A versatile distributed MCMC algorithm for large scale inverse problems

P.-A. THOUVENIN*, A. REPETTI†‡, P. CHAINAIS*

*University of Lille, CNRS, Centrale Lille, UMR 9189 CRIStAL, France
†Department of Actuarial Mathematics & Stats., Heriot-Watt University, U. K.
‡ Institute of Sensors, Signals and Systems, Heriot-Watt University, U. K.

EUSIPCO 2022, Aug. 29 – Sep. 2, Belgrade

AUGUST 30, 2022

Work supported by the ANR-20-CHIA-0031-01 and ANR-16-IDEX-0004 projects.

Inverse problems

- **Observation model**

$$\boldsymbol{y} = \mathcal{D}(\boldsymbol{A}\boldsymbol{x}),$$

$\boldsymbol{y} \in \mathbb{R}^M$	observations
$\boldsymbol{x} \in \mathbb{R}^N$	unknown parameters (image, ...)
$\boldsymbol{A} \in \mathbb{R}^{M \times N}$	measurement operator
$\mathcal{D} : \mathbb{R}^M \to \mathbb{R}^M$	noise

- **Objective**: find estimate of $\boldsymbol{x}$ from $\boldsymbol{y}$

$$\pi(\boldsymbol{x} \mid \boldsymbol{y}) \propto \exp\Big(-f_{\boldsymbol{y}}(\boldsymbol{A}\boldsymbol{x}) - g(\boldsymbol{B}\boldsymbol{x})\Big),$$

$f_{\boldsymbol{y}} : \mathbb{R}^M \to]-\infty, +\infty]$	data-fidelity
$g : \mathbb{R}^P \to]-\infty, +\infty]$, $\boldsymbol{B} \in \mathbb{R}^{P \times N}$	prior

Challenges

Typical applications (imaging, ...)

- large number of parameters (N large, from 10^3 to 10^8)
- large datasets (M large, $M \approx N$)

Scaling with the dimensions of the problem

- distribute parameters and observations
- decompose inference over multiple *workers*
- limit communication bottlenecks

$\Longrightarrow$ Exploit the structure of the problem

$\Longrightarrow$ Algorithm amenable to parallel implementation?

Distributed state-of-the-art approaches

Optimization literature

- ▶ *Splitting* methods: large class of parallelizable algorithms:
  ADMM (Boyd et al. 2011), primal-dual (Chambolle et al. 2011), ...
- ✗ MAP estimator only
- ✓ Distributed implementation (client-server, SPMD)

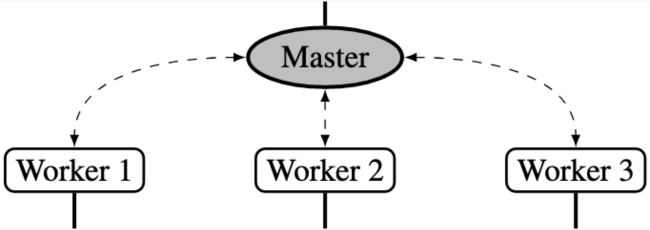
MCMC methods

- ▶ *Asymptotically exact data augmentation* (AXDA) approach (Vono et al. 2021; Rendell et al. 2021)
  Divide to conquer strategy (splitting) *via* data augmentation
- ✓ Estimator (MMSE, MAP, ...) + credibility interval
- ⚠ Distributed implementation (limited to client-server so far)

Client-server vs SPMD distributed architecture

Client-server

- ► each *client* executes a specific task assigned by the *server*
- ► information from clients aggregated periodically on the server
- ✘ communications bottleneck (copies across workers, ...)
- ✘ load balancing issues (if drastically different tasks)

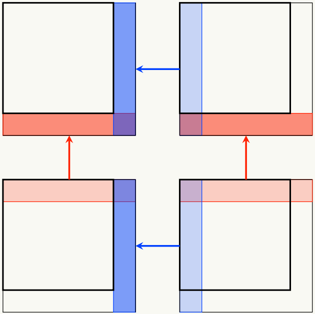


Client-server communications

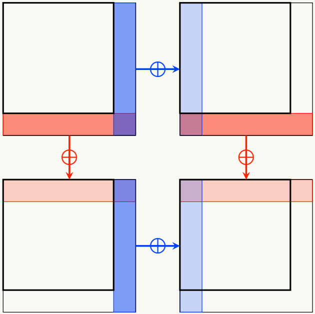
Client-server vs SPMD distributed architecture

