Plug-and-Play split Gibbs sampler: embedding deep generative priors in Bayesian inference

Pierre Chainais

N. Dobigeon, F. Cœurdoux (IRIT)

May 30, 2024

Chaire IA Sherlock

Collaborators

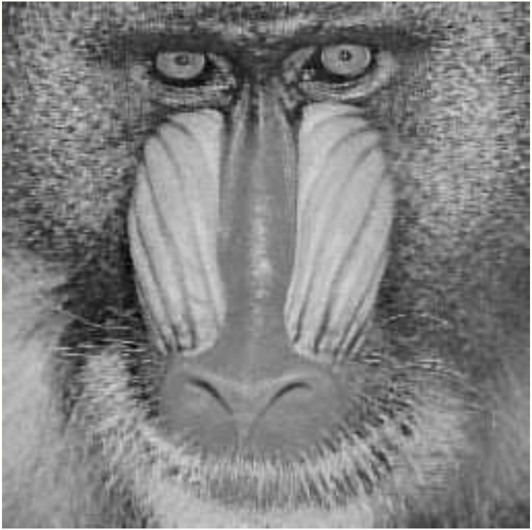
Nicolas Dobigeon, Florentin Cœurdoux, Maxime Vono

Motivations: inverse problems

Image deblurring

Motivations: inverse problems

Image deblurring



Motivations: inverse problems

Image inpainting

Motivations: inverse problems

Image inpainting

Motivations: inverse problems

Confidence intervals

Motivations

$$\mathbf{y} = \mathcal{A}(\mathbf{x}) + \mathbf{n}$$

- **solve complex** ill-posed ML or inverse problems
- **big** data in high dimensions
- **good** performances
- **fast** inference algorithms
- **credibility intervals**

with maybe some additional options such as:

- **privacy** preserving
- **distributed** computing

Bayesian approach + Monte Carlo method

+ machine learning?

Motivations

$$\mathbf{y} = \mathcal{A}(\mathbf{x}) + \mathbf{n}$$

- **solve complex** ill-posed ML or inverse problems
- **big** data in high dimensions
- **good** performances
- **fast** inference algorithms
- **credibility intervals**

with maybe some additional options such as:

- **privacy** preserving
- **distributed** computing

Bayesian approach + Monte Carlo method

+ machine learning?

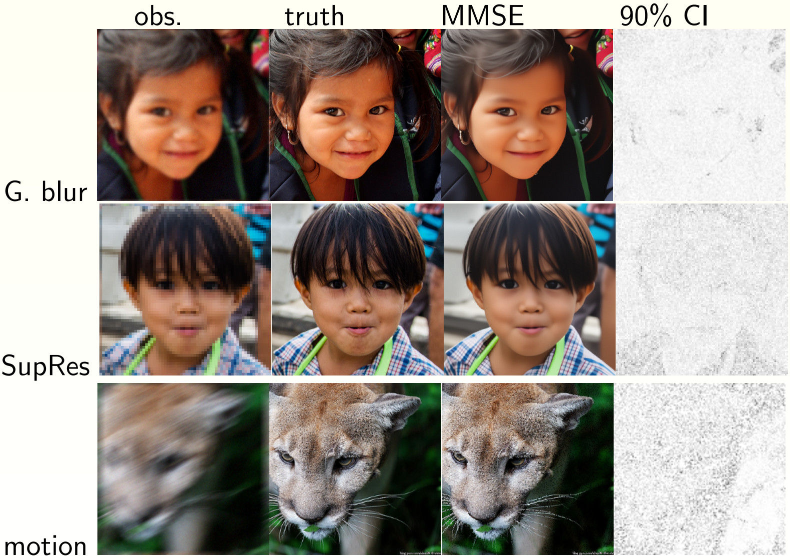
The usual toolbox of inference

- **Optimization**:
  - problem ⇒ loss function
  - efficient algorithms
  - theoretical guarantees
  - interpretability / functional analysis
- **Bayesian approaches**:
  - probabilitic models
  - uncertainty quantification
- **Machine learning** (deep):
  - adaptive ⇒ relevant
  - outstanding performance

toward the best of all worlds?

PnP-SGS for inverse problems

State of the art performance + credibility intervals

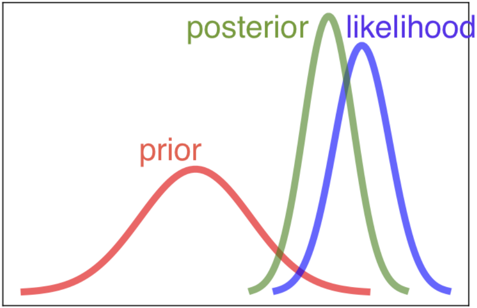


Bayesian inference

$\mathbf{y}$: available data = observations

$\mathbf{x}$: unknown object of interest

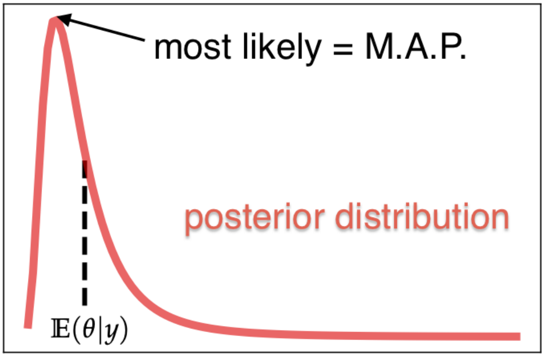
Prior	$\times$	Likelihood	$\longrightarrow$	Posterior
$\mathbf{x} \sim \pi(\mathbf{x})$		$\mathbf{y}\vert\mathbf{x} \sim \pi(\mathbf{y}\vert\mathbf{x})$		$\mathbf{x}\vert\mathbf{y} \sim \pi(\mathbf{x}\vert\mathbf{y})$



