Karlsruhe Institute of Technology

Technical University of Munich

A probabilistic model for asynchronous iterative methods

APDCM workshop, IPDPS 2024, San Francisco, 27th May, 2024

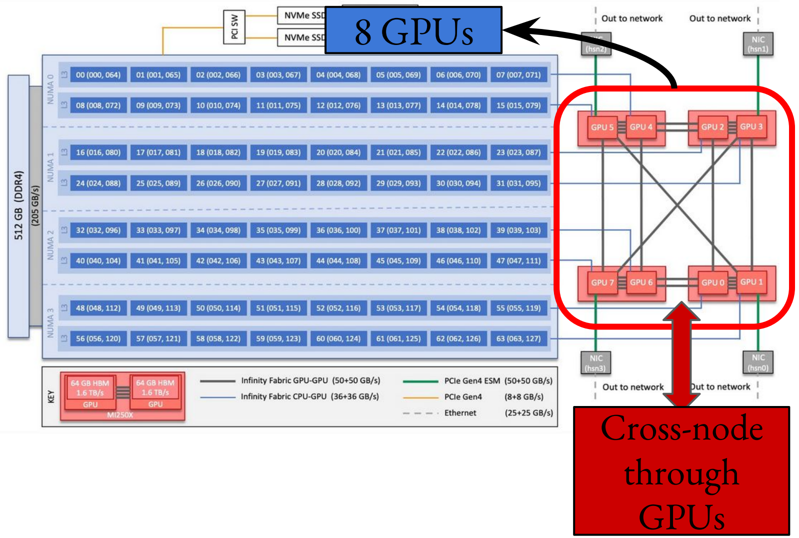
Pratik Nayak and Hartwig Anzt

Outline

- Motivation
- Asynchronous Richardson: Model and analysis
- Asynchronous Schwarz: Model and analysis
- Summary and outlook

Motivation

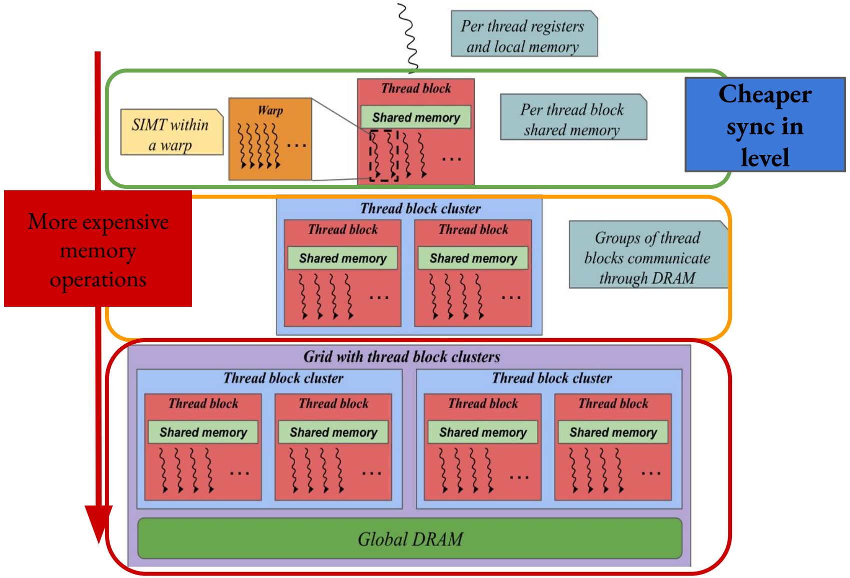
- Compute systems are increasingly heterogeneous



One Frontier node (#1 in top500)

Motivation

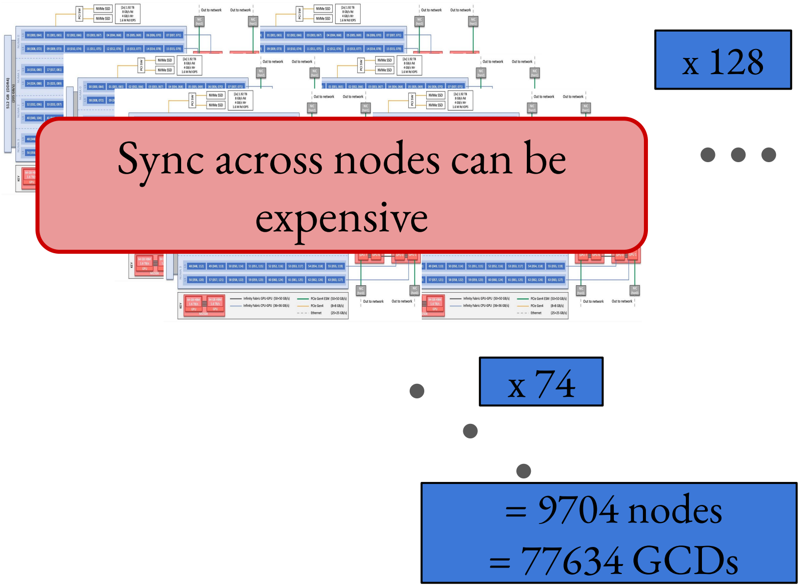
- Compute systems are increasingly heterogeneous
- Compute systems are increasingly hierarchical



NVIDIA GPU schematic

Motivation

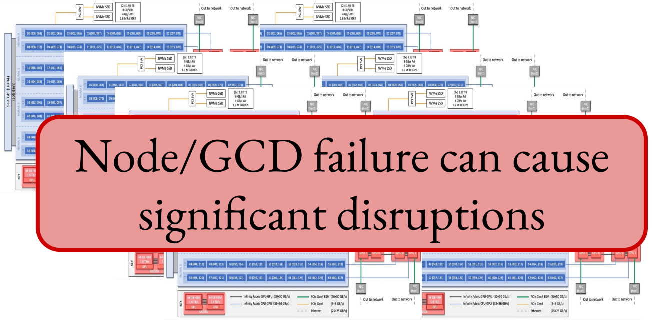
- Compute systems are increasingly heterogeneous
- Compute systems are increasingly hierarchical
- Synchronization bottlenecks restrict scalability



Karlsruhe Institute of Technology

Motivation

- Compute systems are increasingly heterogeneous
- Compute systems are increasingly hierarchical
- Synchronization bottlenecks restrict scalability
- Fault tolerance can improve robustness