Single Program Multiple Data (SPMD) (Darema 2001)

- all workers execute the same task
- ✓ communications between a worker and its neighbours
- ✓ each worker responsible for a chunk of $\boldsymbol{y}$ and $\boldsymbol{x}$ (data locality)



SPMD: exchanging borders

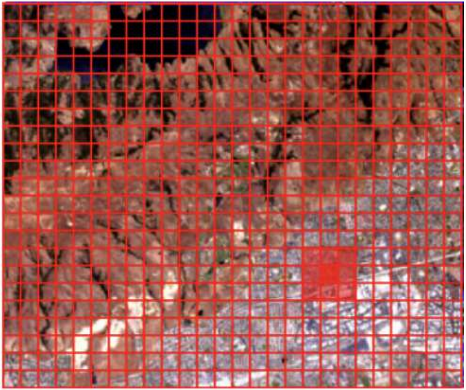


SPMD: aggregating borders

Problem structure

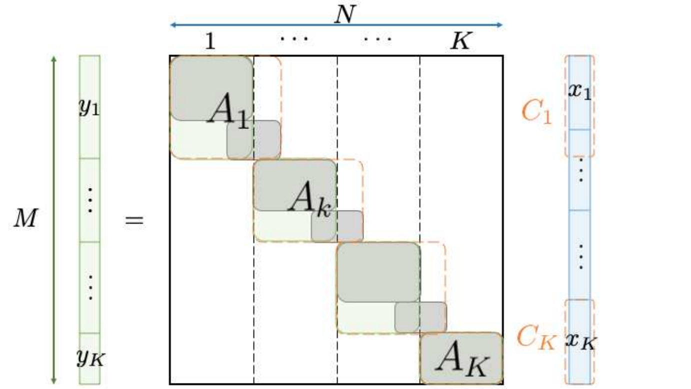
Idea: many problems involve localized operators A and B

e.g., inpainting mask, convolution, Laplacian, gradient, ...



Block-sparse model

Assumption: $\boldsymbol{A}$ and $\boldsymbol{B}$ are block-sparse (localized structure)



- Overlap between contiguous blocks $\ll \lfloor N/K \rfloor$
- $(\boldsymbol{A}_k)_{1 \leq k \leq K}$ (dashed orange), $\boldsymbol{x} = (\boldsymbol{x}_k)_{1 \leq k \leq K}$, $\boldsymbol{x}_k \in \mathbb{R}^{N_k}$

Model (I)

Assumption: $\mathcal{D}$ block-separable: $\mathcal{D}(\boldsymbol{z}) = \big(\mathcal{D}_k(\boldsymbol{z}_k)\big)_{1 \leq k \leq K}$

e.g., additive white Gaussian noise, Poisson noise, ...

$$\boxed{\boldsymbol{y} = \mathcal{D}(\boldsymbol{A}\boldsymbol{x}) = (\boldsymbol{y}_k)_{1 \leq k \leq K}, \text{ where } (\forall k)\ \boldsymbol{y}_k = \mathcal{D}_k(\boldsymbol{A}_k \boldsymbol{C}_k \boldsymbol{x})}$$

$\boldsymbol{C}_k \in \mathbb{R}^{\widetilde{N}_k \times N}$ selection operator, with $N \leq \sum_k \widetilde{N}_k$ (allows overlap in pixel selection)

$\boldsymbol{A}_k \in \mathbb{R}^{M_k \times \widetilde{N}_k}$ local operator

- Additively separable data-fidelity term

$$f_{\boldsymbol{y}}(\boldsymbol{A}\boldsymbol{x}) = \sum_{k=1}^{K} f_{\boldsymbol{y}_k}(\boldsymbol{A}_k \boldsymbol{C}_k \boldsymbol{x}), \quad f_{\boldsymbol{y}_k} \colon \mathbb{R}^{M_k} \to]-\infty, +\infty]$$