Bayesian inference

$\mathbf{y}$: available data = observations

$\mathbf{x}$: unknown object of interest

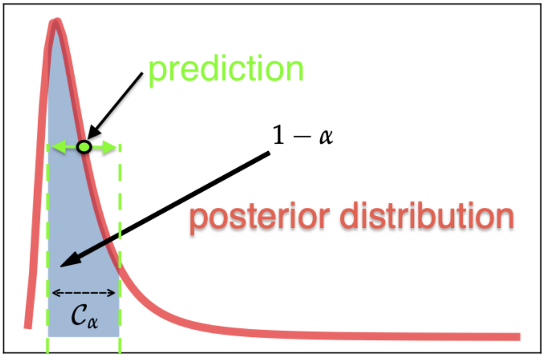


Bayesian estimators $$\arg\min_{\hat{\mathbf{x}}} \int L(\mathbf{x}, \hat{\mathbf{x}}) \pi(\mathbf{x}|\mathbf{y}) \mathrm{d}\mathbf{x}$$

Bayesian inference

y: available data = observations

x: unknown object of interest



Credibility regions $\mathcal{C}_\alpha$ $\int_{\mathcal{C}_\alpha} \pi(\mathbf{x}|\mathbf{y})\mathrm{d}\mathbf{x} = 1-\alpha$

Bayesian inference

$\mathbf{y}$: available data = observations

$\mathbf{x}$: unknown object of interest

Image to restore y

Bayesian inference

$\mathbf{y}$: available data = observations

$\mathbf{x}$: unknown object of interest

Restored image $\hat{\mathbf{x}}$

Bayesian inference

$\mathbf{y}$: available data = observations

$\mathbf{x}$: unknown object of interest

Confidence intervals $\Delta\hat{\mathbf{x}}$

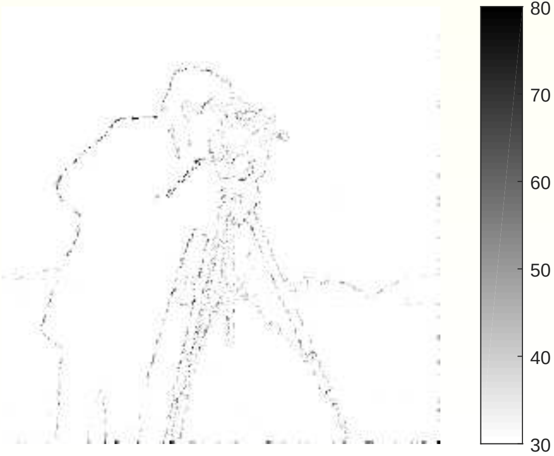


Illustration: 2D Gaussian mixture model - MTM steps

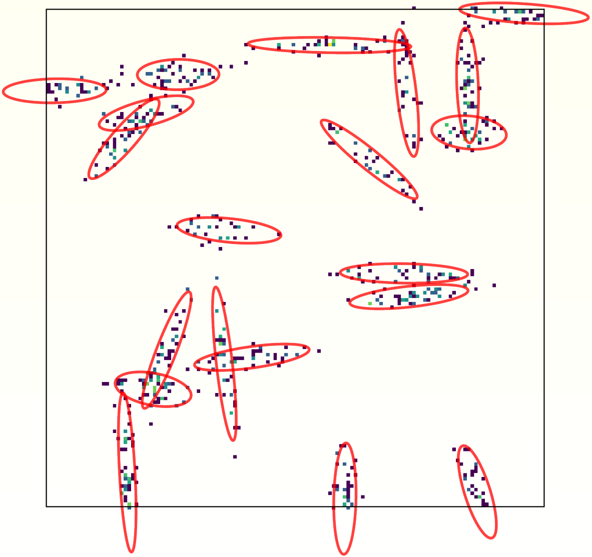
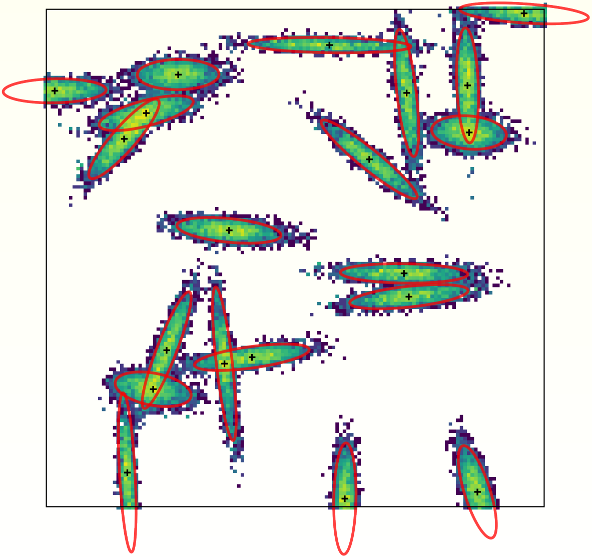


Illustration: 2D Gaussian mixture model - MALA + MTM



Bayesian approach for imaging inverse problems

Linear Gaussian inverse problems

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{n}$$

$\mathbf{A}$ = damaging operator (blur, binary mask...)

$\mathbf{n} \sim \mathcal{N}\left(\mathbf{0}_d, \sigma^2 \mathbf{I}_d\right)$ = noise.

Posterior distribution $\quad p\left(\mathbf{x}|\mathbf{y}\right) \propto \exp\left[-\frac{1}{2\sigma^2}\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 - \lambda g(\mathbf{x})\right]$

where $g(\mathbf{x})$ = prior knowledge on solutions.

Direct sampling of images is challenging

1. generally **high dimension** of the image, e.g. 1 million pixels,
2. **generality** of the prior distribution of probability, cf. $g(\mathbf{x})$

Bayesian approach for imaging inverse problems

Linear Gaussian inverse problems

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{n}$$

$\mathbf{A}$ = damaging operator (blur, binary mask...)