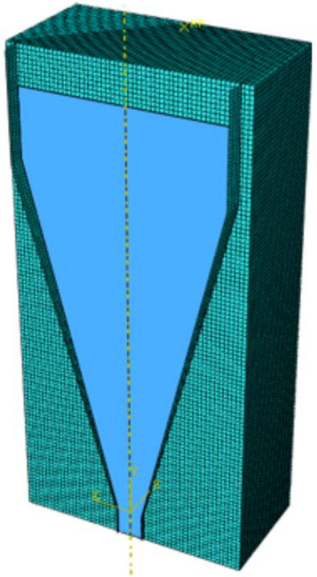
Node/GCD failure can cause significant disruptions

x 128

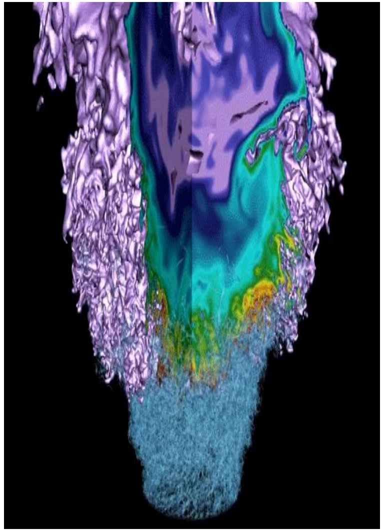
x 74

= 9704 nodes
= 77634 GCDs

A typical computational physics workflow



Study physical behaviour on some domain



Domain (left[1]) and simulation (right[2])

[1]Liu et. al, Oct, 2018
[2]https://amrex-combustion.github.io/

A typical computational physics workflow

Equation describing physics

↓

Non-linear discretization

↓

Linearization of the non-linear iteration

↓

Each step requires a solution of a coupled linear system of the form

$$\mathbf{AX} = \mathbf{B}$$

$$\phi'(t) = \underbrace{R(t,\phi(t))}_{\text{Reaction term}} + \underbrace{F(t,\phi(t))}_{\text{Forcing term}}$$

$$\begin{bmatrix} A_{11} & \cdots & \cdots & A_{1K} \\ \vdots & A_{22} & & \vdots \\ \vdots & & \ddots & \\ A_{K1} & \cdots & & A_{KK} \end{bmatrix} \begin{bmatrix} \Phi_1 \\ \Phi_2 \\ \vdots \\ \Phi_K \end{bmatrix} = \begin{bmatrix} G_1 \\ G_2 \\ \vdots \\ G_K \end{bmatrix}$$

A typical computational physics workflow

Equation describing physics

↓

Non-linear discretization

↓

Linearization of the non-linear iteration

↓

Each step requires a solution of a coupled linear system of the form

$$\mathbf{AX} = \mathbf{B}$$

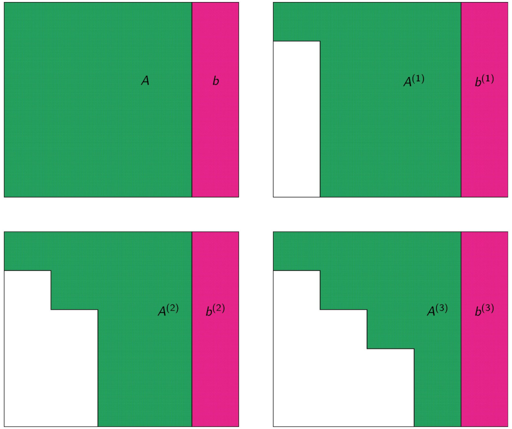
$$\phi'(t) = \underbrace{R(t, \phi(t))}_{\text{Reaction term}} + \underbrace{F(t, \phi(t))}_{\text{Forcing term}}$$

$$\begin{bmatrix} A_{11} & \cdots & \cdots & A_{1K} \\ \vdots & A_{22} & & \vdots \\ \vdots & & \ddots & \\ A_{K1} & \cdots & & A_{KK} \end{bmatrix} \begin{bmatrix} \Phi_1 \\ \Phi_2 \\ \vdots \\ \Phi_K \end{bmatrix} = \begin{bmatrix} G_1 \\ G_2 \\ \vdots \\ G_K \end{bmatrix}$$

Solving linear systems

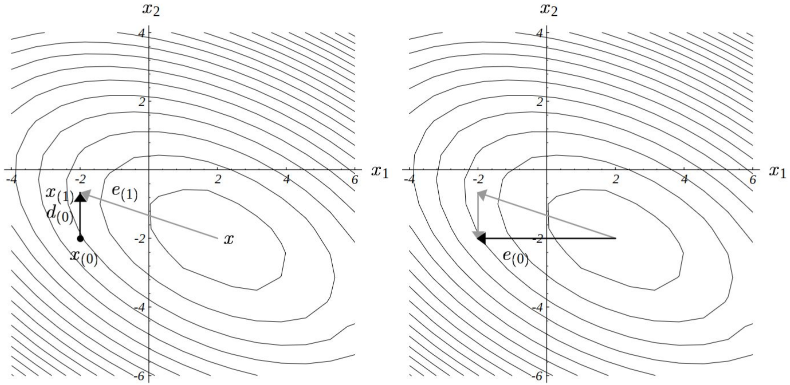
Direct methods

- Provide solution in a fixed number of steps to required precision
- Computationally more expensive ~ $\mathcal{O}(n^3)$



