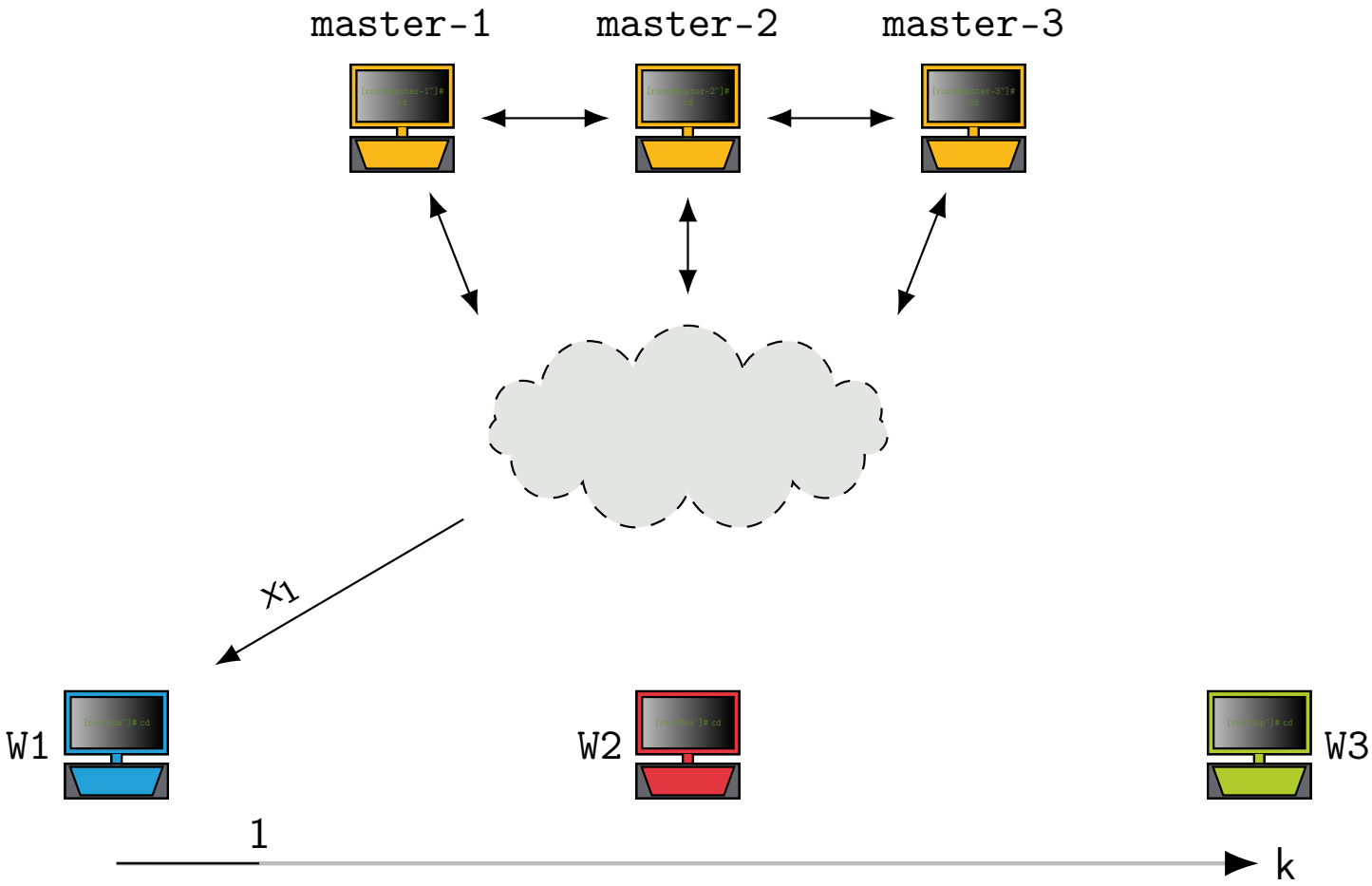
Parallel Programming

Parameter Server (Li et al. 2013)



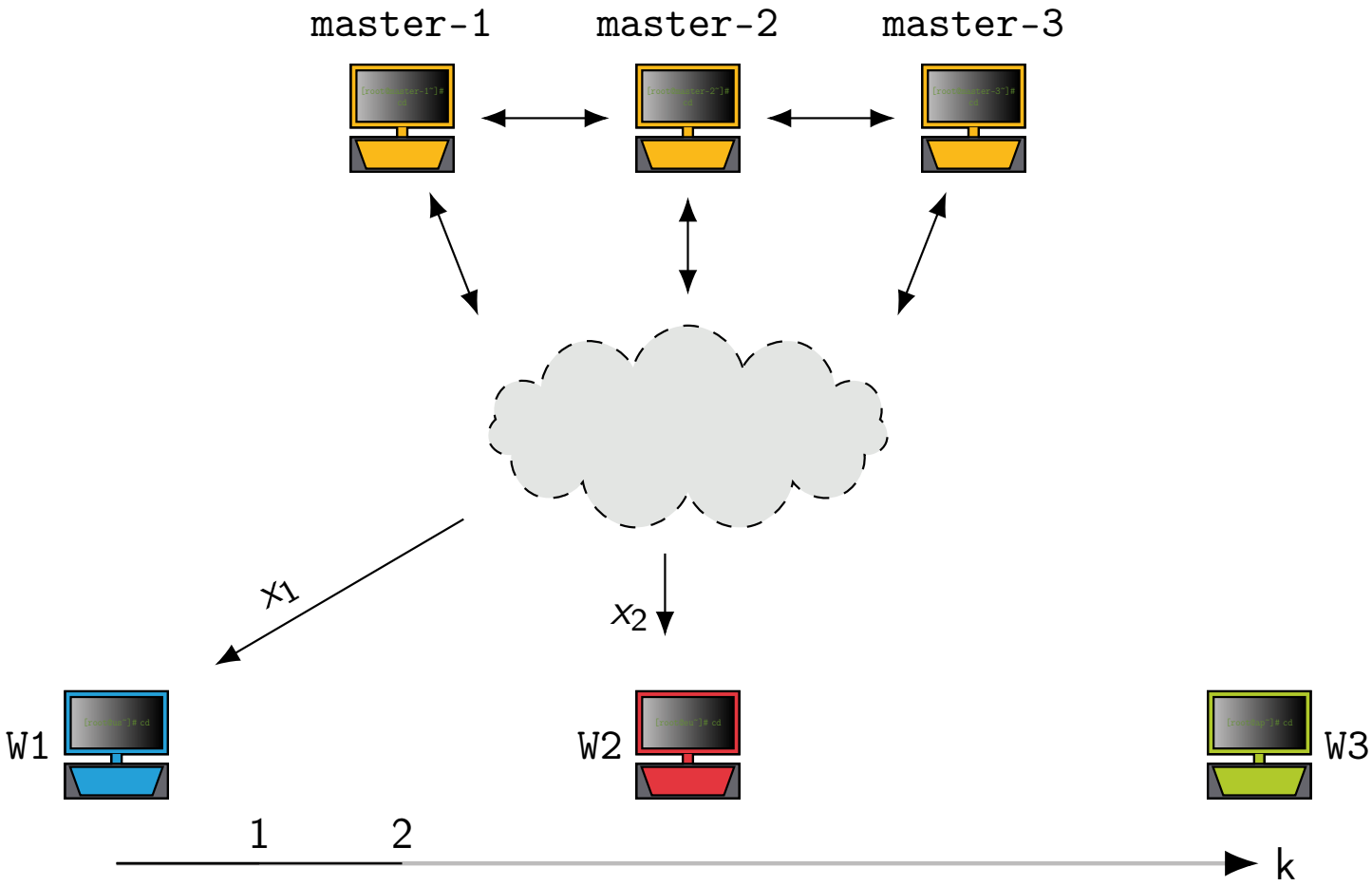
Parallel Programming

Parameter Server (Li et al. 2013)