$\mathbf{n} \sim \mathcal{N}\left(\mathbf{0}_d, \sigma^2 \mathbf{I}_d\right)$ = noise.

Posterior distribution $\quad p\left(\mathbf{x}|\mathbf{y}\right) \propto \exp\left[-\frac{1}{2\sigma^2}\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 - \lambda g(\mathbf{x})\right]$

where $g(\mathbf{x})$ = prior knowledge on solutions.

Direct sampling of images is challenging

1. generally **high dimension** of the image, e.g. 1 million pixels,
2. **generality** of the prior distribution of probability, cf. $g(\mathbf{x})$

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{n}$$

$\mathbf{A}$ = damaging operator (blur, binary mask...)

$\mathbf{n} \sim \mathcal{N}\left(\mathbf{0}_d, \sigma^2 \mathbf{I}_d\right)$ = noise.

Posterior distribution $\quad p\left(\mathbf{x}|\mathbf{y}\right) \propto \exp\left[-\frac{1}{2\sigma^2}\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 - \lambda g(\mathbf{x})\right]$

where $g(\mathbf{x})$ = prior knowledge on solutions.

Direct sampling of images is challenging

1. generally **high dimension** of the image, e.g. 1 million pixels,
2. **generality** of the prior distribution of probability, cf. $g(\mathbf{x})$

Natural images as stochastic processes

Conjecture by D. Mumford & B. Gidas :

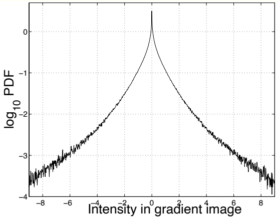
There exists simply described stochastic models for images :

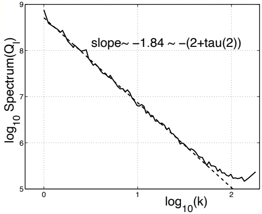
1. *which assign high likelihood to any 'natural' image of the world we live in,*
2. *whose random samples have the 'look and feel' of natural images, i.e. make you look twice to see if you recognize something in them.*

Our purpose

Two main properties :

non Gaussian **scale invariant**: $S(k) \sim 1/k^{2-\eta}$



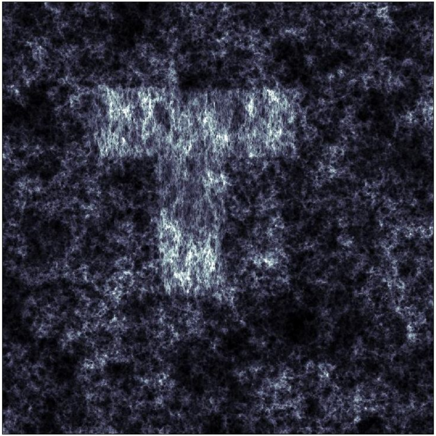


[Field’87, Ruderman’94, Grenander & Srivastava’01...]

How to build stochastic image processes
with non Gaussian statistics and a scale invariant behavior ?

with some desirable properties like *homogeneity*, *isotropy*...

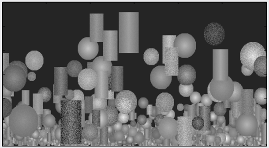
Examples



inserting an inhomogeneity

Physical interpretation

Several models use a distribution of elementary objects like the *transported generator model* by Grenander & Srivastava'01 or the model by Chi'98.



Compound Poisson Cascades can be interpreted as the superposition of transparent objects of random sizes:

$$\log Q_\ell(\mathbf{x}) = \sum_{(\mathbf{x}_i, r_i) \in \mathcal{C}_\ell(\mathbf{x})} \underbrace{\log W_i}_{\textit{transparency}} \cdot \underbrace{f\left(\frac{\mathbf{x}-\mathbf{x}_i}{r_i}\right)}_{\textit{object of size } r_i} + K$$

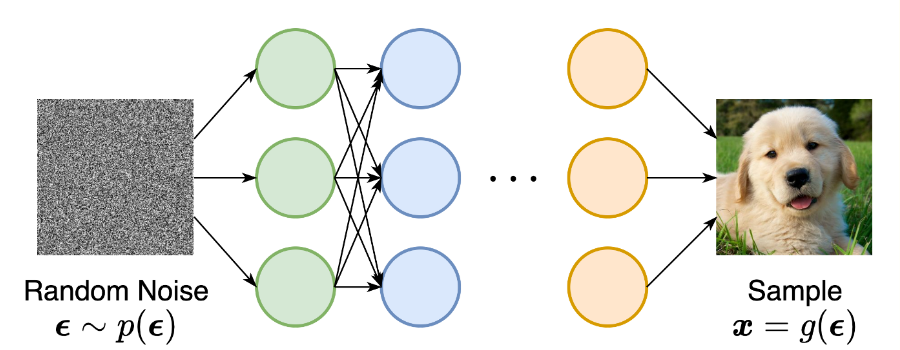
- distribution of sizes $\propto \frac{1}{r^3}$
- positions = Poisson point process $(\mathbf{x}_i, r_i)$

[Matheron'68, Grenander'99, Huang & Mumford'99, Gousseau'01]...