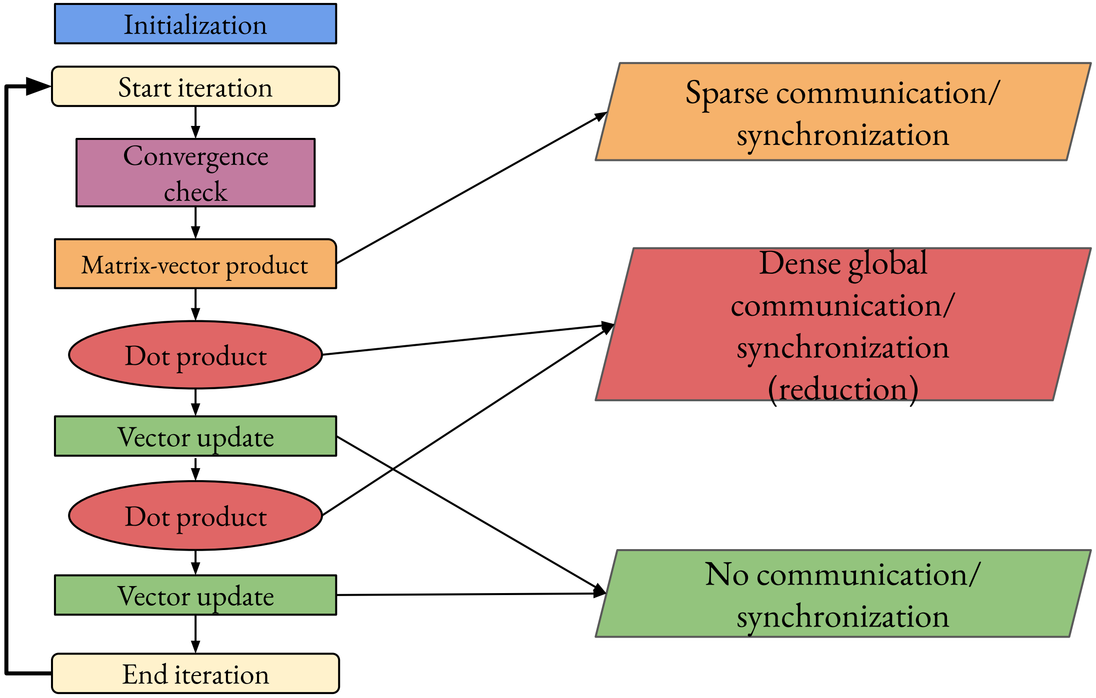
Iterative methods

- Successively approximate solution
- Computationally cheaper, dependent on matrix properties ~ $\mathcal{O}(kn^2)$, $k << n$
- Efficient when exact solution not necessary



Conjugate gradient: Operation breakdown

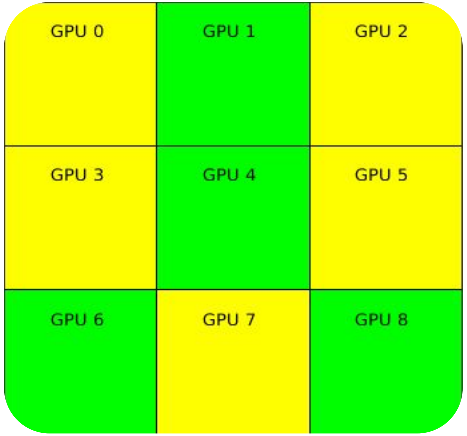
for $i < N_{iter}$ **do**
  if $|\rho| < \tau$ **then**
    break
  end if
  $t \leftarrow Ap$
  $\alpha \leftarrow \frac{\rho}{p \cdot t}$
  $x \leftarrow x + \alpha p$
  $r \leftarrow r - \alpha t$
  $z \leftarrow$ PRECOND(r)
  $\hat{\rho} \leftarrow r \cdot z$
  $p \leftarrow z + \frac{\hat{\rho}}{\rho} \cdot p$
  $\rho \leftarrow \hat{\rho}$
end for



Karlsruhe Institute of Technology

Synchronization-free methods

- Don't explicitly synchronize
- Take latest available data from your neighbor
- Compute on the partially updated data
- Convergence guaranteed only under certain conditions



Asynchronous methods: Convergence conditions[1]

The asynchronous iteration is defined as
$$x^l_{k+1} = \begin{cases} x^l_k & if \quad l \notin \mathcal{I}_k \\ f_l(x^1_{s_1(k)}, \cdots, x^n_{s_n(k)}) & if \quad l \in \mathcal{I}_k \end{cases}$$

With iteration subsets, $\mathcal{I}_k \subseteq \{1, ..., J\}$ and delay subsets $\mathcal{S}_k = s_1(k), \cdots, s_n(k)$, the conditions necessary for convergence are

$$s_j(k) \leq k - 1 \qquad \forall\ j, k$$

Only previous iterations can be used

$$lim_{k \to \infty} s_j(k) = \infty \qquad for\ j = 1, ..., J$$

Use the latest local update

$$the\ sets\ \{k \mid j \in \mathcal{I}_k\}\ are\ unbounded\ for\ j = 1, ..., J$$

Each component needs to be continuously updated

The asynchronous iteration is denoted by $(\mathcal{F}, x_0, \mathcal{I}, \mathcal{S})$

[1]Chazan and Miranker, Chaotic relaxation

Asynchronous methods: Probabilistic model

The asynchronous iteration is defined by $(\mathcal{F}, x_0, \mathcal{I}, \mathcal{S})$, as defined before can be modeled with a probabilistic model. Let $k_{\mathcal{F}}$ be the latest local update and $\mathcal{D} \sim \mathcal{P}(x)$ be some probability distribution where the delays are sampled from. $\mathcal{S}_{\mathcal{P}} = k_{\mathcal{F}} - \mathcal{D}$, therefore gives us iteration from which we incorporate information. The probabilistic model samples the delays, $p(k) \in \mathcal{D} \sim \mathcal{P}$, subject to the conditions,

1. *Positivity:* $p(k) \geq 0$
2. *Validity:* $p(k) \leq k - 1$, *delays cannot be greater than the current iteration.*
3. *Using latest available update:* $p(k+1) - p(k) > 0$
4. *No stagnation:* $p(k+1) \nrightarrow \infty$
5. *Offset:*

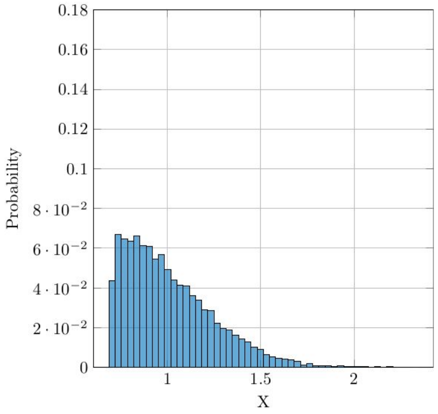
$$p(k) \begin{cases} = 0, & \text{if } \ k < o_{\mathcal{F}} \\ \in \mathcal{P}_S, & \text{otherwise} \end{cases}$$

where $o_{\mathcal{F}} \geq 0$ *is the iteration offset at which the delays are introduced.*