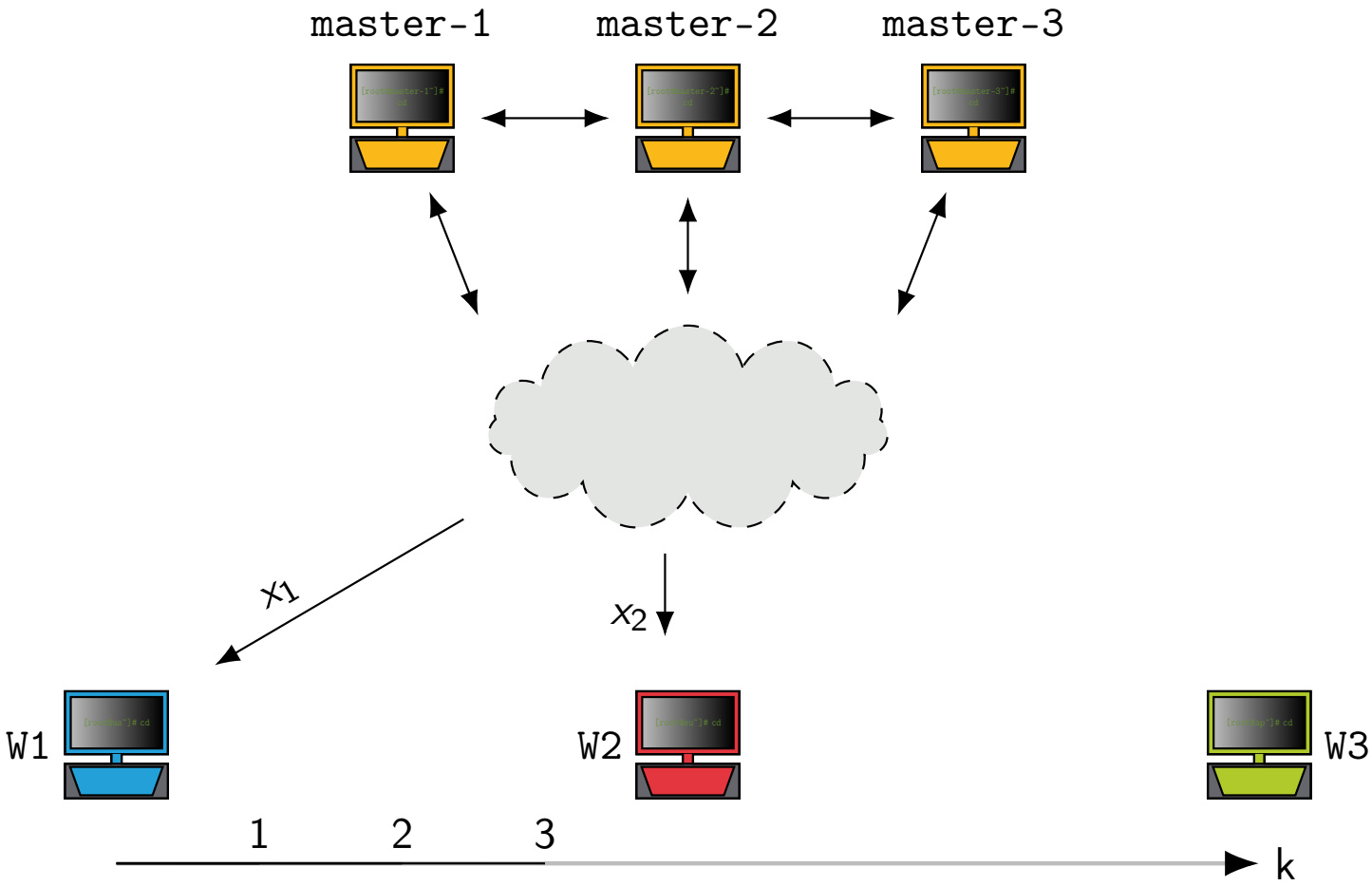
Parallel Programming

Parameter Server (Li et al. 2013)



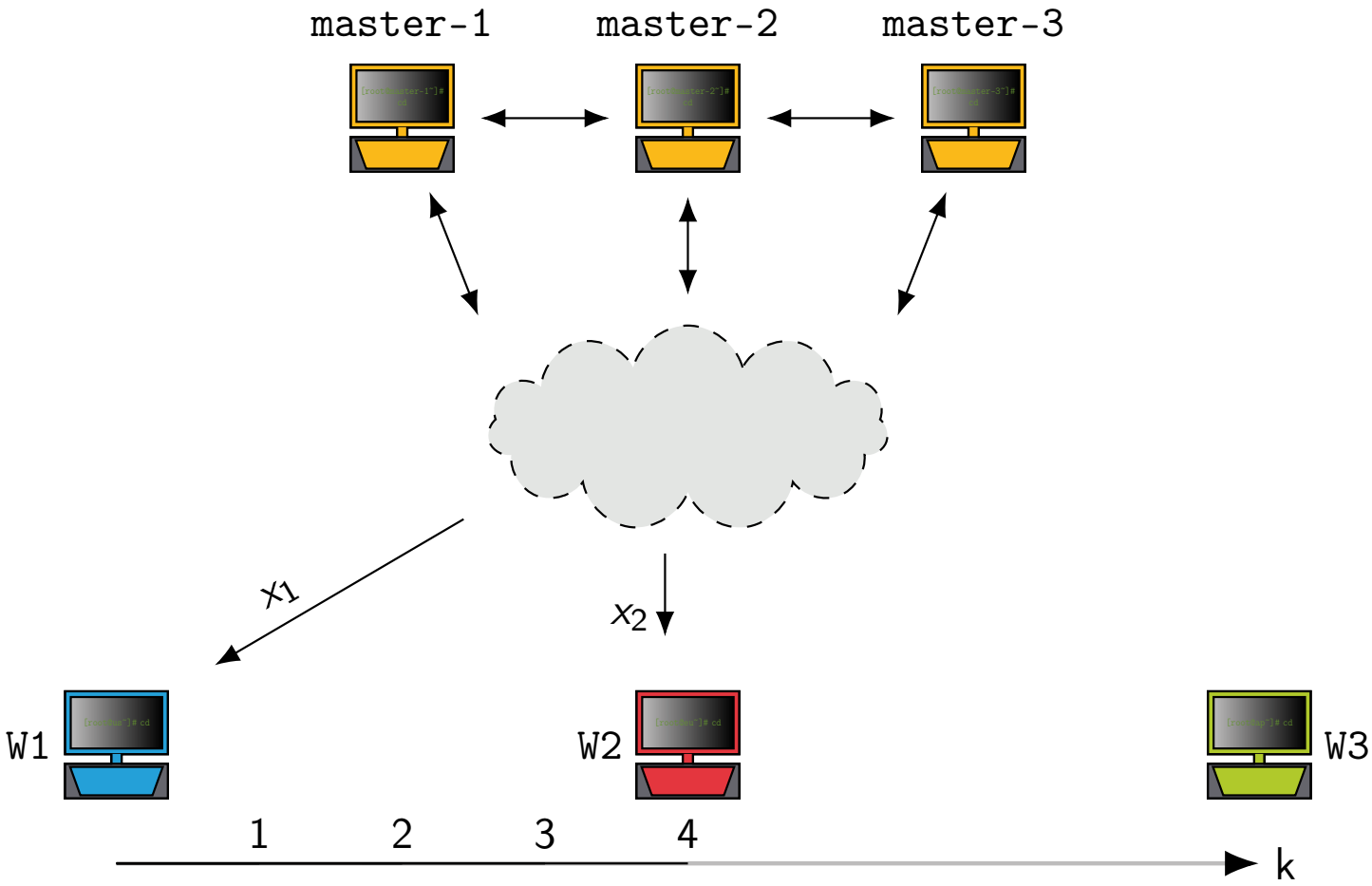
Parallel Programming

Parameter Server (Li et al. 2013)



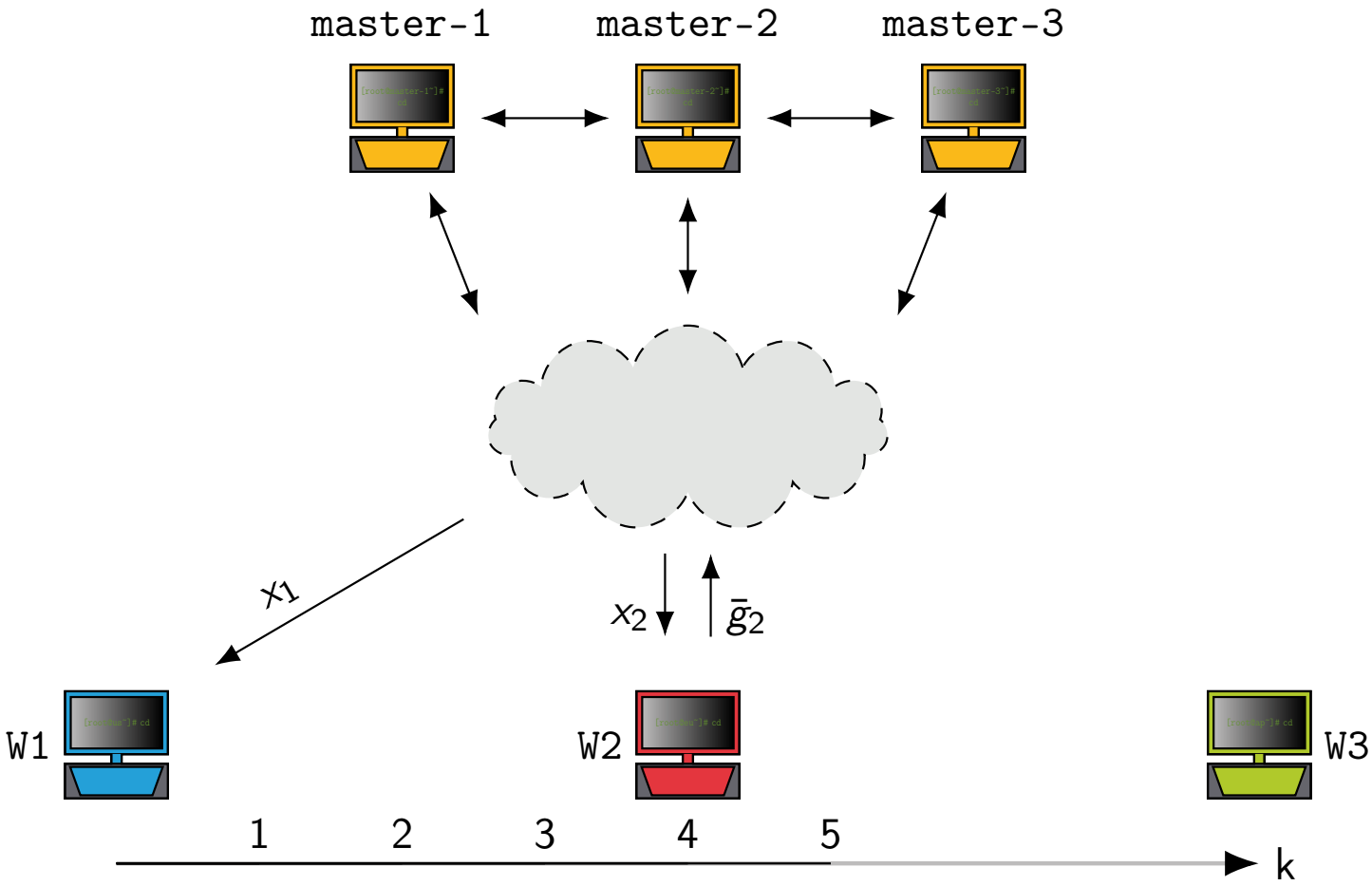
Parallel Programming

Parameter Server (Li et al. 2013)



