Multi-GPU distributed PnP-ULA for high-dimensional imaging inverse problems

Maxime Bouton*, P.-A. Thouvenin*, A. Repetti†‡ et P. Chainais*

*Université de Lille, CNRS, Centrale Lille, UMR 9189 CRIStAL, F-59000 Lille, France
†School of Mathematical and Computer Sciences, Heriot-Watt University, Edinburgh EH14 4AS, UK
‡Maxwell Institute for Mathematical Sciences, Edinburgh EH8 9BT, UK

1. Context

Inverse problems : estimate $\overline{\boldsymbol{x}} \in \mathbb{R}^N$ from observations $\boldsymbol{y} \in \mathbb{R}^M$

$$\boldsymbol{y} = \mathcal{A}(\boldsymbol{H}\overline{\boldsymbol{x}}) \quad \rightsquigarrow \quad \pi(\boldsymbol{x} \mid \boldsymbol{y}) \propto \exp\left(-\phi_{\boldsymbol{y}}(\boldsymbol{H}\boldsymbol{x})\right) p(\boldsymbol{x}),$$

with
- $\mathcal{A}$ noise model
- $\boldsymbol{H} \in \mathbb{R}^{M \times N}$ measurement operator
- $\phi_{\boldsymbol{y}} \circ \boldsymbol{H} : \mathbb{R}^N \to \mathbb{R}^M$ data-fidelity term
- p *prior* on $\boldsymbol{x}$

Uncertainty quantification $\Rightarrow$ Monte-Carlo Markov Chains (MCMC)

Estimation quality $\Rightarrow$ learned *prior*, encoded by a neural network

$\rightsquigarrow$ Plug-and-Play (PnP) approaches

Very high-dimensional problems ($M \approx N > 10^6$)
$\Rightarrow$ distributed sampler on *SPMD* architecture (Thouvenin et al. 2024):
- split data and processing on B workers
- restrain communications

2. Towards a distributed PnP-ULA

PnP-ULA (Laumont et al. 2022) :

$$\boldsymbol{z}^{(t+1)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_N)$$
$$\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)} - \delta \boldsymbol{H}^* \nabla \phi_{\boldsymbol{y}}(\boldsymbol{H}\boldsymbol{x}^{(t)}) + \sqrt{2\delta}\boldsymbol{z}^{(t+1)}$$
$$+ \frac{\alpha\delta}{\epsilon}\left(D_\epsilon(\boldsymbol{x}^{(t)}) - \boldsymbol{x}^{(t)}\right) + \frac{\delta}{\lambda}\left(\Pi_{\mathcal{C}}(\boldsymbol{x}^{(t)}) - \boldsymbol{x}^{(t)}\right),$$

with
- D_ϵ pre-trained (deep) Gaussian denoiser, variance ϵ
- $\Pi_{\mathcal{C}}$ projection onto $\mathcal{C} \subset \mathbb{R}^N$, non-empty compact convex set

Proposed distributed strategy: exploit operation locality

Property: f *local* $\Leftrightarrow [f(\boldsymbol{x})]_i = \widetilde{f}_i(\boldsymbol{x}_{[i]})$, $\quad \boldsymbol{x}_{[i]}$: "small" subset of $\boldsymbol{x}$

Assumptions :
- $\phi_{\boldsymbol{y}}$ block-additively separable $\rightsquigarrow (\boldsymbol{y}_b)_{1 \le b \le B} \perp\!\!\!\perp \boldsymbol{x}$
- $\boldsymbol{H}, \nabla\phi_{\boldsymbol{y}}, \Pi_{\mathcal{C}}$ and D_ϵ local $\rightsquigarrow (\boldsymbol{x}_b)_{1 \le b \le B}$ + limited communications

Choice of D_ϵ : Deep Dual Forward-Backward (Le et al. 2024)
- Convolutions $\Rightarrow$ Local ✓
- Few layers $\Rightarrow$ Number of communications ↘