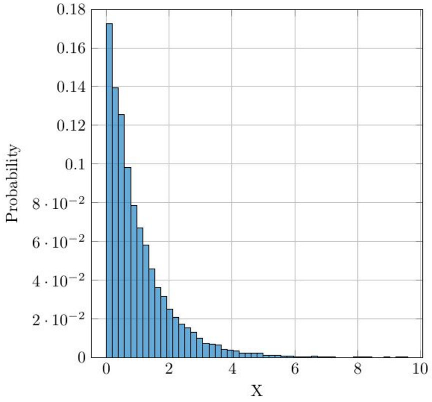
Asynchronous methods: Probability distributions

- Aim to model realistic systems
- Exponential: High probability of no delays
- Normal: Delays distributed around the mean.

Probability distribution	Probabilistic distribution function	Mean/Expected value ($E[X]$)
Half-Normal	$f(x\|a,b) = \sqrt{\frac{2}{\pi}}\frac{1}{b}e^{-\frac{1}{2}(\frac{x-a}{b})^2}; x \geq a$	$\mu = a + b\sqrt{(\frac{2}{\pi})}$
Exponential	$f(x\|\lambda) = \lambda e^{-\lambda x}$	$\mu = \frac{1}{\lambda}$



(a) Half Normal, $\mu = 1.01$, $\sigma = 0.05$



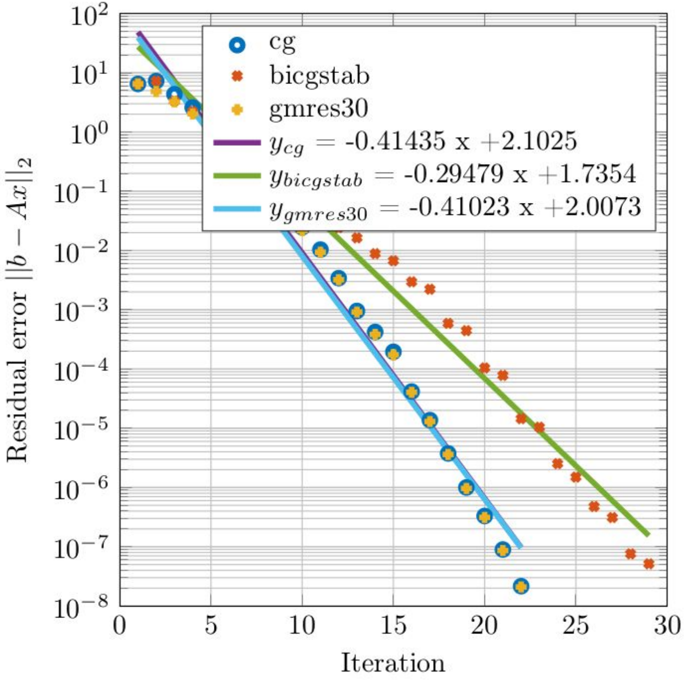
(b) Exponential, $\mu = 1$

Asynchronous methods: Empirical estimation

- Linear fit on the residual, and convergence rate given by the slope of the fit.
- Assuming linear convergence, number of iterations for convergence

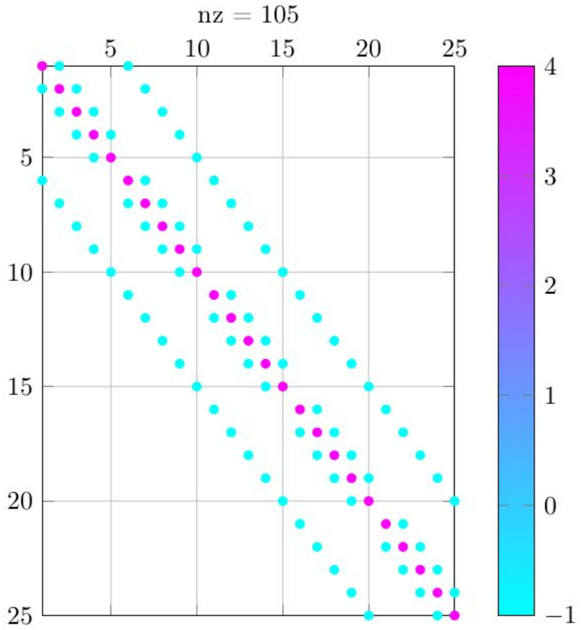
$$i = \frac{\log(\tau_{final})}{\log(\varrho(A))}$$

- Therefore, $0 < \varrho(A) \leq 1$



Asynchronous Richardson: Problem

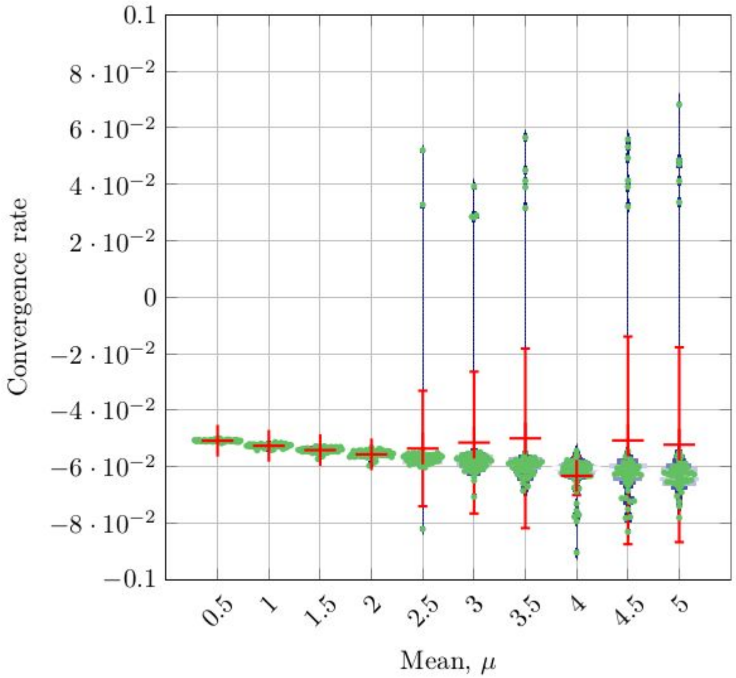
- Laplace 2D problem
- Symmetric and positive definite matrix
- On a 5 x 5 grid, 25 degrees of freedom