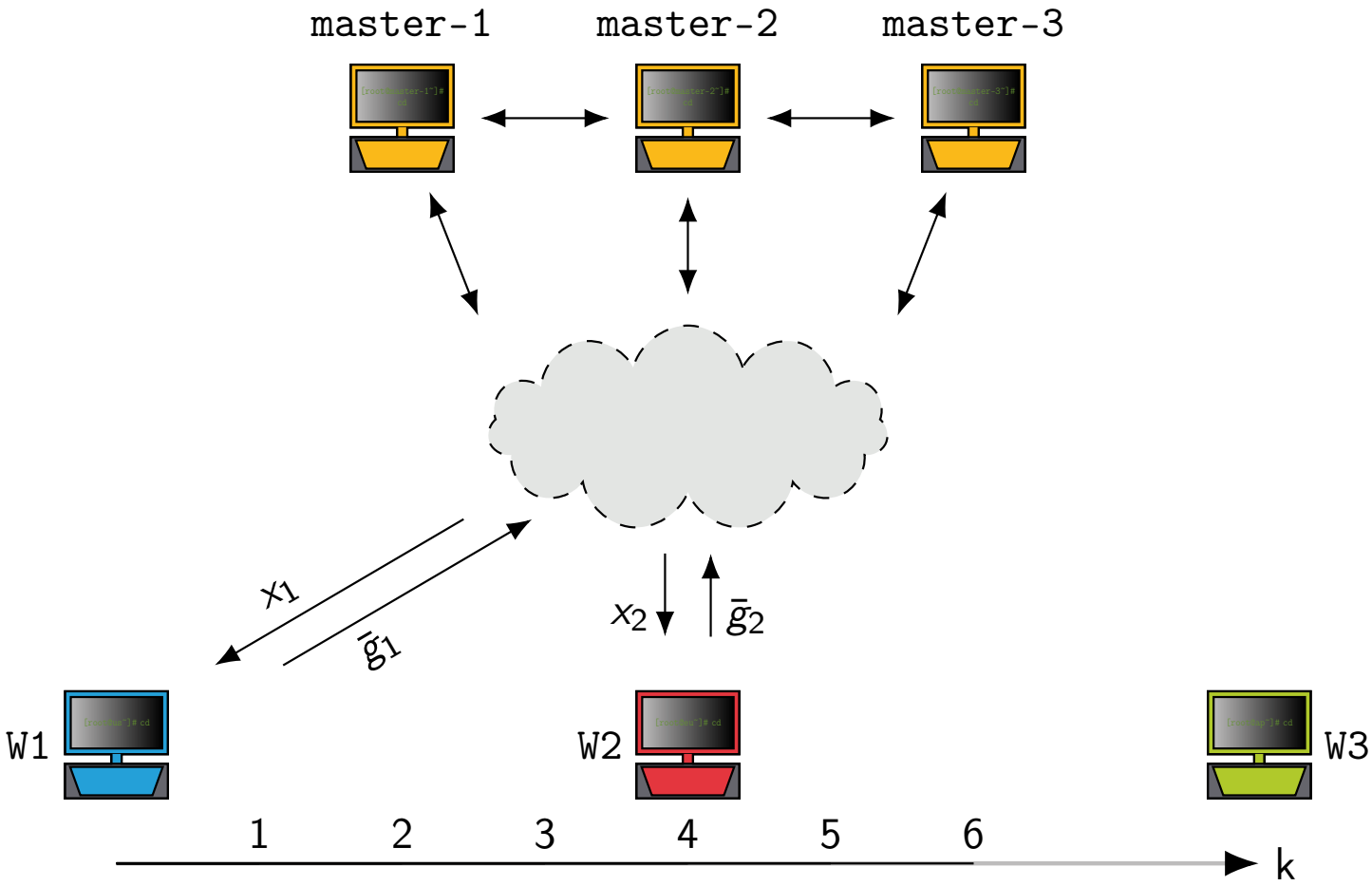
Parallel Programming

Parameter Server (Li et al. 2013)



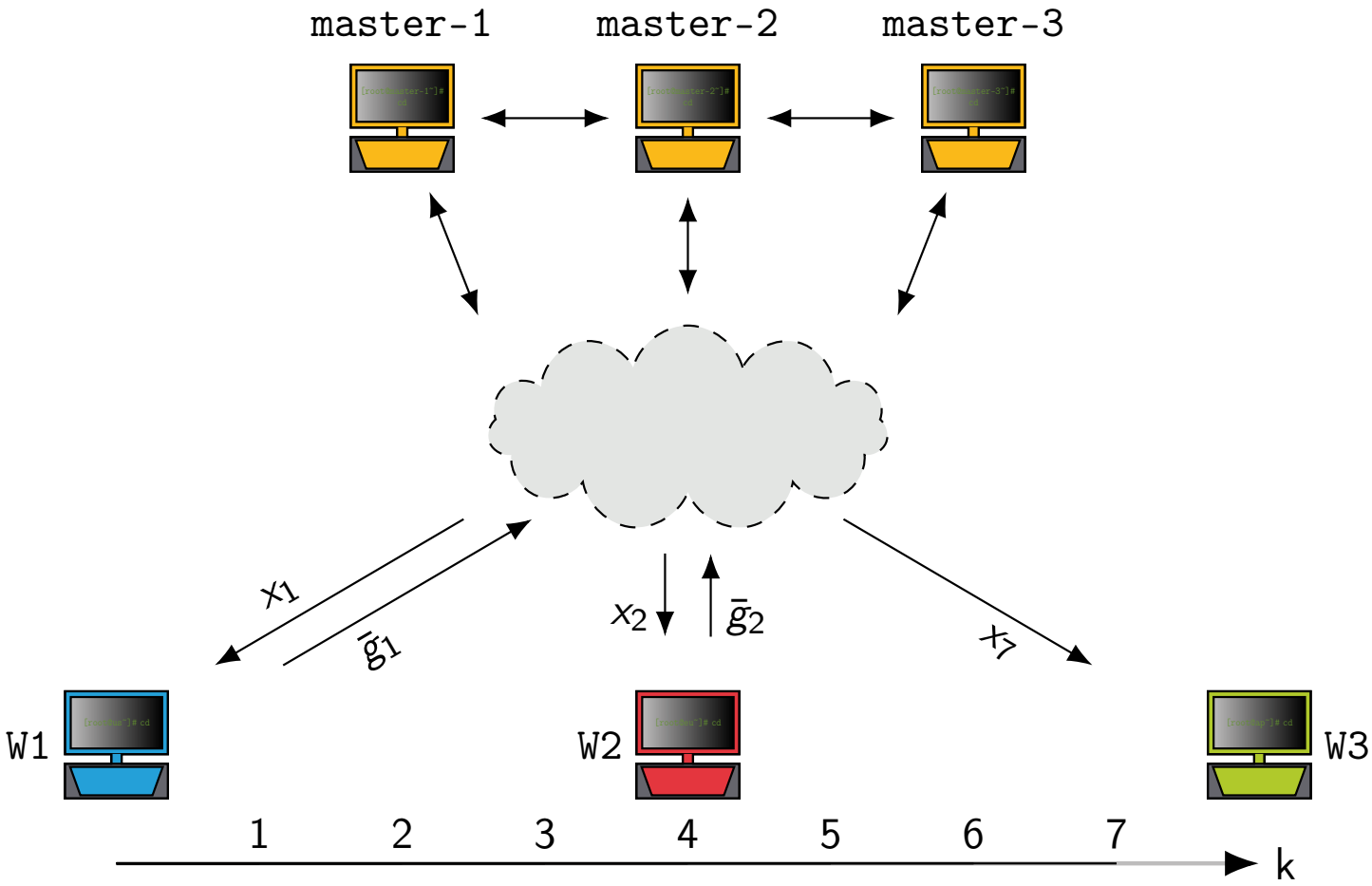
Parallel Programming

Parameter Server (Li et al. 2013)