Model (II)

- Additively separable data-fidelity term

$$f_{\boldsymbol{y}}(\boldsymbol{A}\boldsymbol{x}) = \sum_{k=1}^{K} f_{\boldsymbol{y}_k}(\boldsymbol{A}_k \boldsymbol{C}_k \boldsymbol{x}), \quad f_{\boldsymbol{y}_k} \colon \mathbb{R}^{M_k} \to]-\infty, +\infty]$$

- g additively separable (*e.g.*, ℓ_1-norm (Laplace prior), $\ell_{1,2}$-norm)

$$g(\boldsymbol{B}\boldsymbol{x}) = \sum_{k=1}^{K} g_k(\boldsymbol{B}_k \boldsymbol{D}_k \boldsymbol{x}) \quad g_k \colon \mathbb{R}^{P_k} \to]-\infty, +\infty]$$

$\boldsymbol{D}_k \in \mathbb{R}^{\overline{N}_k \times N}$ selection operator, $\boldsymbol{B}_k \in \mathbb{R}^{P_k \times \overline{N}_k}$

✓ Problem amenable to SPMD architecture

Algorithm structure compatible with model structure?

Proposed approach

Model of interest

1. Block sparse structure for matrices
2. Additively separable functions

Towards a SPMD sampler

1. Leverage AXDA (Vono et al. 2019b)
   - splitting via approximate data augmentation
2. Gibbs sampler
3. Langevin kernels to handle non-trivial conditional distributions (*e.g.*, PSGLA (Salim et al. 2020), MYULA (Durmus et al. 2018))
4. Parallel implementation (SPMD)

Illustration on image inpainting

- Observation model: additive white Gaussian noise

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{w}, \text{ with } \boldsymbol{w} \sim \mathcal{N}\left(\mathbf{0}_M, \sigma^2 \boldsymbol{I}_{M\times M}\right)$$

- Data fidelity term

$$f_{\boldsymbol{y}_k}(\boldsymbol{A}_k\boldsymbol{C}_k\boldsymbol{x}) = \frac{1}{2\sigma^2}\|\boldsymbol{A}_k\boldsymbol{x}_k - \boldsymbol{y}_k\|_2^2, \quad \boldsymbol{A}_k : \text{ local inpainting mask}$$

- Total variation (TV) prior: $\boldsymbol{B}$ discrete gradient (Chambolle 2004),

$$g_k(\boldsymbol{B}_k\boldsymbol{D}_k\boldsymbol{x}) = \tau\|\boldsymbol{B}_k\boldsymbol{D}_k\boldsymbol{x}\|_{1,2}, \ \tau > 0.$$

Application of AXDA

- Approximate posterior distribution $\widetilde{\pi}_\alpha$
  - $\widetilde{\pi}_\alpha \to \pi$ when $\alpha \to 0$
  - splitting variable $\boldsymbol{z}$, with $p(\boldsymbol{z} \mid \alpha) \propto \exp(-\phi_\alpha(\boldsymbol{z}))$, ϕ_α separable
- (Optional) additional exact augmentation, variable $\boldsymbol{u}$
  - improves mixing of the chain (Vono et al. 2019a)

$$\widetilde{\pi}_\alpha\Big(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u} \mid \boldsymbol{y}, \beta\Big) \propto \exp\Big(-\sum_{k=1}^{K} h_k(\boldsymbol{x}, \boldsymbol{z}_k, \boldsymbol{u}_k; \alpha, \beta)\Big),$$

$$h_k(\boldsymbol{x}, \boldsymbol{z}_k, \boldsymbol{u}_k; \alpha, \beta) = f_{\boldsymbol{y}_k}(\boldsymbol{A}_k \boldsymbol{x}_k) + \tau \|\boldsymbol{z}_k\|_{2,1} + \frac{1}{2\beta}\|\boldsymbol{u}_k\|_2^2$$

$$+ \underbrace{\frac{1}{2\alpha}\|\boldsymbol{B}_k \boldsymbol{D}_k \boldsymbol{x} - \boldsymbol{z}_k + \boldsymbol{u}_k\|_2^2}_{\phi_{k,\alpha}(\boldsymbol{z}_k - \boldsymbol{u}_k)}.$$