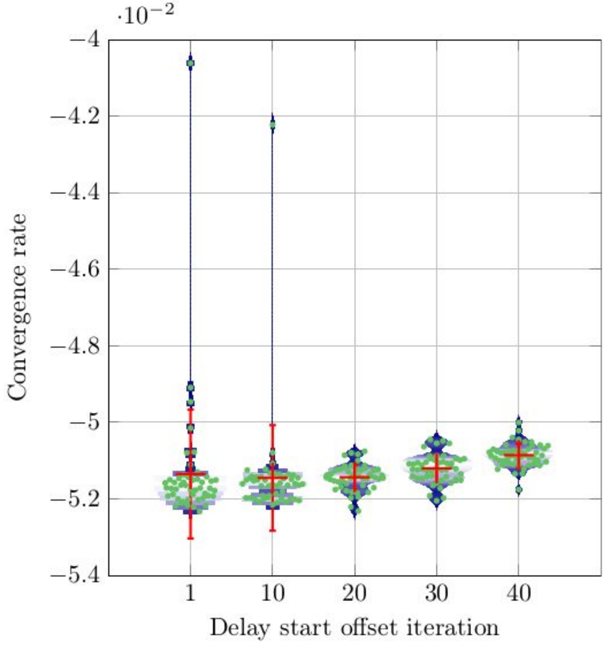
Asynchronous Richardson: Random sampling algorithm

A, b, x, ω
$r \leftarrow b - Ax, e \leftarrow \mathbf{0}$
for $k < N_{iter}$ **do**
  $d \leftarrow$ RANDOM_SAMPLE(k, a, b)
  $r \leftarrow b - Ax(d)$
  if $\|r\| < \tau$ **then**
    break
  end if
  $e \leftarrow$ PRECOND(r)
  $x \leftarrow x + \omega e$
end for