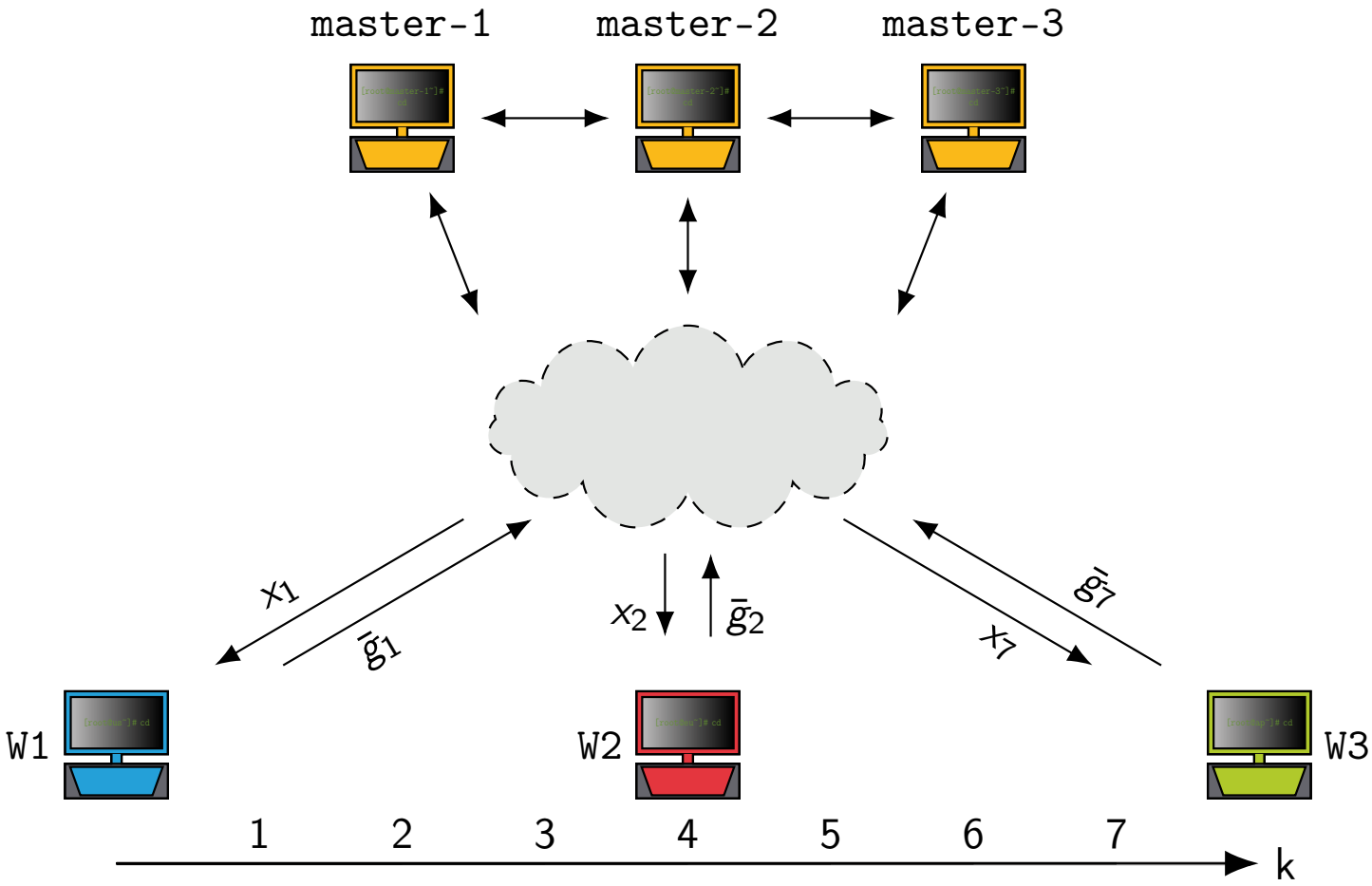
Parallel Programming

Parameter Server (Li et al. 2013)



Parallel Programming

Parameter Server (Li et al. 2013)



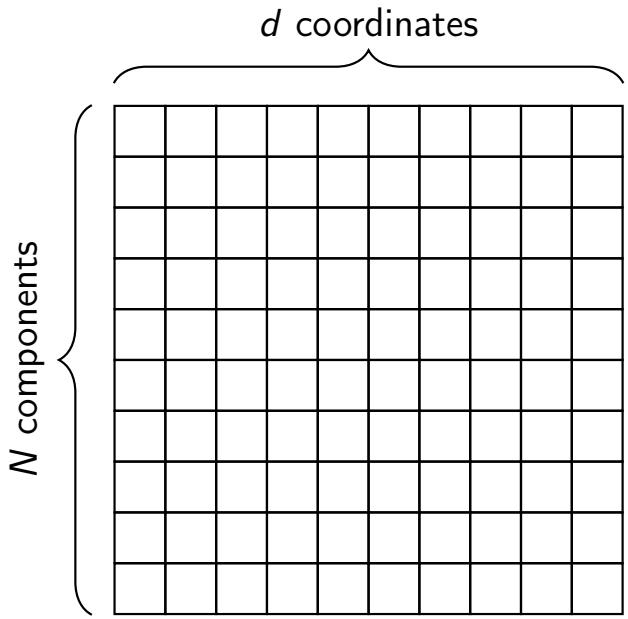
Problem Setup

Problem

$$\underset{x\in\mathbb{R}^d}{\text{minimize}} \quad \sum_{n=1}^{N} f_n(x) + h(x)$$

Iterations

$$x_{k+1} \leftarrow \Pi\Big(x_k, \gamma_k \nabla \quad f \quad (x_k \quad)\Big)$$



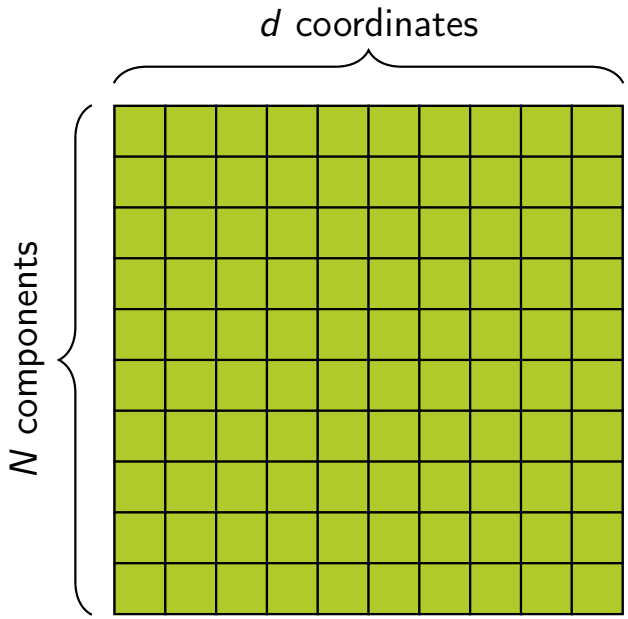
Problem Setup

Problem

$$\underset{x\in\mathbb{R}^d}{\text{minimize}} \quad \sum_{n=1}^{N} f_n(x) + h(x)$$

Iterations

$$x_{k+1} \leftarrow \Pi\Big(x_k, \gamma_k \nabla \quad f \quad (x_k \quad)\Big)$$



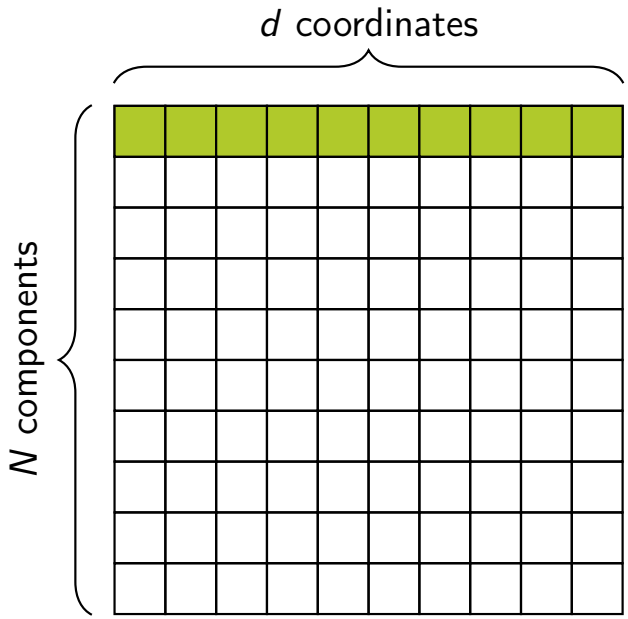
Problem Setup

Problem

$$\underset{x\in\mathbb{R}^d}{\text{minimize}} \quad \sum_{n=1}^{N} \mathsf{f}_n(x) + \mathsf{h}(x)$$

Asynchronous Iterations

$$x_{k+1} \leftarrow \Pi\left(x_k, \gamma_k \nabla \quad \mathsf{f}_{n_k}\left(x_{k-\tau_k}\right)\right)$$



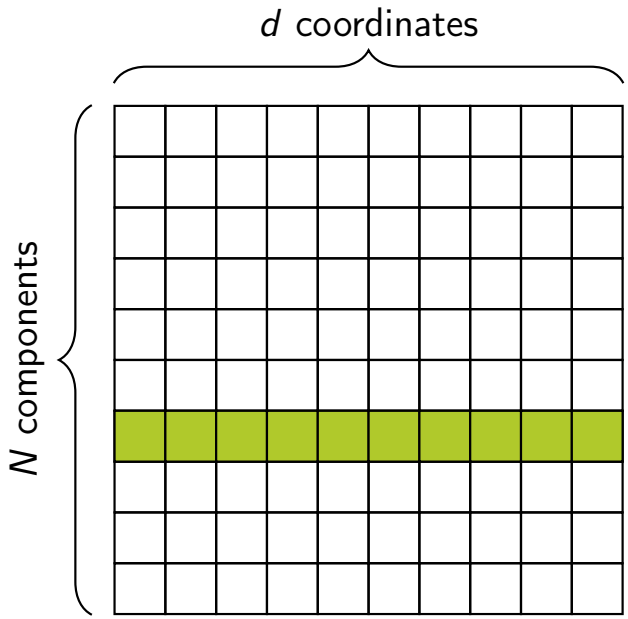
Problem Setup

Problem

$$\underset{x\in\mathbb{R}^d}{\text{minimize}} \quad \sum_{n=1}^{N} \mathsf{f}_n(x) + \mathsf{h}(x)$$

Asynchronous Iterations

$$x_{k+1} \leftarrow \Pi\left(x_k, \gamma_k \nabla \quad \mathsf{f}_{n_k}\left(x_{k-\tau_k}\right)\right)$$



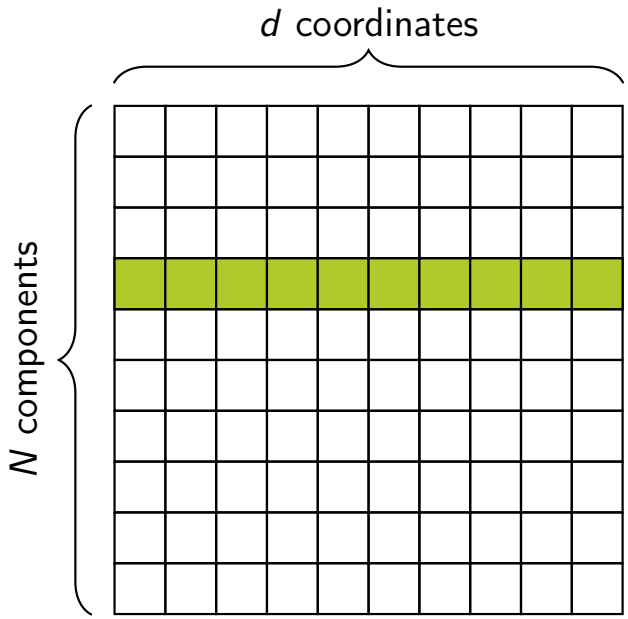
Problem Setup

Problem

$$\underset{x\in\mathbb{R}^d}{\text{minimize}} \quad \sum_{n=1}^{N} \mathsf{f}_n(x) + \mathsf{h}(x)$$