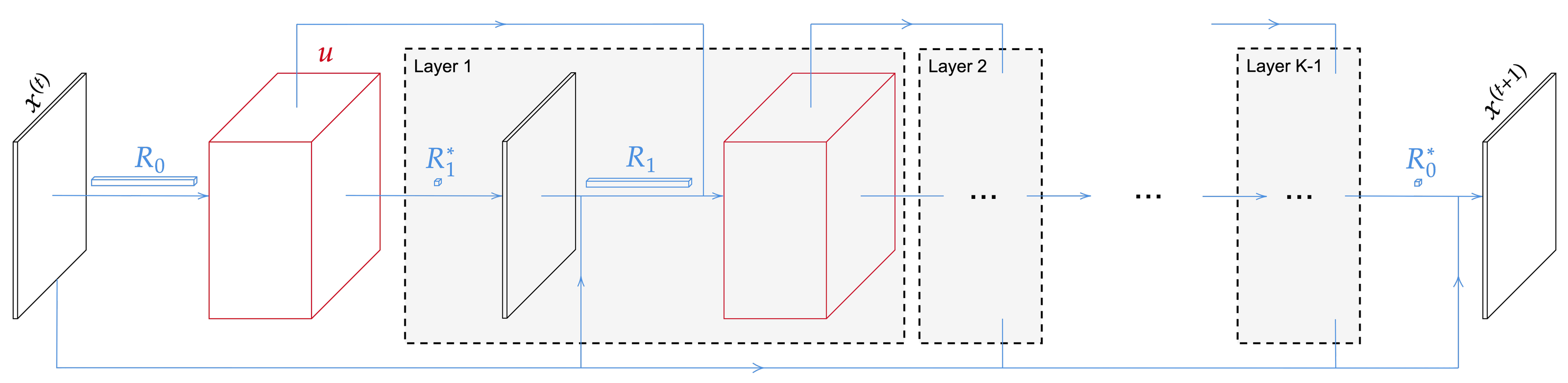


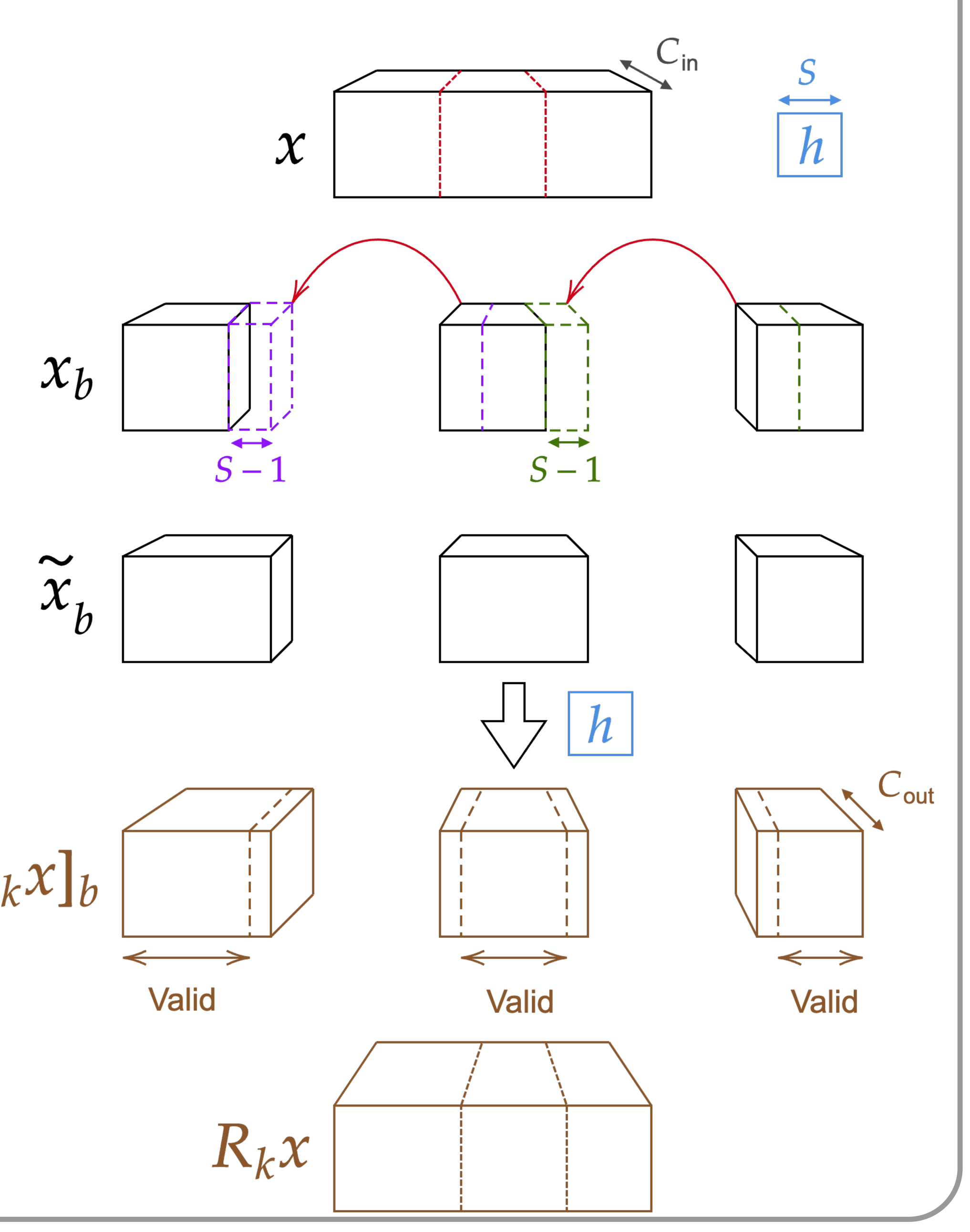
Figure 1: DDFB architecture

Distributing a convolution layer of DDFB :

1 Communications

2 Applying the operator

3 Handling side effects



3. Proposed distributed algorithm

Input: $\boldsymbol{y} \in \mathbb{R}^M, \epsilon > 0, \alpha > 0, \lambda, \delta$

for *each worker* $b \in \{1, \dots, B\}$ **do** in parallel
  Load and store $\boldsymbol{y}_b \in \mathbb{R}^{M_b}$
  Initialize $\boldsymbol{x}_b^{(0)} \in \mathbb{R}^{N_b}$
  for $t = 0$ **to** $T - 1$ **do**
    Communicate to compute $\boldsymbol{g}_b{}^{(t)} = \frac{1}{\sigma^2}\left[\boldsymbol{H}^*\left(\boldsymbol{H}\boldsymbol{x}^{(t)} - \boldsymbol{y}\right)\right]_b$
    // Communicate within each layer of DDFB to compute
    $\boldsymbol{d}_b{}^{(t)} = \left[D_\epsilon(\boldsymbol{x}^{(t)})\right]_b$
    // Local update
    Draw $\boldsymbol{z}_b{}^{(t+1)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{N_b})$
    $\boldsymbol{x}_b{}^{(t+1)} = \boldsymbol{x}_b{}^{(t)} - \delta\boldsymbol{g}_b{}^{(t)} + \frac{\alpha\delta}{\epsilon}\left(\boldsymbol{d}_b{}^{(t)} - \boldsymbol{x}_b{}^{(t)}\right)$
    $+ \frac{\delta}{\lambda}\left(\Pi_{\mathcal{C}_b}(\boldsymbol{x}_b{}^{(t)}) - \boldsymbol{x}_b{}^{(t)}\right) + \sqrt{2\delta}\boldsymbol{z}_b{}^{(t+1)}$

Output: $(\boldsymbol{x}^{(t)})_{1 \le t \le T}$

4. Experiments on synthetic data

Application: deconvolution on colored image, Gaussian noise $\sigma = 0.03$
Comparison: PnP-FB using DRUNet (serial implementation)