Asynchronous Richardson: Exponential distribution



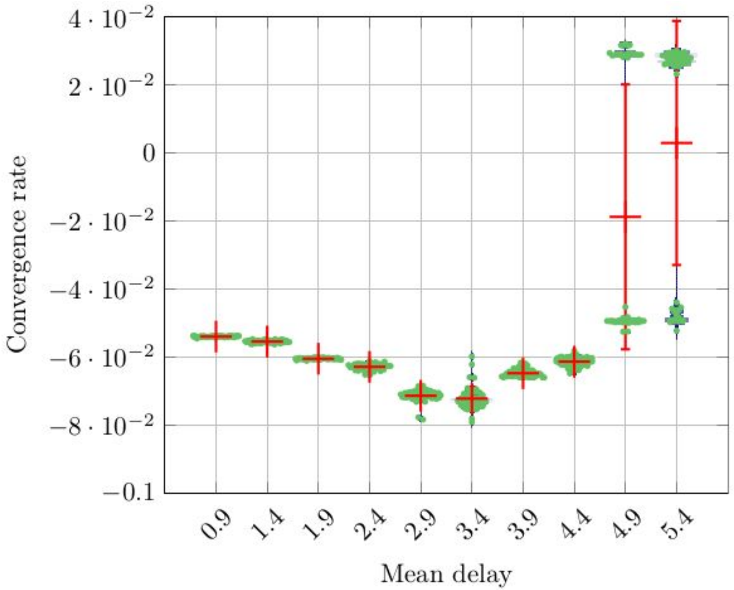
Offset iteration: 40



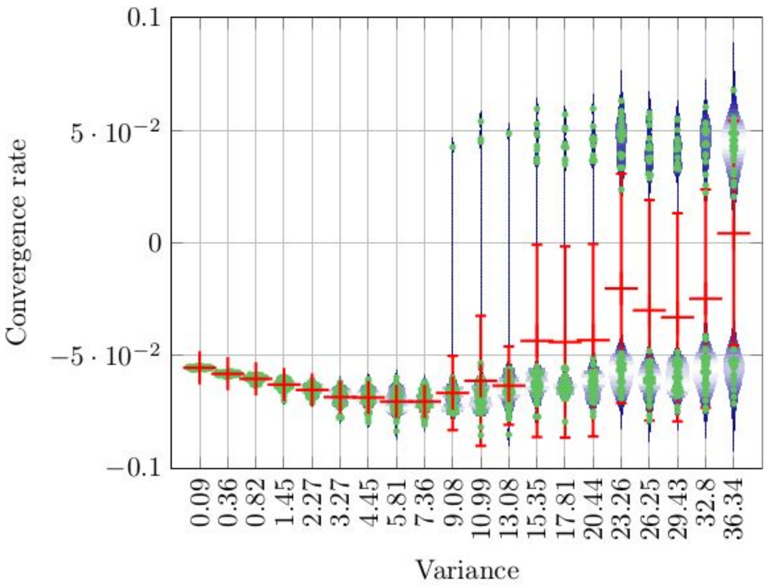
Mean delay: 0.5

$\omega = 0.8.$

Asynchronous Richardson: Half-normal distribution



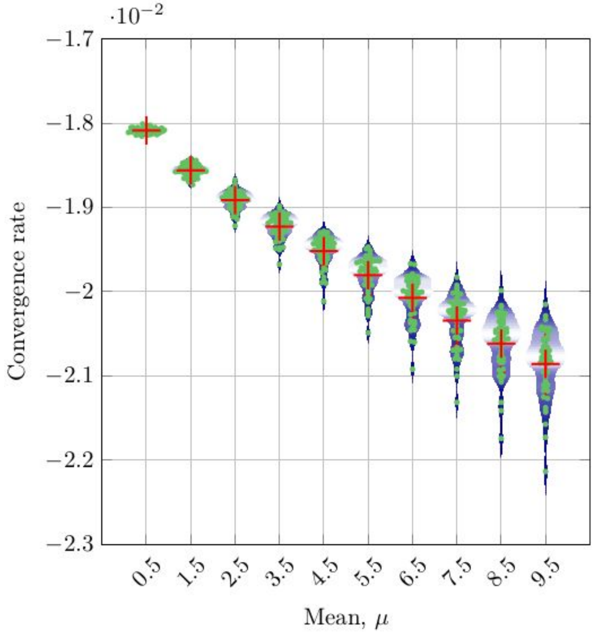
Variance : 0.098



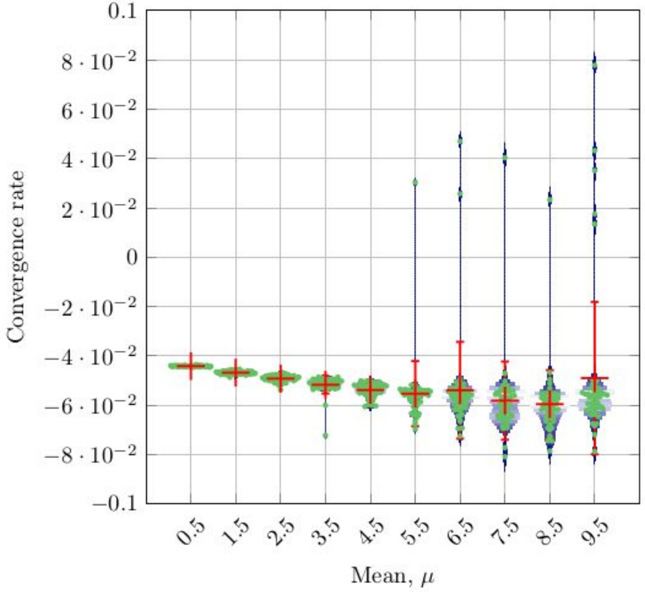
Location parameter: a = 0.5

$\omega = 0.8.$

Asynchronous Richardson: Relaxation parameter



(b) $\omega = 0.3$



(d) $\omega = 0.7$

Asynchronous Schwarz: Definition[1]

The asynchronous iteration is defined as $x^l_{k+1} = \begin{cases} x^l_k & \textit{if} \quad l \notin \mathcal{I}_k \\ \sum_{j=1}^{J} D^{(k)}_{l,j} \quad y^j_{s_j(k)} & \textit{if} \quad l \in \mathcal{I}_k \end{cases}$

With iteration subsets, $\mathcal{I}_k \subseteq \{1, ..., J\}$ and delay subsets $\mathcal{S}_k = s_1(k), \cdots, s_n(k)$, and

$$y^j_{s_j(k)} = M_j^{-1}(N_j x^j_{s_j(k)} + f)$$

With the splittings $A = M_j - N_j \quad j = 1, \cdots, J$, and the weighting matrices

$$D^{(k)}_{l,j} \quad \textit{such that} \sum_{j=1}^{J} D^{(k)}_{l,j} = I \quad \forall l, k$$

The asynchronous iteration is denoted by $(\mathcal{F}_{schw}, x_0, \mathcal{I}, \mathcal{S})$is called the asynchronous Schwarz method.

[1]Frommer and Szyld

Asynchronous Two-level Schwarz: Definition[1]

A coarse level enables robustness for domain decomposition methods, enhancing the exchange of global information. We emulate the action of a global coarse matrix, A_0^{-1}, and incorporate it into the solution as shown below

$$\mathbf{u}^{k+1/2} = \mathbf{u}^k + \sum_{i=1}^{p} R_i^\top D_i A_i^{-1} R_i(\mathbf{f} - A\mathbf{u}^k)$$

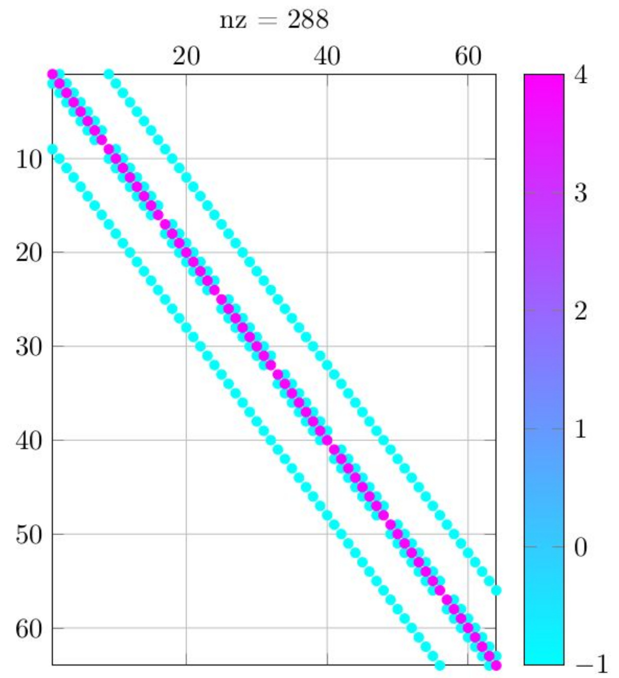
$$\mathbf{u}^{k+1} = \mathbf{u}^{k+1/2} + \boxed{R_0^\top A_0^{-1} R_0(\mathbf{f} - A\mathbf{u}^{k+1/2})}$$