Asynchronous Iterations

$$x_{k+1} \leftarrow \Pi\left(x_k, \gamma_k \nabla \quad \mathsf{f}_{n_k}\left(x_{k-\tau_k}\right)\right)$$



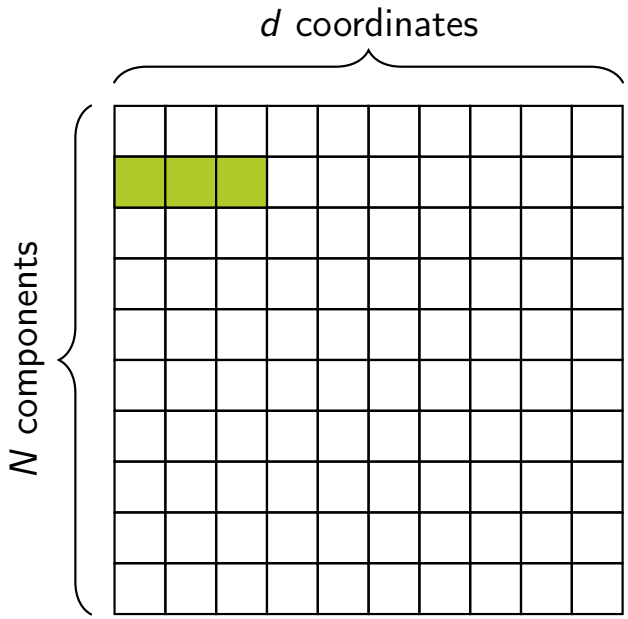
Problem Setup

Problem

$$\underset{x\in\mathbb{R}^d}{\text{minimize}} \quad \sum_{n=1}^{N} \mathsf{f}_n(x) + \mathsf{h}(x)$$

Asynchronous Iterations

$$x_{k+1} \leftarrow \Pi\left(x_k, \gamma_k \nabla^{[b_k]} \mathsf{f}_{n_k}\left(x_{k-\tau_k}\right)\right)$$



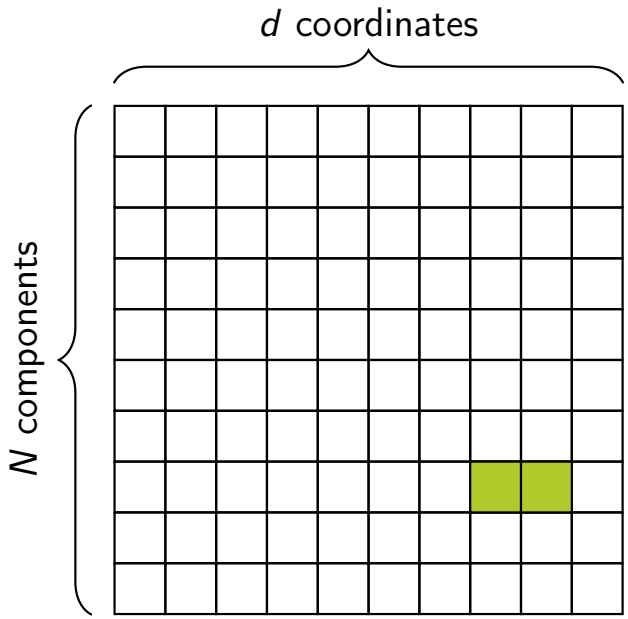
Problem Setup

Problem

$$\underset{x\in\mathbb{R}^d}{\text{minimize}} \quad \sum_{n=1}^{N} \mathsf{f}_n(x) + \mathsf{h}(x)$$

Asynchronous Iterations

$$x_{k+1} \leftarrow \Pi\left(x_k, \gamma_k \nabla^{[b_k]} \mathsf{f}_{n_k}\left(x_{k-\tau_k}\right)\right)$$



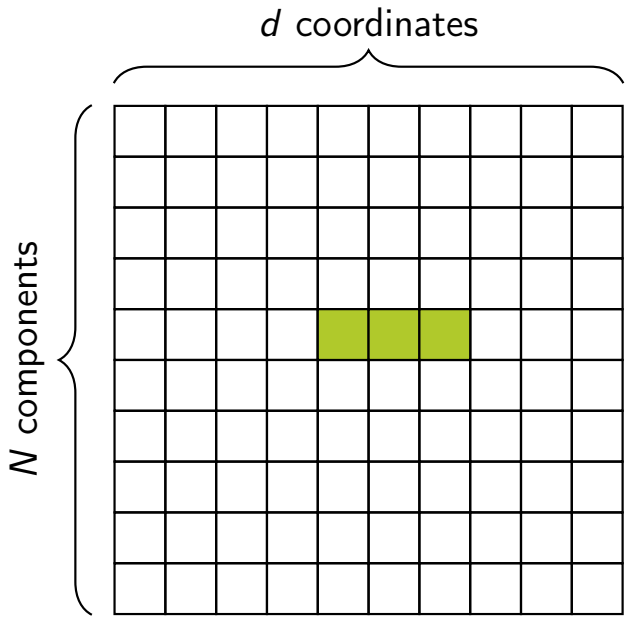
Problem Setup

Problem

$$\underset{x\in\mathbb{R}^d}{\text{minimize}} \quad \sum_{n=1}^{N} \mathsf{f}_n(x) + \mathsf{h}(x)$$

Asynchronous Iterations

$$x_{k+1} \leftarrow \Pi\left(x_k, \gamma_k \nabla^{[b_k]} \mathsf{f}_{n_k}\left(x_{k-\tau_k}\right)\right)$$



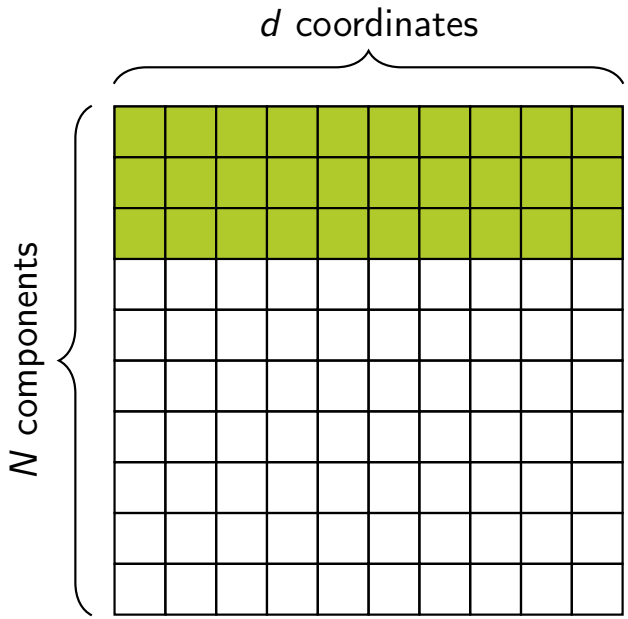
Problem Setup

Problem

$$\underset{x\in\mathbb{R}^d}{\text{minimize}} \quad \sum_{n=1}^{N} f_n(x) + h(x)$$

Asynchronous Iterations

$$g_k \leftarrow \sum_{n\in\mathbb{N}_w} \nabla f_n\left(x_{k-\tau_k}\right)$$



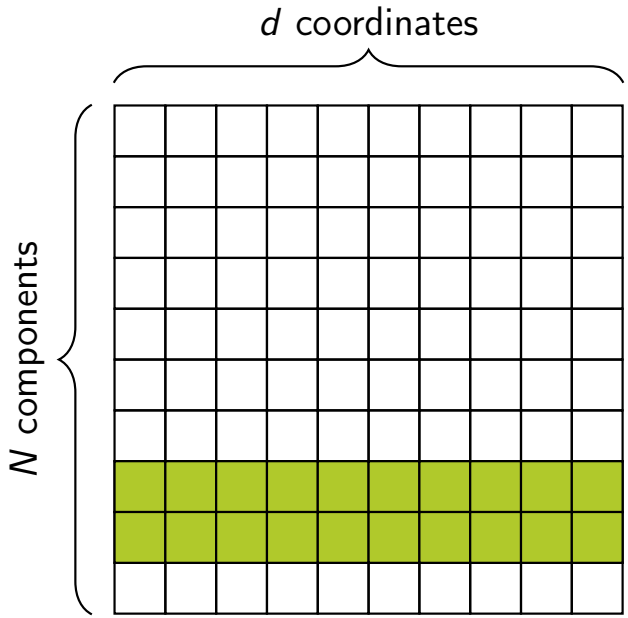
Problem Setup

Problem

$$\underset{x\in\mathbb{R}^d}{\text{minimize}} \quad \sum_{n=1}^{N} f_n(x) + h(x)$$

Asynchronous Iterations

$$g_k \leftarrow \sum_{n\in\mathbb{N}_w} \nabla f_n\left(x_{k-\tau_k}\right)$$



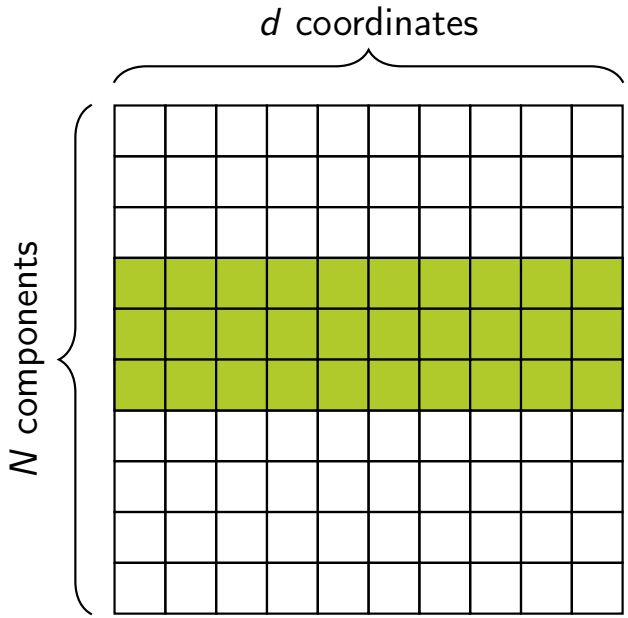
Problem Setup

Problem

$$\underset{x\in\mathbb{R}^d}{\text{minimize}} \quad \sum_{n=1}^{N} f_n(x) + h(x)$$