Capitalizing on machine learning

Expressivity of deep neural networks

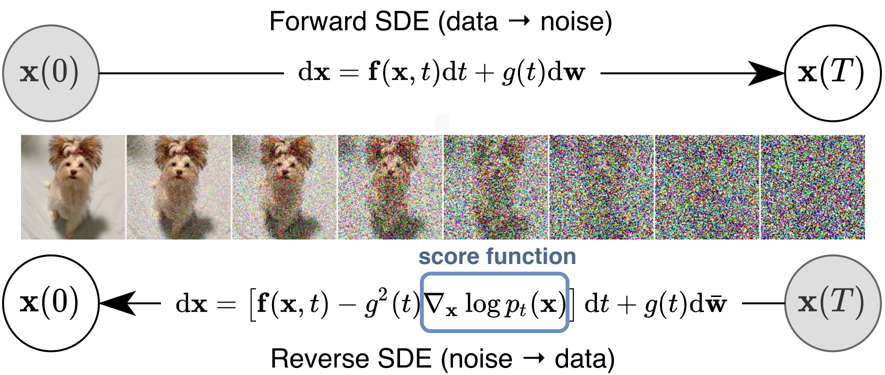
- **Deep learning**
  - strong expressivity
  - very efficient for supervised learning
  - large data set needed



Capitalizing on machine learning

Expressivity of deep neural networks

- **Deep learning**
  - strong expressivity
  - very efficient for supervised learning
  - large data set needed



Capitalizing on machine learning

Expressivity of deep neural networks

- **Deep learning**
  - strong expressivity
  - very efficient for supervised learning
  - large data set needed

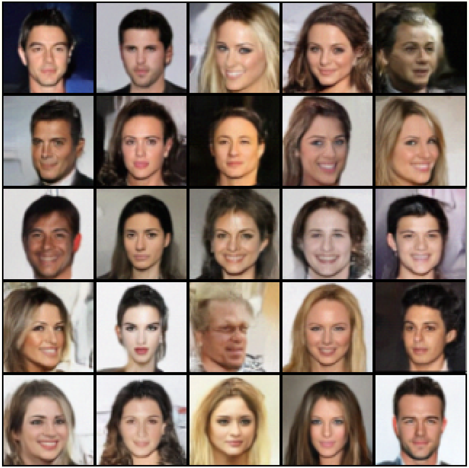
GO TO ANIMATION

Sampling using normalizing flows (deep learning)

F. Cœurdoux's PhD, N. Dobigeon - IRIT

(Deep) **Learning changes of variables** (optimal transport)

- **MALAFlow**: sampling in the Gaussian latent space



Coeurdoux et al. (2023) preprint

Split Gibbs sampling (SGS) for inverse problems

Linear Gaussian inverse problems

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{n}$$

$\mathbf{A}$ = damaging operator (blur, binary mask...)

$\mathbf{n} \sim \mathcal{N}\left(\mathbf{0}_d, \sigma^2 \mathbf{I}_d\right)$ = noise.

Posterior distribution $\quad p\left(\mathbf{x}|\mathbf{y}\right) \propto \exp\left[-\frac{1}{2\sigma^2}\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 - \lambda g(\mathbf{x})\right]$

where $g(\mathbf{x})$ = prior knowledge on solutions.

Direct sampling of images is challenging

1. generally **high dimension** of the image, e.g. 1 million pixels,
2. **generality** of the prior distribution of probability, cf. $g(\mathbf{x})$

Split Gibbs sampling (SGS)

Posterior distrib. $$p(\mathbf{x}|\mathbf{y}) \propto \exp\left[\underbrace{-\frac{1}{2\sigma^2}\|\mathbf{y}-\mathbf{A}\mathbf{x}\|_2^2}_{f_1(\mathbf{x})} - \underbrace{\lambda g(\mathbf{x})}_{f_2(\mathbf{x})}\right]$$

$$p(\mathbf{x}|\mathbf{y}) \propto \exp\left[-f_1(\mathbf{x}) - f_2(\mathbf{x})\right]$$

$$\Downarrow$$

$$\pi(\mathbf{x},\mathbf{z}|\mathbf{x}=\mathbf{z}) \propto \exp\left[-f_1(\mathbf{x}) - f_2(\mathbf{z})\right] \text{ such that } \mathbf{x}=\mathbf{z}$$

$$\Downarrow$$

$$\pi_\rho(\mathbf{x},\mathbf{z}) \propto \exp\left[-f_1(\mathbf{x}) - f_2(\mathbf{z}) - \frac{1}{2\rho^2}\|\mathbf{x}-\mathbf{z}\|_2^2\right]$$

Vono et al. (2019a)

Split Gibbs sampling (SP): conditional distributions

Full conditional distributions under the split distribution π_ρ:

$$\pi_\rho(\mathbf{x}|\mathbf{z}) \propto \exp\left(-f_1(\mathbf{x}) - \frac{1}{2\rho^2}\|\mathbf{x}-\mathbf{z}\|_2^2\right)$$

$$\pi_\rho(\mathbf{z}|\mathbf{x}) \propto \exp\left(-f_2(\mathbf{z}) - \frac{1}{2\rho^2}\|\mathbf{x}-\mathbf{z}\|_2^2\right)$$

Note that f_1 and f_2 are now **split** in **2 distinct distributions**

"easy to use" thanks to state of the art sampling methods

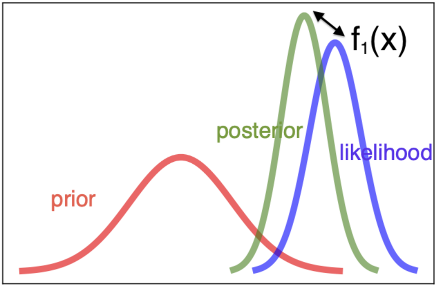
Vono et al. (2019a)

Split Gibbs sampling (SP): conditional distributions

Full conditional distributions under the split distribution π_ρ:

$$\pi_\rho(\mathbf{x}|\mathbf{z}) \propto \exp\left(-f_1(\mathbf{x}) - \frac{1}{2\rho^2}\|\mathbf{x}-\mathbf{z}\|_2^2\right)$$

$$\pi_\rho(\mathbf{z}|\mathbf{x}) \propto \exp\left(-f_2(\mathbf{z}) - \frac{1}{2\rho^2}\|\mathbf{x}-\mathbf{z}\|_2^2\right)$$