Proposed SPMD sampler

For $t \in \{0, \dots, N_{\mathrm{MC}} - 1\}$, on worker k, each sample generated as

```
// Update x_k with PSGLA kernel (Salim et al. 2020)
```

Communications induced by $\boldsymbol{D}_k$ to compute $\nabla_{\boldsymbol{x}} h_k$

$\boldsymbol{x}_k^{(t+1)} = \boldsymbol{x}_k^{(t)} - \gamma \nabla_{\boldsymbol{x}} h_k\Big(\boldsymbol{x}^{(t)}, \boldsymbol{z}_k^{(t)}, \boldsymbol{u}_k^{(t)}; \alpha, \beta\Big) + \sqrt{2\gamma}\xi_k,$

with $\xi_k \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{N_k \times N_k})$

Communications to compute $\boldsymbol{B}_k \boldsymbol{D}_k \boldsymbol{x}^{(t+1)}$

```
// Update z with PSGLA kernel (Salim et al. 2020)
```

$\boldsymbol{z}_k^{(t+1)} = \mathrm{prox}_{\eta g_k}\Big(\boldsymbol{z}_k^{(t)} - \frac{\eta}{\alpha}(\boldsymbol{z}_k^{(t)} + \boldsymbol{B}_k \boldsymbol{D}_k \boldsymbol{x}^{(t+1)} - \boldsymbol{u}_k^{(t)}) + \sqrt{2\eta}\zeta_k\Big),$

with $\zeta_k \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{M_k \times M_k})$

```
// Sample u from its full conditional
```

$\boldsymbol{u}_k^{(t+1)} \mid \boldsymbol{x}^{(t+1)}, \boldsymbol{z}_k^{(t+1)} \sim \mathcal{N}\Big(\frac{\nu}{\alpha}(\boldsymbol{z}_k^{(t+1)} - \boldsymbol{B}_k \boldsymbol{D}_k \boldsymbol{x}^{(t+1)}), \nu \mathbf{I}_{P_k \times P_k}\Big)$

Experiment

Simulation settings

- $M = \lfloor 0.6N \rfloor$ observations, σ^2 such that SNR = 40 dB
- $\mathrm{N_{MC}} = 10^4$ samples, $\mathrm{N_{bi}} = 5 \times 10^3$ burn-in, $(\alpha, \beta, \tau) = (9, 1, 0.2)$
- Strong scaling (fixed problem size, increasing K)

Parallel setting

- HPC computer grid from University of Lille[1]
- Single node: two 2.1 GHz, 18-core, Intel Xeon E5-2695 v4 series processors (36 CPU cores in total)
- Parallelization: `mpi4py` (Dalcin et al. 2021) library
  (1 worker = 1 process on a CPU core)

[1] `https://hpc.univ-lille.fr/`

Strong scaling experiment

K	SNR (MMSE)	SNR (MAP)	Time per iter. ($\times 10^{-3}$ s)	Speedup	Runtime (s)
1 (Vono et al. 2019a)	23.33	22.45	65.56 (2.08)	0.19	262.20
1	23.45	22.95	12.21 (0.63)	1.00	61.04
2	23.46	22.88	6.07 (0.42)	2.01	30.37
4	23.48	22.88	3.50 (0.21)	3.49	17.50
8	23.44	22.86	1.93 (0.77)	6.33	9.63
16	23.48	22.90	1.08 (2.35)	11.30	5.38

Table 1: Strong scaling experiment results.

Experiment results

(a) Ground truth

(b) MMSE (Vono 2019)

(c) MMSE (prop.)

(a) Obs.

(b) CI (Vono 2019)

(c) CI (prop.)

Conclusions and perspectives

Conclusions: SPMD-distributed sampler (PSGLA within Gibbs)

- ✓ quality comparable to (Vono et al. 2019a);
- ✓ lower runtime (distribution flexibility);
- ✓ strong scaling behaviour.