Coarse correction

[1]Szyld et.al

Asynchronous Richardson: Problem

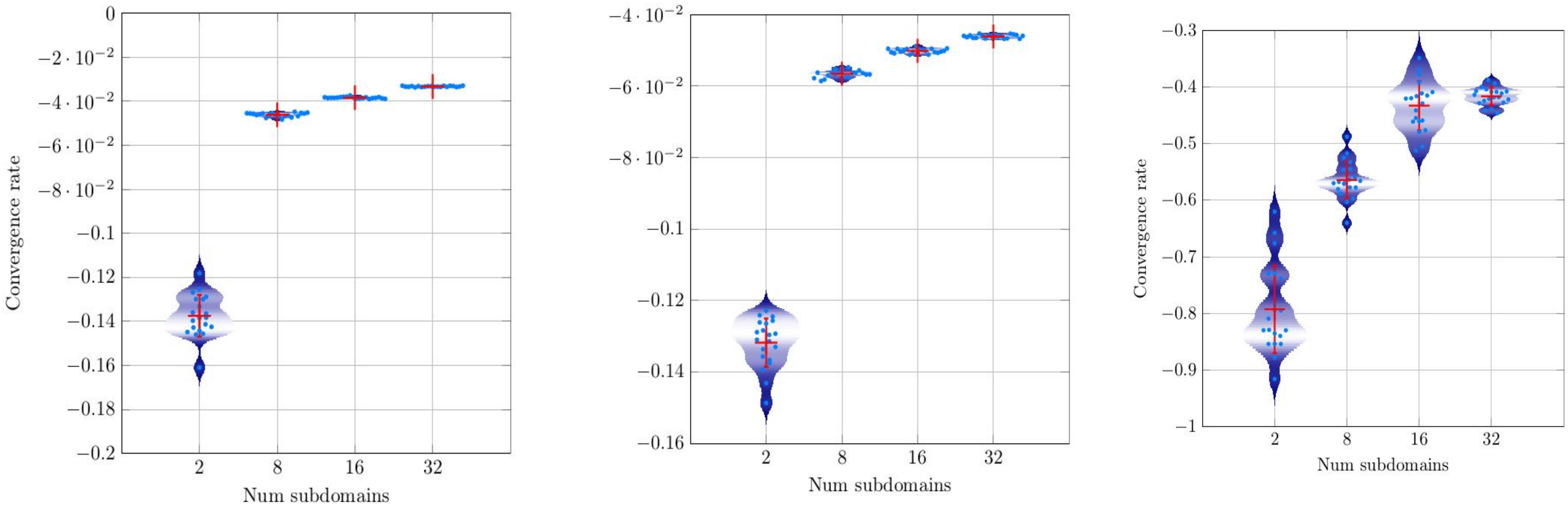
- Laplace 2D problem
- Symmetric and positive definite matrix
- On a 8 x 8 grid, 64 degrees of freedom



Asynchronous Schwarz: Two-level probabilistic model

A, b, x, ω
$r \leftarrow b - Ax, e \leftarrow \mathbf{0}$
for $k < N_{iter}$ **do**
  if $\|r\| < \tau$ **then**
    break
  end if
  for $j < N_{domains}$ **do**
    $d_j \leftarrow s_j(k) \in \mathcal{S}_k$ ▷ Get the sampled delay
    $r_j \leftarrow b_j - A_j x_j^{d_j}$ ▷ Communicate and update global solution with delay
    $e_j = A_{j,j} \backslash r_j$ ▷ Solve locally
    $x_j^{k+1} := x_j^k + e_j$ ▷ Update local solution
  end for
  if coarse_corr == additive **then**
    if mod(f, N_{iter})==0 **then** ▷ Update only for coarse correction frequency, f
      $e_{coarse} = A_{coarse} \backslash Rr$ ▷ Compute coarse correction
      $x^{k+1} := x^{k+1} + \alpha_{coarse} P e_{coarse}$ ▷ Weighted additive coarse correction
    end if
  else if coarse_corr == multiplicative **then**
    $r \leftarrow b - Ax^{k+1}$ ▷ Compute updated residual
    $e_{coarse} = A_{coarse} \backslash Rr$ ▷ Compute coarse correction
    $x^{k+1} := x^{k+1} + P e_{coarse}$ ▷ Multiplicative coarse correction
  end if
end for