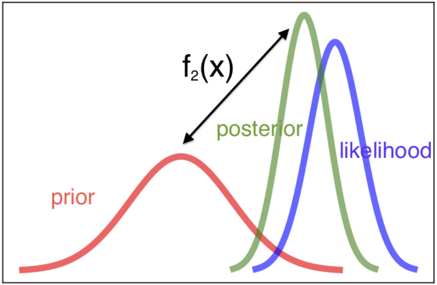


Split Gibbs sampling (SP): conditional distributions

Full conditional distributions under the split distribution π_ρ:

$$\pi_\rho(\mathbf{x}|\mathbf{z}) \propto \exp\left(-f_1(\mathbf{x}) - \frac{1}{2\rho^2}\|\mathbf{x}-\mathbf{z}\|_2^2\right)$$

$$\pi_\rho(\mathbf{z}|\mathbf{x}) \propto \exp\left(-f_2(\mathbf{z}) - \frac{1}{2\rho^2}\|\mathbf{x}-\mathbf{z}\|_2^2\right)$$



Split Gibbs sampling (SP): linear inverse problems

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n}$$

where

- $\mathbf{H}$ = forward operator
- $\mathbf{n}$ = noise with covariance $\boldsymbol{\Omega}^{-1}$

Likelihood:

$$p(\mathbf{y} \mid \mathbf{x}) \propto \exp\left[\underbrace{-\frac{1}{2}(\mathbf{H}\mathbf{x} - \mathbf{y})^T \boldsymbol{\Omega}(\mathbf{H}\mathbf{x} - \mathbf{y})}_{f_1(\mathbf{x})}\right]$$

Splitted Gibbs sampling (SP): linear inverse problems

Full conditional distributions under the split distribution π_ρ:

$$\begin{cases} \pi_\rho(\mathbf{x}|\mathbf{z}) & \propto & \exp\left[-\underbrace{f_1(\mathbf{x})}_{\text{quadratic}} - \frac{1}{2\rho^2}\|\mathbf{x}-\mathbf{z}\|_2^2\right] \\ \pi_\rho(\mathbf{z}|\mathbf{x}) & \propto & \exp\left[-\underbrace{f_2(\mathbf{z})}_{\text{prior}} - \frac{1}{2\rho^2}\|\mathbf{x}-\mathbf{z}\|_2^2\right] \end{cases}$$

Important notice: $\pi_\rho(\mathbf{z}|\mathbf{x})$ is the posterior of **a denoiser** !

Splitted Gibbs sampling (SP): linear inverse problems

Full conditional distributions under the split distribution π_ρ:

$$\begin{cases} \pi_\rho(\mathbf{x}|\mathbf{z}) & \propto \quad \mathcal{N}\left(\mathbf{x}; \boldsymbol{\mu}_{\mathbf{x}}, Q_{\mathbf{x}}^{-1}\right) \\ \pi_\rho(\mathbf{z}|\mathbf{x}) & \propto \quad DDPM(t_n^*, \mathbf{z}|\mathbf{x}) \end{cases}$$

Likelihood = Gaussian distribution

$$\begin{cases} \mathbf{Q}_{\mathbf{x}} = \mathbf{H}^T \boldsymbol{\Omega} \mathbf{H} + \frac{1}{\rho^2} \mathbf{I}_N \\ \boldsymbol{\mu}_{\mathbf{x}} = \mathbf{Q}_{\mathbf{x}}^{-1} \left(\mathbf{H}^T \boldsymbol{\Omega} \mathbf{y} + \frac{1}{\rho^2} \mathbf{z} \right) \end{cases}$$

DDPM = Denoising Diffusion Probabilistic Model

Plug-and-Play and splitting: **PnP-SGS**

F. Cœurdoux's PhD, N. Dobigeon

- **PnP-SGS**: using a deep denoiser as a prior in SGS

SGS uses Gibbs sampling from conditional posteriors

$$\begin{cases} p(\mathbf{x} \mid \mathbf{y}, \mathbf{z}) & \propto & \exp\left[-f_1(\mathbf{y}, \mathbf{x}) - \frac{1}{2\rho^2}\|\mathbf{x}-\mathbf{z}\|_2^2\right] \\ p(\mathbf{z} \mid \mathbf{x}) & \propto & \exp\left[-f_2(\mathbf{z}) - \frac{1}{2\rho^2}\|\mathbf{x}-\mathbf{z}\|_2^2\right] \end{cases}$$

where $p(\mathbf{z} \mid \mathbf{x}) =$ posterior of a denoiser with reg. $\propto f_2(\mathbf{z})$

$$\Downarrow$$

use a stochastic denoiser to sample from $\mathbf{z}$

PnP-SGS using a DDPM

DDPM: Denoising Diffusion Probabilistic Models

The forward diffusion process adds noise from $t-1$ to t:

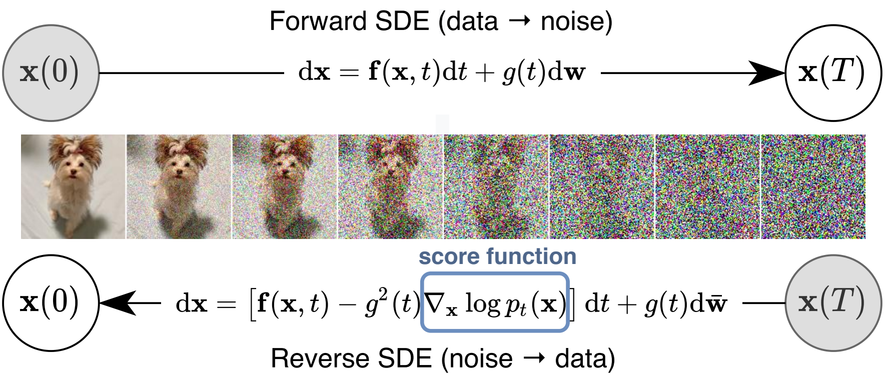
$$p(\mathbf{u}_t \mid \mathbf{u}_{t-1}) = \mathcal{N}\left(\mathbf{u}_t; \sqrt{1-\beta(t)}\mathbf{u}_{t-1}, \beta(t)\mathbf{I}\right)$$