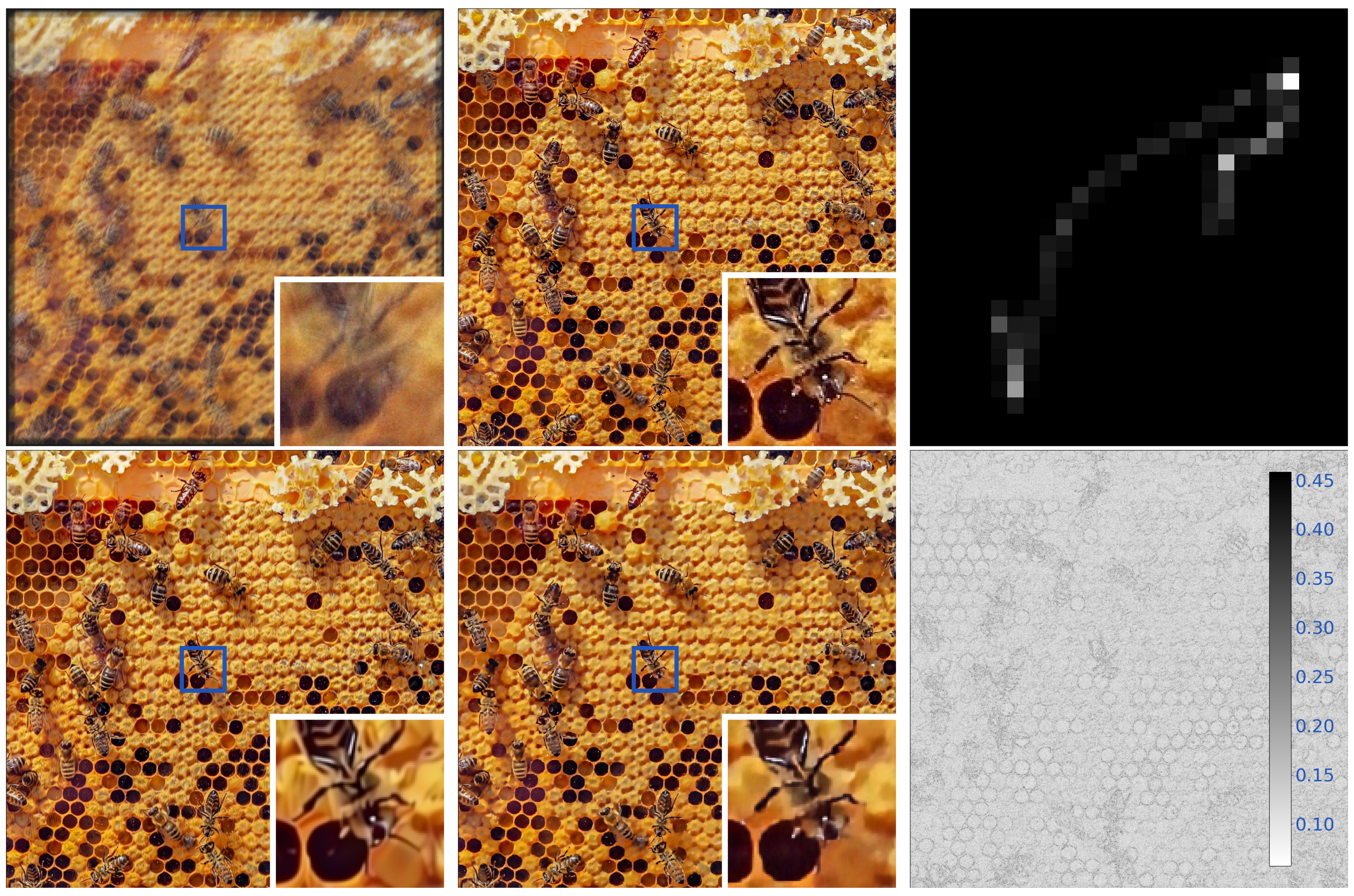


Figure 2: First row: Observations, ground truth, convolution kernel. Second row: MAP (PnP-FB), MMSE (ours, $B = 4$) and credibility interval.

B	Time / iter (ms)	Speedup	rSNR	SSIM
1	199 ± 1.43	1	19.36	0.77
2	135 ± 0.87	1.47	19.37	0.77
4	108 ± 0.68	1.84	19.39	0.77

Table 1: Strong scalability

5. Conclusions

Distributed PnP-ULA sampler
- → same theoretical guarantees as serial counterpart;
- → scalability: Single Program Multiple Data (SPMD) architecture;
- → multi-GPU implementation;
- → constraints on the denoiser structure.

Laumont, Rémi et al. (2022). “Bayesian Imaging Using Plug & Play Priors: When Langevin Meets Tweedie”. In: *SIAM Journal on Imaging Sciences* 15.2, pp. 701–737. arXiv: 2103.04715 [cs, eess, math, stat].

Le, Hoang Trieu Vy, Audrey Repetti, and Nelly Pustelnik (2024). “Unfolded Proximal Neural Networks for Robust Image Gaussian Denoising”. In: *IEEE Trans. Image Process.* 33, pp. 4475–4487.

Thouvenin, Pierre-Antoine, Audrey Repetti, and Pierre Chainais (2024). “A Distributed Split-Gibbs Sampler with Hypergraph Structure for High-Dimensional Inverse Problems”. In: *J. Comput. and Graph. Stat.* 33.3, pp. 814–832.

Work supported by the ANR project “Chaire IA Sherlock” ANR-20-CHIA-0031-01 hold by P. Chainais, the national support within the *programme d’investissements d’avenir* ANR-16-IDEX-0004 ULNE, Région HDF, and the CNRS IEA “DAISHI” hold by P.-A. Thouvenin and A. Repetti. Project provided with computing HPC and storage resources by GENCI at IDRIS on the supercomputer Jean Zay’s V100 partition (grant 2024-AD010615597).