Conclusions and perspectives

Conclusions: SPMD-distributed sampler (PSGLA within Gibbs)

- ✓ quality comparable to (Vono et al. 2019a);
- ✓ lower runtime (distribution flexibility);
- ✓ strong scaling behaviour.

Perspectives

- → infer AXDA parameters α, β;
- → more general applications
  - ⇝ inverse problems on hyper-graphs;
  - ⇝ assessment on multiple nodes;
- → extension to handle asynchronous communications.

Conclusions and perspectives

Conclusions: SPMD-distributed sampler (PSGLA within Gibbs)

- ✓ quality comparable to (Vono et al. 2019a);
- ✓ lower runtime (distribution flexibility);
- ✓ strong scaling behaviour.

Perspectives

- → infer AXDA parameters α, β;
- → more general applications
  - ⇝ inverse problems on hyper-graphs;
  - ⇝ assessment on multiple nodes;
- → extension to handle asynchronous communications.

Thank you for your attention.

A versatile distributed MCMC algorithm for large scale inverse problems

P.-A. Thouvenin*, A. Repetti†‡, P. Chainais*

*University of Lille, CNRS, Centrale Lille, UMR 9189 CRIStAL, France
†Department of Actuarial Mathematics & Stats., Heriot-Watt University, U. K.
‡ Institute of Sensors, Signals and Systems, Heriot-Watt University, U. K.

EUSIPCO 2022, Aug. 29 – Sep. 2, Belgrade

August 30, 2022

Work supported by the ANR-20-CHIA-0031-01 and ANR-16-IDEX-0004 projects.

Backup slides

- References

Backup slide

References I

Boyd, Stephen et al. (2011). “Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers”. In: *Foundations and Trends® in Machine Learning* 3.1, pp. 1–122.

Chambolle, Antonin (Jan. 2004). “An algorithm for total variation minimization and applications”. In: *J. Math. Imag. Vision* 20.1, pp. 89–97.

Chambolle, Antonin and Thomas Pock (May 2011). “A First-Order Primal-Dual Algorithm for Convex Problems with Applications to Imaging”. In: *J. Math. Imag. Vision* 40.1, pp. 120–145.

Dalcin, Lisandro and Yao-Lung L. Fang (2021). “mpi4py: Status Update After 12 Years of Development”. In: *IEEE Comput. Sci. Eng.* 23.4, pp. 47–54.

Backup slide

References II

Darema, Frederica (2001). "The SPMD Model: Past, Present and Future". In: *Recent Advances in Parallel Virtual Machine and Message Passing Interface*. Ed. by Yiannis Cotronis and Jack Dongarra. Berlin, Heidelberg, pp. 1–1.

Durmus, Alain, Eric Moulines, and Marcelo Pereyra (2018). "Efficient Bayesian computation by proximal Markov chain Monte Carlo: when Langevin meets Moreau". In: *SIAM J. Imaging Sci.* 11.1, pp. 473–506.

Rendell, Lewis J. et al. (2021). "Global Consensus Monte Carlo". In: *J. Comput. and Graph. Stat.* 30.2, pp. 249–259.

Salim, Adil and Peter Richtàrik (2020). "Primal Dual Interpretation of the Proximal Stochastic Gradient Langevin Algorithm". In: *Adv. in Neural Information Processing Systems*. Vol. 33, pp. 3786–3796.

Backup slide

References III

Vono, M., N. Dobigeon, and P. Chainais (Mar. 2019a). “Split-and-augmented Gibbs sampler - Application to large-scale inference problems”. In: *IEEE Trans. Signal Process.* 67.6, pp. 1648–1661.

Vono, Maxime, Nicolas Dobigeon, and Pierre Chainais (2019b). “Asymptotically exact data augmentation: models, properties and algorithms”. In: arXiv preprint.

— (2021). “Asymptotically Exact Data Augmentation: Models, Properties, and Algorithms”. In: *J. Comput. and Graph. Stat.* 30.2, pp. 335–348.