Learn the backward SDE denoiser from t to $t-1$:

$$q_{\boldsymbol{\theta}}(\mathbf{u}_{t-1} \mid \mathbf{u}_t) = \mathcal{N}(\mathbf{u}_{t-1}; \boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{u}_t, t), \boldsymbol{\Sigma}_{\boldsymbol{\theta}}(\mathbf{u}_t, t))$$

PnP-SGS using a DDPM

- **Deep learning**
  - strong expressivity
  - very efficient for supervised learning
  - large data set needed



PnP-SGS using a DDPM

Starting from noise-free image $\mathbf{u}_0$:

$$p\left(\mathbf{u}_t \mid \mathbf{u}_0\right) = \mathcal{N}\left(\mathbf{u}_t; \sqrt{\bar{\alpha}(t)}\mathbf{u}_0, \alpha(t)\mathbf{I}\right)$$

where $\alpha(t) = \prod_{j=1}^{t}\left(1-\beta(j)\right)$ and $\bar{\alpha}(t) = 1 - \alpha(t)$.

$\Rightarrow$ Gaussian noise with variance $\alpha(t^*)$

$$p\left(\mathbf{z} \mid \mathbf{x}\right) \propto \exp\left[-f_2(\mathbf{z}) - \frac{1}{2\rho^2}\left\|\mathbf{x}-\mathbf{z}\right\|_2^2\right]$$

$=$ Gaussian denoiser

$\Longrightarrow \mathbf{z} =$ backward process from $\mathbf{u}_{t^*} = \mathbf{x} \Longrightarrow p\left(\mathbf{z} \mid \mathbf{x}\right)$

PnP-SGS using a DDPM

Remarks:

- noise level $\alpha(t^*) \iff$ unique instant t^*
- the larger t^*, the noisier the image $\mathbf{z}_{t^*}$.

Given a current sample $\mathbf{x}^{(n)}$ of SGS with noise level σ_n^2

$$t^* = \alpha^{-1}(\widehat{\sigma_n^2})$$

using any good conventional estimator $\widehat{\sigma^2} = \Phi(\mathbf{x}^{(n)})$

- larger $t^* \Rightarrow$ higher regularization

(Donoho and Johnstone 1994; Guo et al. 2021; Li et al. 2022)

Inpainting task

Plug-and-Play and splitting: **PnP-SGS** - F. Cœurdoux's PhD, N. Dobigeon - IRIT

Original image - noisy masked - PnP-ADMM - PnP-SGS - 90% cred. int.

Inpainting task

Plug-and-Play and splitting: **PnP-SGS** - F. Cœurdoux's PhD, N. Dobigeon - IRIT

original image

Inpainting task

Plug-and-Play and splitting: **PnP-SGS** - F. Cœurdoux's PhD, N. Dobigeon - IRIT

noisy masked image

PnP-SGS MMSE

Inpainting task

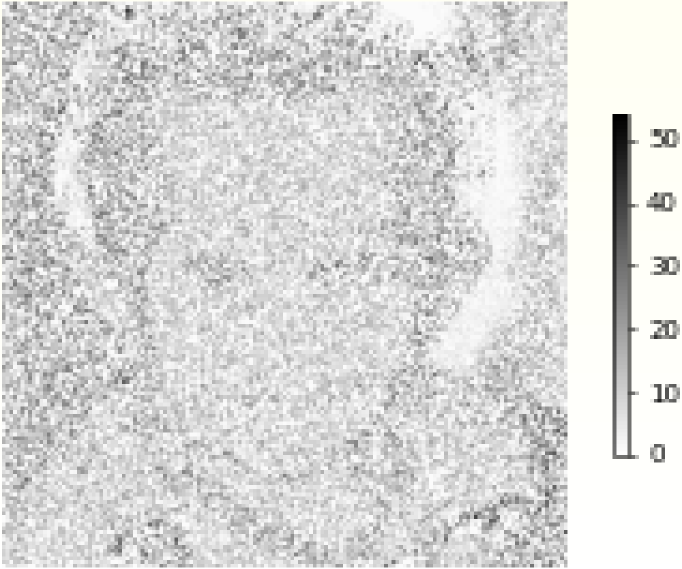
Plug-and-Play and splitting: **PnP-SGS** - F. Cœurdoux's PhD, N. Dobigeon - IRIT

original image

Inpainting task

Plug-and-Play and splitting: **PnP-SGS** - F. Cœurdoux's PhD, N. Dobigeon - IRIT

90% credibility intervals (PnP-SGS)



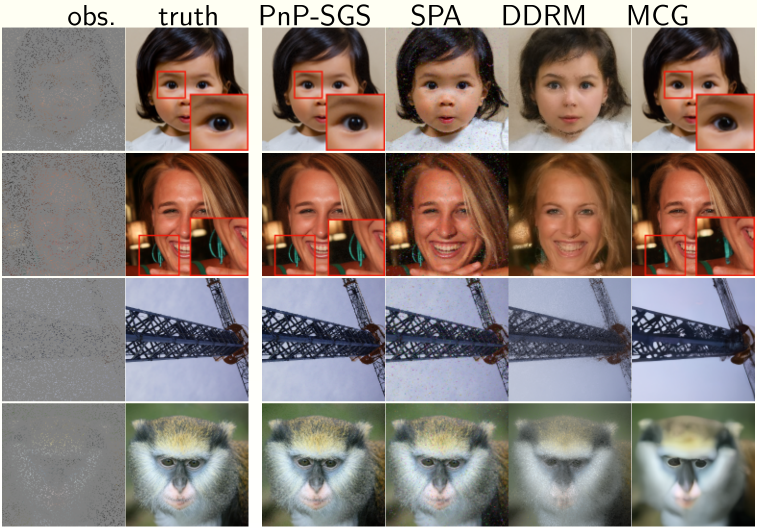
Inpainting task: comparisons on FFHQ

Runtime for each algorithm in Wall-clock time
(computed with a single GTX 2080Ti GPU).