Asynchronous Iterations

$$g_k \leftarrow \sum_{n\in\mathbb{N}_w} \nabla f_n\left(x_{k-\tau_k}\right)$$



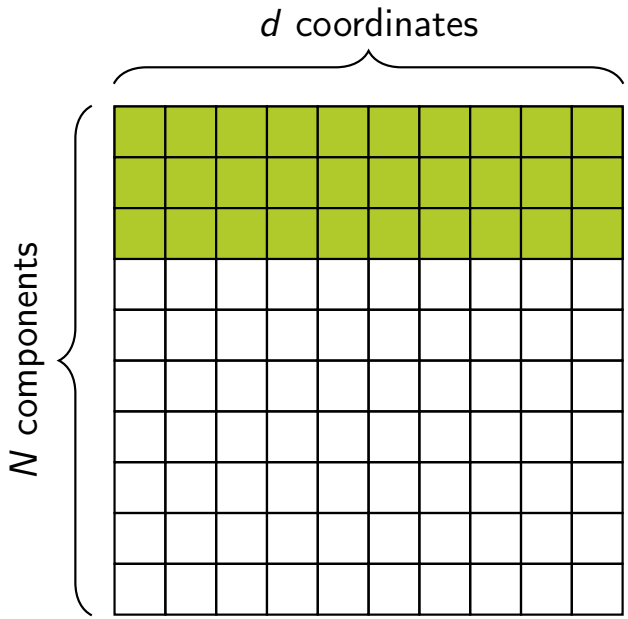
Problem Setup

Problem

$$\underset{x\in\mathbb{R}^d}{\text{minimize}} \quad \sum_{n=1}^{N} \mathrm{f}_n(x) + \mathrm{h}(x)$$

Asynchronous Iterations

$$g_k \leftarrow \sum_{w\in\mathbb{W}} \sum_{n\in\mathbb{N}_w} \nabla \mathrm{f}_n\left(x_{k-\tau_k^w}\right)$$



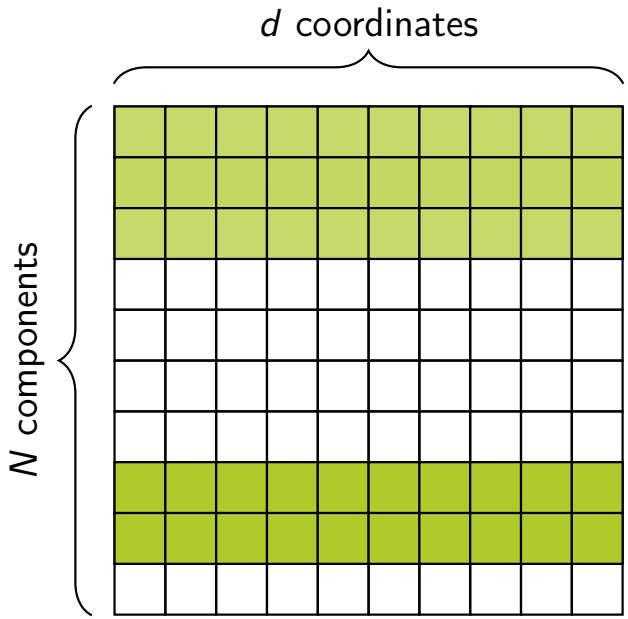
Problem Setup

Problem

$$\underset{x\in\mathbb{R}^d}{\text{minimize}} \quad \sum_{n=1}^{N} \mathrm{f}_n(x) + \mathrm{h}(x)$$

Asynchronous Iterations

$$g_k \leftarrow \sum_{w\in\mathbb{W}} \sum_{n\in\mathbb{N}_w} \nabla \mathrm{f}_n\left(x_{k-\tau_k^w}\right)$$



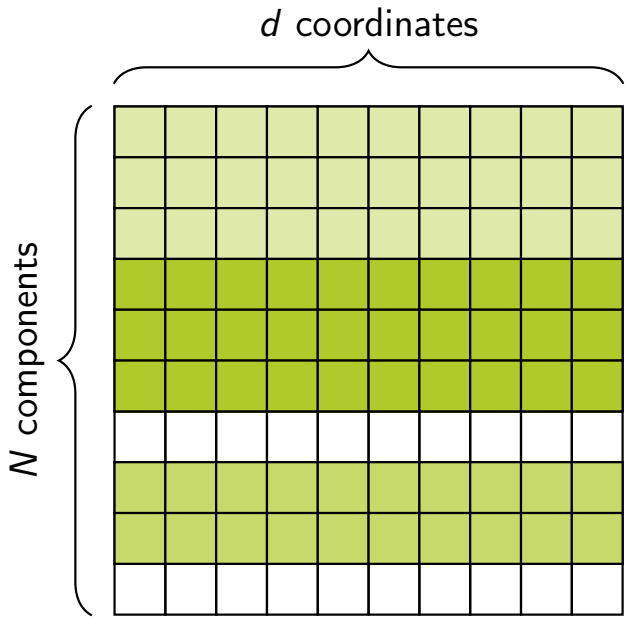
Problem Setup

Problem

$$\underset{x\in\mathbb{R}^d}{\text{minimize}} \quad \sum_{n=1}^{N} \mathsf{f}_n(x) + \mathsf{h}(x)$$

Asynchronous Iterations

$$g_k \leftarrow \sum_{w\in\mathbb{W}} \sum_{n\in\mathbb{N}_w} \nabla \mathsf{f}_n\left(x_{k-\tau_k^w}\right)$$



Incremental Aggregated Gradient

Problem Formulation

Problem

$$\underset{x\in\mathbb{R}^d}{\text{minimize}} \quad \sum_{n=1}^{N} f_n(x) + h(x)$$

Algorithm

$$g_k \leftarrow \sum_{w\in\mathbb{W}} \sum_{n\in\mathbb{N}_w} \nabla f_n\left(x_{k-\tau_k^w}\right)$$

$$x_{k+1} \leftarrow \arg\min_x \left\{ \langle g_k, x - x_k\rangle + \frac{1}{2\gamma}\|x - x_k\|^2 + h(x) \right\}$$

Incremental Aggregated Gradient

Prior Work

- Blatt, Hero, and Gauchman 2007
  - global convergence; each f_n convex quadratic; $h(x) = 0$

Incremental Aggregated Gradient

Prior Work

- Blatt, Hero, and Gauchman 2007
  - global convergence; each f_n convex quadratic; $h(x) = 0$
- Gurbuzbalaban, Ozdaglar, and Parrilo 2015
  - linear convergence; f_n strongly convex; $h(x) = 0$

Incremental Aggregated Gradient

Prior Work

- Blatt, Hero, and Gauchman 2007
  - global convergence; each f_n convex quadratic; $h(x) = 0$
- Gurbuzbalaban, Ozdaglar, and Parrilo 2015 (First Sequence Lemma)
  - linear convergence; f_n strongly convex; $h(x) = 0$

Incremental Aggregated Gradient

Prior Work

- Blatt, Hero, and Gauchman 2007
  - global convergence; each f_n convex quadratic; $h(x) = 0$
- Gurbuzbalaban, Ozdaglar, and Parrilo 2015 (First Sequence Lemma)
  - linear convergence; f_n strongly convex; $h(x) = 0$
- D. P. Bertsekas 2015, Review Paper
  - "*. . . to our knowledge, a linear convergence rate . . . with an orthant constraint is not currently available.*"

Incremental Aggregated Gradient

Prior Work

- Blatt, Hero, and Gauchman 2007
  - global convergence; each f_n convex quadratic; $h(x) = 0$
- Gurbuzbalaban, Ozdaglar, and Parrilo 2015 (First Sequence Lemma)
  - linear convergence; f_n strongly convex; $h(x) = 0$
- D. P. Bertsekas 2015, Review Paper
  - "*. . . to our knowledge, a linear convergence rate . . . with an orthant constraint is not currently available.*"
- Vanli, Gurbuzbalaban, and Ozdaglar 2016

- Aytekin, Feyzmahdavian, and Johansson 2016

Incremental Aggregated Gradient

Prior Work

- Blatt, Hero, and Gauchman 2007
  - global convergence; each f_n convex quadratic; $h(x) = 0$
- Gurbuzbalaban, Ozdaglar, and Parrilo 2015 (First Sequence Lemma)
  - linear convergence; f_n strongly convex; $h(x) = 0$
- D. P. Bertsekas 2015, Review Paper
  - "*. . . to our knowledge, a linear convergence rate . . . with an orthant constraint is not currently available.*"
- Vanli, Gurbuzbalaban, and Ozdaglar 2016
  - proximal term; linear convergence; after sufficiently many iterations
- Aytekin, Feyzmahdavian, and Johansson 2016

Incremental Aggregated Gradient

Prior Work

- Blatt, Hero, and Gauchman 2007
  - global convergence; each f_n convex quadratic; $h(x) = 0$
- Gurbuzbalaban, Ozdaglar, and Parrilo 2015 (First Sequence Lemma)
  - linear convergence; f_n strongly convex; $h(x) = 0$
- D. P. Bertsekas 2015, Review Paper
  - "*. . . to our knowledge, a linear convergence rate . . . with an orthant constraint is not currently available.*"
- Vanli, Gurbuzbalaban, and Ozdaglar 2016 (First Sequence Lemma)
  - proximal term; linear convergence; after sufficiently many iterations
- Aytekin, Feyzmahdavian, and Johansson 2016

Incremental Aggregated Gradient

Prior Work

- Blatt, Hero, and Gauchman 2007
  - global convergence; each f_n convex quadratic; $h(x) = 0$
- Gurbuzbalaban, Ozdaglar, and Parrilo 2015 (First Sequence Lemma)
  - linear convergence; f_n strongly convex; $h(x) = 0$
- D. P. Bertsekas 2015, Review Paper
  - "*. . . to our knowledge, a linear convergence rate . . . with an orthant constraint is not currently available.*"
- Vanli, Gurbuzbalaban, and Ozdaglar 2016 (First Sequence Lemma)
  - proximal term; linear convergence; after sufficiently many iterations
- Aytekin, Feyzmahdavian, and Johansson 2016
  - proximal term; linear convergence.
  - Second Sequence Lemma

Incremental Aggregated Gradient

Lemma (First Sequence Lemma [FAJ'14])

Let $\{V_k\}$ be a sequence of real numbers satisfying

$$V_{k+1} \leq a_1 V_k + a_2 \max_{k-\tau_k \leq \tilde{k} \leq k} V_{\tilde{k}} + a_3\,, \quad k \in \mathbb{N}_0\,,$$

for some nonnegative constants a_1, a_2, and a_3. If $a_1 + a_2 < 1$ and

$$0 \leq \tau_k \leq \bar{\tau}\,, \quad k \in \mathbb{N}_0\,,$$

then

$$V_k \leq \rho^k V_0 + \epsilon\,, \quad k \in \mathbb{N}_0\,,$$

where $\rho = (a_1 + a_2)^{\frac{1}{1+\bar{\tau}}}$ and $\epsilon = a_3/(1 - a_1 - a_2)$.