Asynchronous Schwarz: Probabilistic model, coarse correct

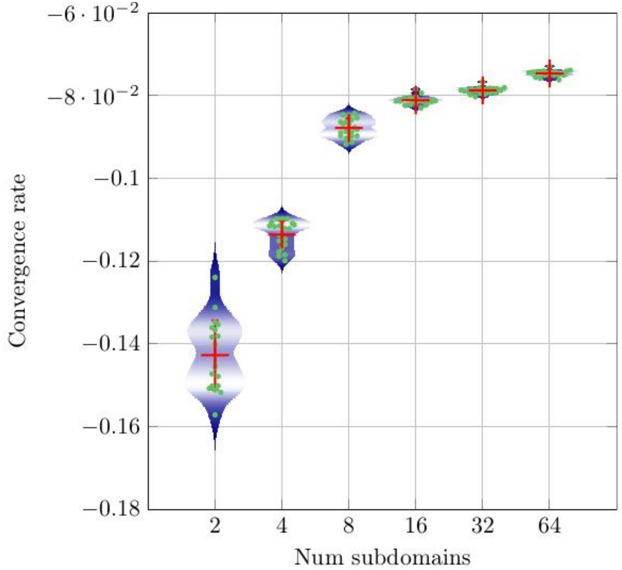


No coarse

Additive coarse, $\alpha = 0.2$

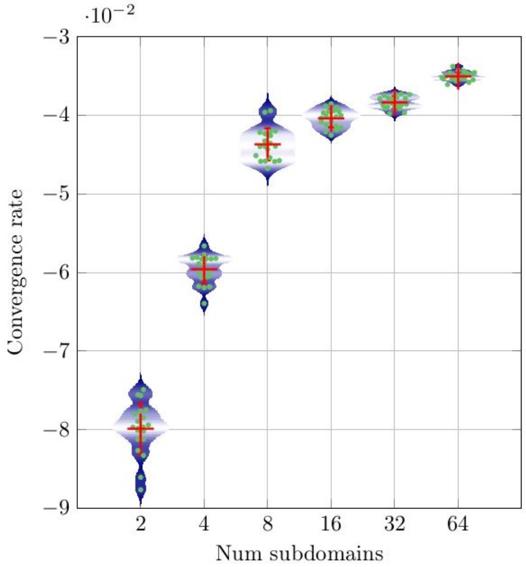
Multiplicative coarse

Asynchronous Schwarz: Probabilistic model, delay



Small mean delay

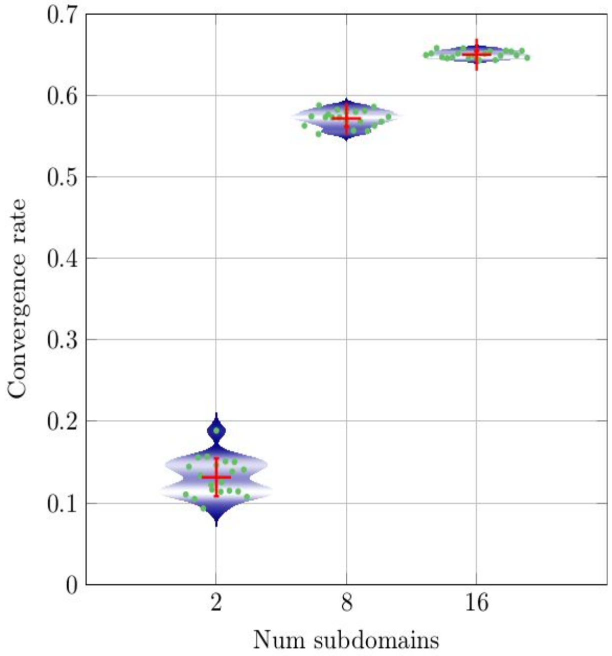
$\mu = 0.32$



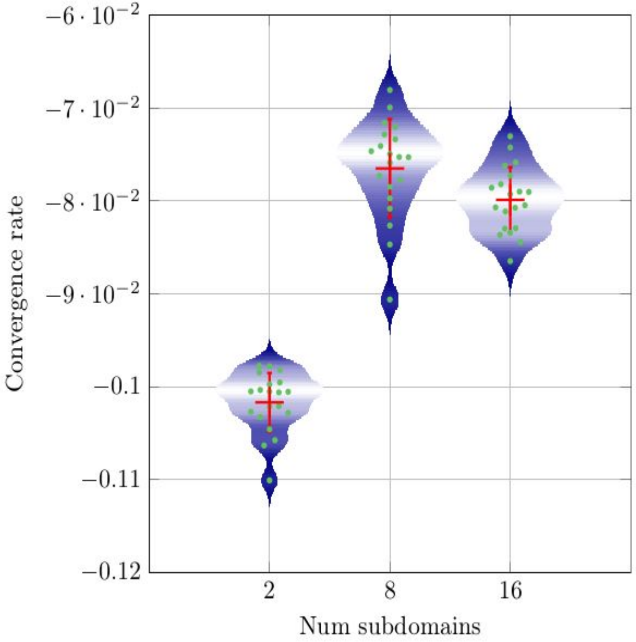
Large mean delay

$\mu = 1.32$

Asynchronous Schwarz: Probabilistic model, coarse freq

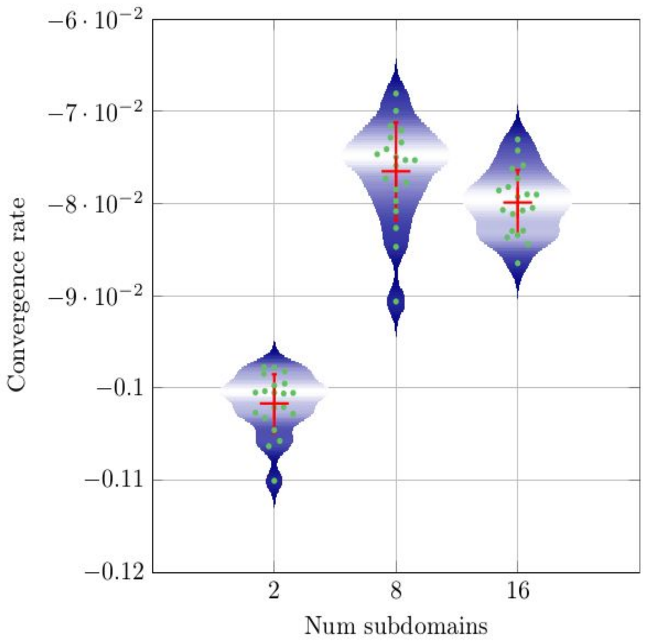


$f = 1$

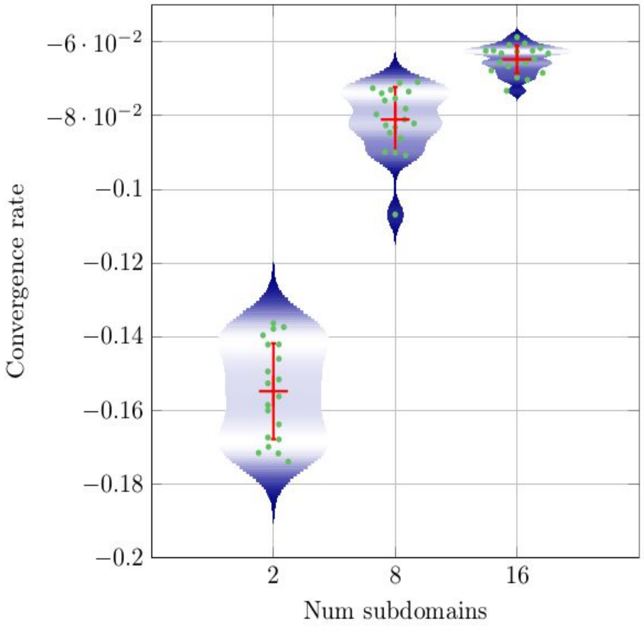


$f = 4$

Asynchronous Schwarz: Probabilistic model, coarse levels



1 coarse level



2 coarse levels

Summary and future work

- Asynchronous methods can be modeled with a probabilistic approach
- Effects of delays can be controlled to balance between convergence and robustness
- Extending these methods to faster converging methods such as Krylov methods can be very interesting.
- Utilizing probabilistic linear solvers[1] within this framework could allow to bundle in uncertainty within the solution and the system instabilities.

[1]Hennig et.al, Probabilistic linear solvers

Asynchronous methods: Richardson convergence[1]

The asynchronous iteration is defined by $(\mathcal{L}, x_0, \mathcal{I}, \mathcal{S})$ with an iteration matrix T, converges if

(a) *converges if there exists a positive vector,* v *and a scalar* $\alpha < 1$ *such that* $|T|v \leq \alpha v$.

(b) *The spectral radius of the iteration matrix,* T, $\rho(|T|) < 1$.

Similarly the Richardson iteration with $(T_\omega, x_0, \mathcal{I}, \mathcal{S})$, $T_\omega = I - \omega M^{-1} A$, $A = M - N$, $M = A_{ii}$ *and* $N = A_{i \neq j}$ converges if

(a) $\rho(|T_\omega|) = \alpha < 1$

(b) $0 < \omega < \frac{2}{1+\alpha}$

[1]Chazan and Miranker, Chaotic relaxation