Method	Wall-clock time (s)	Ref.
PnP-ADMM	3.63	Chan et al. (2016)
Score-SDE	36.71	Song et al. (2022)
DDRM	2.03	Kawar et al. (2022)
DPIR	4.18	Zhang et al. (2021)
SGS-ULA	218.90	Vono et al. (2019b)
MCG	80.10	Chung et al. (2023)
DPS	43.89	Chung et al. (2023)
PnP-SGS	13.81	(IEEE Trans. Image Proc. 2024)

Inpainting task on FFHQ 256×256 & Imagenet images

FFHQ (2 top rows) and Imagenet (2 bottom rows)



FFHQ 256 × 256 data set: image reconstruction

		PnP-SGS	SPA	TV-ADMM	PnP-ADMM	DPIR	Score-SDE	DDRM	MCG	DPS
Inpainting	PSNR ↑	**32.59**	24.09	22.03	8.41	24.41	13.52	9.19	21.57	25.23
	SSIM ↑	**0.913**	0.524	0.784	0.325	0.809	0.437	0.319	0.751	0.851
	FID ↓	37.36	71.12	181.56	123.61	52.73	76.54	69.71	**29.26**	38.82
	LPIPS ↓	**0.144**	0.785	0.463	0.692	0.398	0.612	0.587	0.286	0.262
Deblurring (Gaussian)	PSNR ↑	**27.96**	23.17	22.37	24.93	26.09	7.12	23.36	6.72	24.25
	SSIM ↑	**0.837**	0.499	0.801	0.812	0.820	0.109	0.767	0.051	0.811
	FID ↓	**59.667**	78.67	186.74	90.42	80.18	109.07	74.92	101.2	62.72
	LPIPS ↓	**0.331**	0.452	0.507	0.441	0.392	0.403	0.332	0.340	0.444
Deblurring (motion)	PSNR ↑	**28.46**	17.73	21.36	24.65	27.33	6.58	N/A	6.72	20.92
	SSIM ↑	**0.828**	0.211	0.751	0.825	0.814	0.102	N/A	0.055	0.808
	FID ↓	**60.01**	103.87	152.39	89.08	78.95	292.28	N/A	310.5	56.08
	LPIPS ↓	**0.294**	0.446	0.508	0.405	0.386	0.657	N/A	0.702	0.389
Superres. (×4)	PSNR ↑	25.99	N/A	23.86	25.55	**26.14**	17.62	25.36	19.97	25.67
	SSIM ↑	0.862	N/A	0.803	0.865	**0.889**	0.617	0.835	0.703	0.852
	FID ↓	58.82	N/A	110.64	66.52	63.98	96.72	62.15	87.64	**52.82**
	LPIPS ↓	**0.279**	N/A	0.428	0.353	0.319	0.563	0.294	0.520	0.337

FFHQ 256 × 256 data set: image reconstruction

		PnP-SGS	SPA	TV-ADMM	PnP-ADMM	DPIR	Scor
Inpainting	PSNR ↑	**32.59**	24.09	22.03	8.41	24.41	13
	SSIM ↑	**0.913**	0.524	0.784	0.325	0.809	0.
	FID ↓	37.36	71.12	181.56	123.61	52.73	76
	LPIPS ↓	**0.144**	0.785	0.463	0.692	0.398	0.
Deblurring (Gaussian)	PSNR ↑	**27.96**	23.17	22.37	24.93	26.09	7
	SSIM ↑	**0.837**	0.499	0.801	0.812	0.820	0.
	FID ↓	**59.667**	78.67	186.74	90.42	80.18	10
	LPIPS ↓	**0.331**	0.452	0.507	0.441	0.392	0.
Deblurring (motion)	PSNR ↑	**28.46**	17.73	21.36	24.65	27.33	6
	SSIM ↑	**0.828**	0.211	0.751	0.825	0.814	0.
	FID ↓	**60.01**	103.87	152.39	89.08	78.95	29
	LPIPS ↓	**0.294**	0.446	0.508	0.405	0.386	0.
Superres. (×4)	PSNR ↑	25.99	N/A	23.86	25.55	**26.14**	17
	SSIM ↑	0.862	N/A	0.803	0.865	**0.889**	0.
	FID ↓	58.82	N/A	110.64	66.52	63.98	96
	LPIPS ↓	**0.279**	N/A	0.428	0.353	0.319	0.