Incremental Aggregated Gradient

Lemma (Second Sequence Lemma [AFJ'16])

Assume that the non-negative sequences $\{V_k\}$ *and* $\{w_k\}$ *satisfy the following inequality:*

$$V_{k+1} \leq a_1 V_k - a_2 w_k + a_3 \sum_{j=k-k_0}^{k} w_j \,,$$

for some real numbers $a_1 \in (0,1)$ *and* $a_2, a_3 \geq 0$*, and some integer* $k_0 \in \mathbb{N}_0$*. Assume also that* $w_k = 0$ *for* $k < 0$*, and that the following holds:*

$$\frac{a_3}{1-a_1} \frac{1-a_1^{k_0+1}}{a_1^{k_0}} \leq a_2 \,.$$

Then, $V_k \leq a_1^k V_0$ *for all* $k \geq 0$*.*

Incremental Aggregated Gradient

Main Result

Theorem

Assume that f_n *are* L_n*-smooth and* f *is* μ*-strongly convex, and that the step-size* γ *satisfies:*

$$\gamma \leq \frac{\left(1+\frac{\mu}{L}\frac{1}{\bar{\tau}+1}\right)^{\frac{1}{(\bar{\tau}+1)}}-1}{\mu},$$

where $L=\sum_{n=1}^{N} L_n$*. Then, the iterates satisfy:*

$$\|x_k-x^\star\|^2 \leq \left(\frac{1}{\mu\gamma+1}\right)^k \|x_0-x^\star\|^2 .$$

for all $k \geq 0$*.*

simple step-size rule, linear convergence, recovers $\gamma \leq \frac{1}{L}$ when $\bar{\tau}=0$

Incremental Aggregated Gradient

Main Result

Corollary

Consider using the following proximal gradient method

$$x_{k+1} = \arg\min_x \left\{ \langle g_k, x - x_k \rangle + \frac{1}{\gamma} \mathsf{D}_\omega (x, x_k) + \mathsf{h}(x) \right\} ,$$

$$g_k = \sum_{n=1}^{N} \nabla \mathsf{f}_n \left(x_{k-\tau_k^n} \right) ,$$

$$\gamma \leq \frac{L_\omega \left(1 + \frac{\mu}{L} \frac{1}{\bar{\tau}+1} \frac{\mu_\omega}{L_\omega} \right)^{\frac{1}{\bar{\tau}+1}} - 1}{\mu} ,$$

where $L = \sum_{n=1}^{N} L_n$. *Then, the iterates satisfy:*

$$\mathsf{D}_\omega (x^\star, x_k) \leq \left(\frac{L_\omega}{\mu\gamma + L_\omega} \right)^k \mathsf{D}_\omega (x^\star, x_0) .$$

Numerical Example

Example (Synthetic Problem)

$$\underset{x\in\mathbb{R}^d}{\text{minimize}} \quad \sum_{n=1}^{N} \mathsf{f}_n(x) + \mathsf{h}(x)$$

with

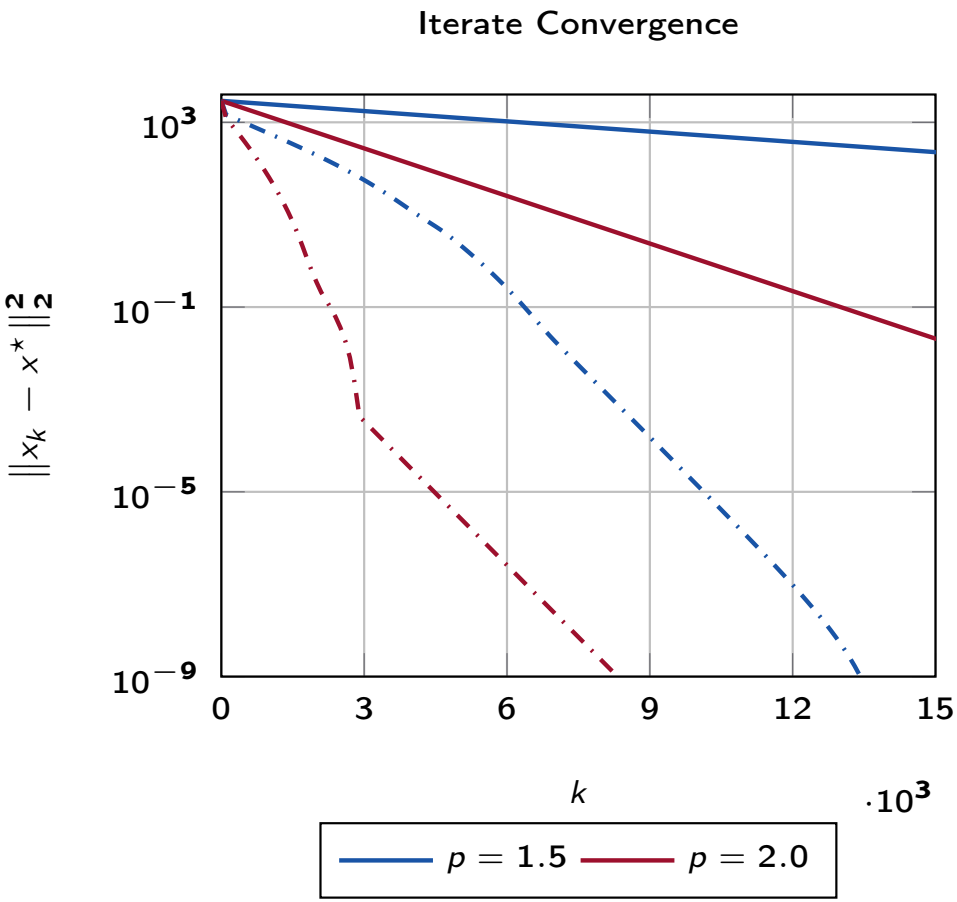
$$\mathsf{f}_n(x) = \begin{cases} (x_n - c_1)^2 + \frac{1}{2}(x_{n+1} + c_1)^2 & n = 1\,, \\ \frac{1}{2}(x_{n-1} + c_1)^2 + \frac{1}{2}(x_n - c_1)^2 & n = N\,, \\ \frac{1}{2}(x_{n-1} + c_1)^2 + \frac{1}{2}(x_n - c_1)^2 + \frac{1}{2}(x_{n+1} + c_1)^2 & \text{otherwise,} \end{cases}$$

$$\mathsf{h}(x) = \lambda_1 \|x\|_1 + \mathbf{I}_{\mathbb{C}}(x)\,,$$

$$\mathbb{C} = \{x \geq 0\}\,,$$

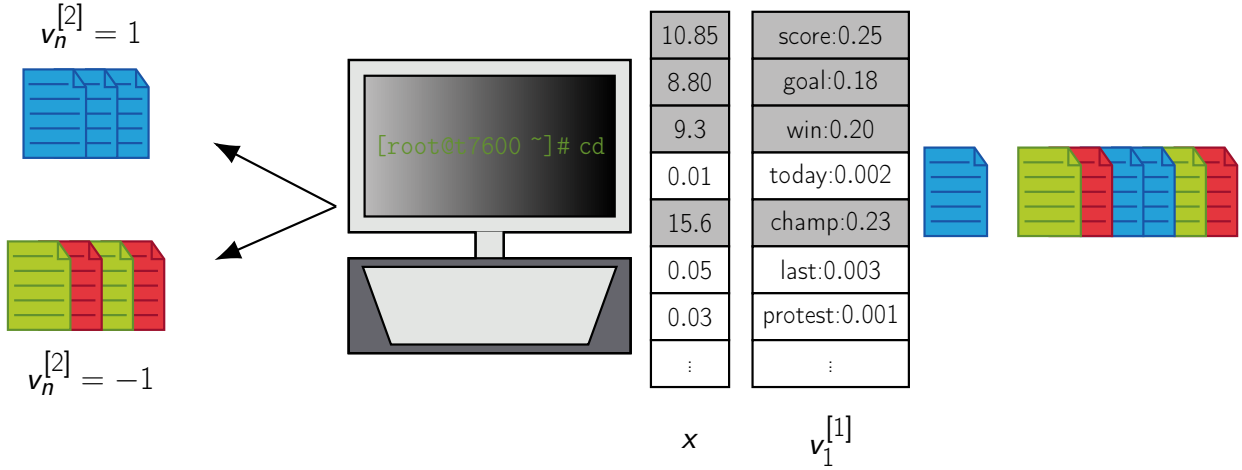
Numerical Example

Toy Example



Numerical Example

Binary Classification



$$\underset{x}{\text{minimize}} \quad \underbrace{\frac{1}{N}\sum_{n=1}^{N} \log\left(1+\exp\left(-v_n^{[2]}\left\langle v_n^{[1]}, x\right\rangle\right)\right) + \frac{1}{2}\lambda_2 \|x\|_2^2}_{\text{empirical model error}} + \underbrace{\lambda_1 \|x\|_1}_{\text{regularization}}$$

Numerical Example

Amazon EC2

Example (Binary Classification)

$$\underset{x}{\text{minimize}} \quad \sum_{n=1}^{N} \log\left(1+\exp\left(-v_n^{[2]}\left\langle v_n^{[1]}, x\right\rangle\right)\right) + \frac{1}{2}\lambda_2 \|x\|_2^2 + \lambda_1 \|x\|_1$$

with

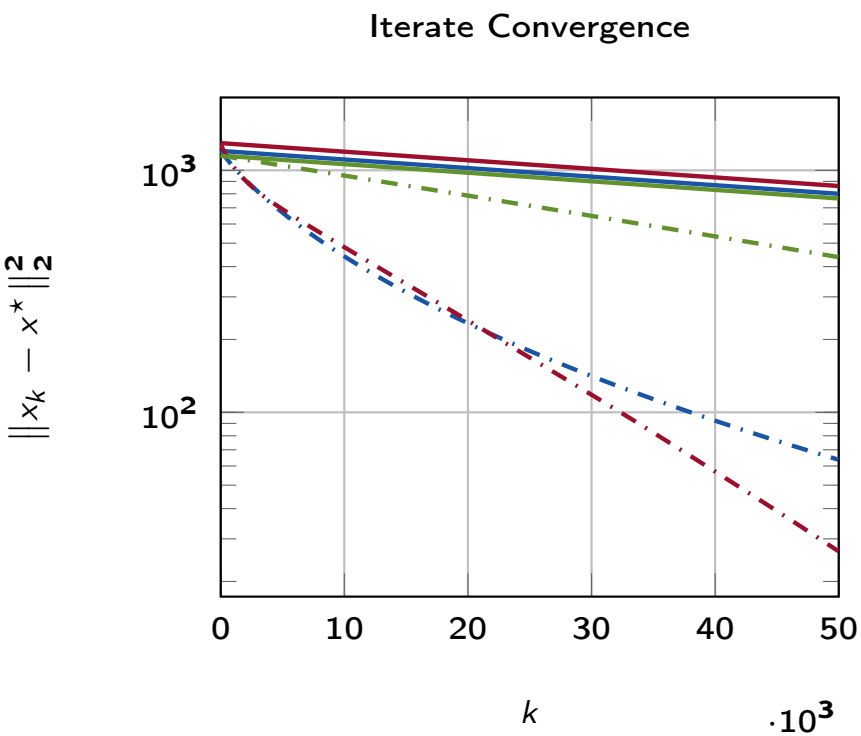
- `epsilon`: $N = 500,000$, $d = 2,000$ (density: 1.0)
- `rcv1`: $N = 804,414$, $d = 47,236$ (density: 0.0016)
- `url`: $N = 2,396,961$, $d = 64$ (density: 0.1808)

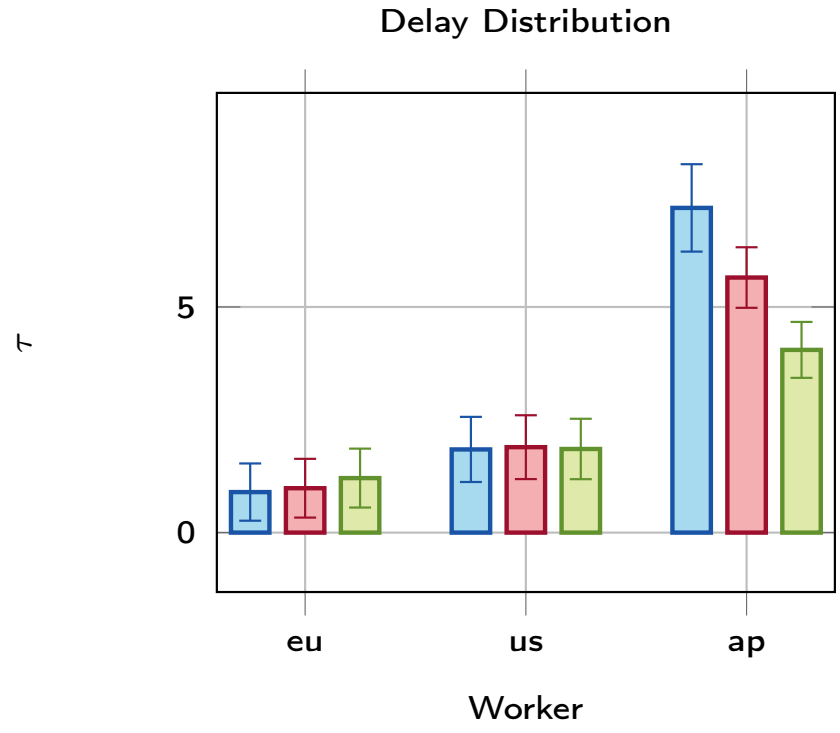
on

- Amazon EC2 with workers in Europe, US and Pacific Asia

Numerical Example

Amazon EC2

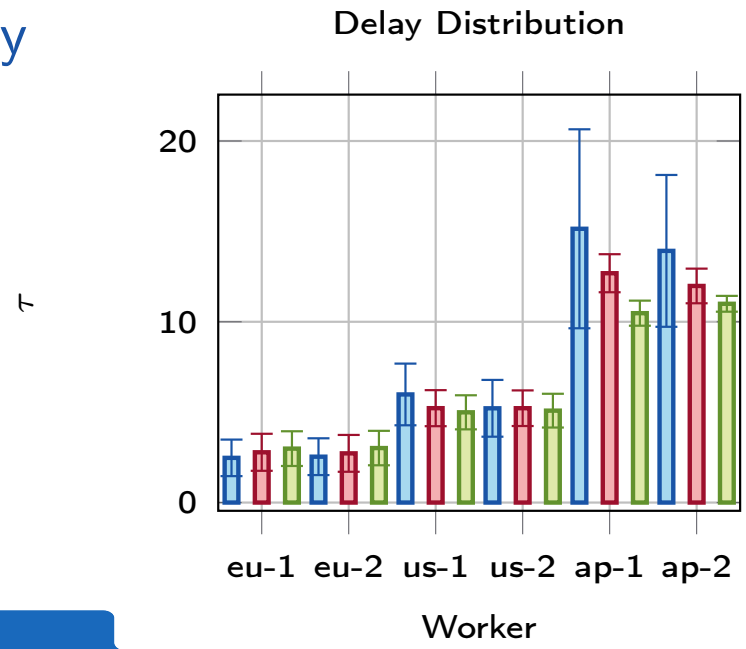




Convergence results and delay distributions for `rcv1`, `url` and `epsilon`

Conclusion

- Delays are inherent in **asynchronous** parallel programming
- Two important lemmas in **linear convergence** of **delayed** iterations
- Exploit sparsity pattern
  - **average degree**, other measures
- Exploit delay information
  - stochastic delays, delay-adaptive step sizes
- Efficient algorithms respecting data locality



THANK YOU!

Lemma (First Sequence Lemma [FAJ'14])

Let $\{V_k\}$ be a sequence of real numbers satisfying

$$V_{k+1} \leq a_1 V_k + a_2 \max_{k-\tau_k \leq \tilde{k} \leq k} V_{\tilde{k}} + a_3\,, \quad k \in \mathbb{N}_0\,,$$

for some nonnegative constants a_1, a_2, and a_3. If $a_1 + a_2 < 1$ and

$$0 \leq \tau_k \leq \bar{\tau}\,, \quad k \in \mathbb{N}_0\,,$$

then

$$V_k \leq \rho^k V_0 + \epsilon\,, \quad k \in \mathbb{N}_0\,,$$

where $\rho = (a_1 + a_2)^{\frac{1}{1+\bar{\tau}}}$ and $\epsilon = a_3/(1 - a_1 - a_2)$.

Proof Sketch.

1. Use induction to have $V_{\bar{k}+1} \leq \left(a_1 + a_2 \rho^{-\bar{\tau}}\right) \rho^{\bar{k}} V_0 + \epsilon$
2. Use $a_1 + a_2 < 1$ to upper bound $a_1 + a_2 \rho^{-\bar{\tau}}$

■

Lemma (Second Sequence Lemma [AFJ'16])

Assume that the non-negative sequences $\{V_k\}$ *and* $\{w_k\}$ *satisfy the following inequality:*

$$V_{k+1} \leq a_1 V_k - a_2 w_k + a_3 \sum_{j=k-k_0}^{k} w_j \,,$$

for some real numbers $a_1 \in (0,1)$ *and* $a_2, a_3 \geq 0$*, and some integer* $k_0 \in \mathbb{N}_0$*. Assume also that* $w_k = 0$ *for* $k < 0$*, and that the following holds:*

$$\frac{a_3}{1-a_1} \frac{1-a_1^{k_0+1}}{a_1^{k_0}} \leq a_2 \,.$$

Then, $V_k \leq a_1^k V_0$ *for all* $k \geq 0$*.*

Proof Sketch.

1. Divide by a_1^{k+1}
2. Expand series
3. Use nonnegativity to upper bound
4. Make sure w_k vanishes

■

Assumption

- (Subdifferentiability) $\mathsf{h}(x) \geq \mathsf{h}(y) + \langle s_y, x - y \rangle \ , \quad \forall s_y \in \partial\, \mathsf{h}(y)$
- (Smoothness) Each f_n is L_n-smooth
- (Strong Convexity) f is μ-strongly convex
- (Bounded Delay) $0 \leq \tau_k^n \leq \bar{\tau}, \quad k \in \mathbb{N}_0$

Proof Sketch.

1. Use convexity and smoothness
2. Use optimality condition of the proximal step
3. Three-point identity, subdifferentiability, strong convexity
4. Jensen's inequality to have a relation between $\|x_{k+1} - x^\star\|^2$ and $\|x_{k+1} - x_k\|^2$
5. Use Second Sequence Lemma

■