Imagenet 256 × 256 data set: image reconstruction

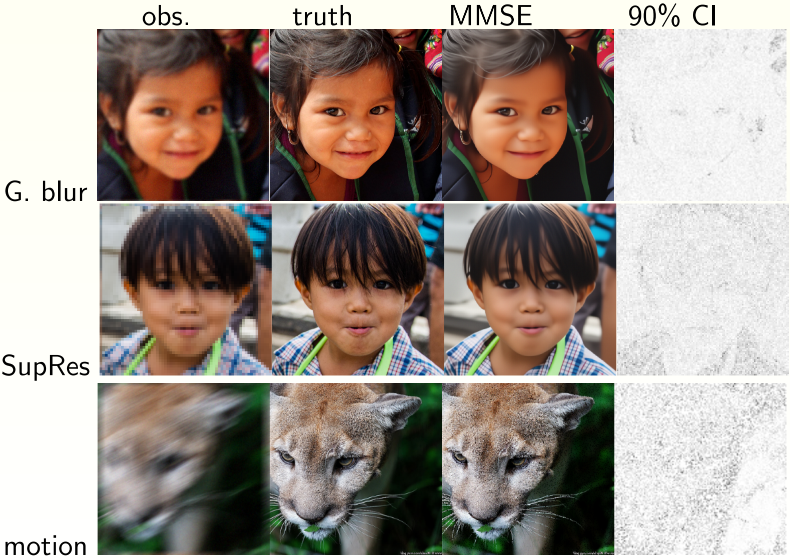
		PnP-SGS	SPA	TV-ADMM	PnP-ADMM	DPIR	Score-SDE	DDRM	MCG	DPS
Inpainting	PSNR ↑	**25.22**	<u>23.14</u>	20.96	8.39	22.08	18.62	14.29	19.03	18.90
	SSIM ↑	**0.870**	<u>0.802</u>	0.676	0.300	0.762	0.517	0.403	0.546	0.794
	FID ↓	**34.28**	41.33	189.3	114.7	37.47	127.1	114.9	39.19	<u>35.87</u>
	LPIPS ↓	**0.297**	0.323	0.510	0.677	0.448	0.659	0.665	0.414	<u>0.303</u>
Deblurring (Gaussian)	PSNR ↑	21.76	21.08	19.99	<u>21.81</u>	<u>21.81</u>	15.97	22.73	16.32	**24.25**
	SSIM ↑	0.701	0.577	0.634	0.669	0.612	0.436	<u>0.705</u>	0.441	**0.811**
	FID ↓	<u>64.12</u>	98.78	155.7	100.6	98.1	120.3	**63.02**	95.04	64.72
	LPIPS ↓	**0.399**	0.537	0.588	0.519	0.499	0.667	<u>0.427</u>	0.550	0.444
Deblurring (motion)	PSNR ↑	21.47	20.49	20.79	21.98	<u>22.49</u>	7.21	N/A	5.89	**24.92**
	SSIM ↑	0.695	0.681	0.677	0.702	<u>0.731</u>	0.120	N/A	0.037	**0.859**
	FID ↓	**47.57**	91.51	138.8	89.76	76.11	98.25	N/A	186.9	<u>56.08</u>
	LPIPS ↓	**0.372**	0.538	0.525	0.483	0.448	0.591	N/A	0.758	<u>0.389</u>
Superres. (×4)	PSNR ↑	24.33	N/A	22.17	23.75	24.30	12.25	<u>24.96</u>	13.39	**25.67**
	SSIM ↑	<u>0.772</u>	N/A	0.679	0.761	0.769	0.256	**0.790**	0.227	0.752
	FID ↓	<u>59.09</u>	N/A	130.9	97.27	88.85	170.7	59.57	144.5	**50.66**
	LPIPS ↓	<u>0.418</u>	N/A	0.523	0.433	0.424	0.701	0.339	0.637	**0.337**

Imagenet 256 × 256 data set: image reconstruction

		PnP-SGS	SPA	TV-ADMM	PnP-ADMM	DPIR	Score-S
Inpainting	PSNR ↑	**25.22**	23.14	20.96	8.39	22.08	18.62
	SSIM ↑	**0.870**	0.802	0.676	0.300	0.762	0.517
	FID ↓	**34.28**	41.33	189.3	114.7	37.47	127.1
	LPIPS ↓	**0.297**	0.323	0.510	0.677	0.448	0.659
Deblurring (Gaussian)	PSNR ↑	21.76	21.08	19.99	21.81	21.81	15.97
	SSIM ↑	0.701	0.577	0.634	0.669	0.612	0.436
	FID ↓	64.12	98.78	155.7	100.6	98.1	120.3
	LPIPS ↓	**0.399**	0.537	0.588	0.519	0.499	0.667
Deblurring (motion)	PSNR ↑	21.47	20.49	20.79	21.98	22.49	7.21
	SSIM ↑	0.695	0.681	0.677	0.702	0.731	0.120
	FID ↓	**47.57**	91.51	138.8	89.76	76.11	98.25
	LPIPS ↓	**0.372**	0.538	0.525	0.483	0.448	0.591
Superres. (×4)	PSNR ↑	24.33	N/A	22.17	23.75	24.30	12.25
	SSIM ↑	0.772	N/A	0.679	0.761	0.769	0.256
	FID ↓	59.09	N/A	130.9	97.27	88.85	170.7
	LPIPS ↓	0.418	N/A	0.523	0.433	0.424	0.701

PnP-SGS: inference with uncertainty quantification

Plug-and-Play and splitting: F. Cœurdoux's PhD, N. Dobigeon - IRIT



Efficient sampling for high dimensional problems

Nicolas Dobigeon, Florentin Cœurdoux, Maxime Vono

PnP-SGS: efficient sampling for inverse problems in high dimensions

- **SGS** & SPA split-and-augment strategy
  - Bayesian inference for complex models
  - large scale problems (big & tall)
  - **confidence intervals**
- **Efficient algorithms** for sampling: ULA, MALA, MYULA
  - **acceleration** of state-of-the-art sampling algorithms
  - **distributed** inference (privacy, distr. comput.)
- **AXDA**: **unifying** statistical framework
  - asymptotically exact: control parameter ρ
  - **non-asymptotic theoretical guarantees**
- **Capitalizing on ML**: **trained denoisers**
  - learning from **representative samples**
  - **State-of-the-art** performance

Prospects

- **Motivations** for **PnP-SGS**
  - posterior distribution $\rightarrow$ **Bayesian** + **MCMC**
  - quantify **uncertainty**
  - adapt to **any likelihood** thanks to splitting
- **Distributed sampling: fast and scalable**: **SPMD**
  - localized operators (Thouvenin et al. 2023)
  - **distributed computing**
- **Equivariance**: of course!
- **Theoretical guarantees** under mild assumptions?

Chan, S. H., Wang, X., and Elgendy, O. A. (2016), "Plug-and-play ADMM for image restoration: Fixed-point convergence and applications," *IEEE Transactions on Computational Imaging*, 3, 84–98.

Chung, H., Kim, J., Mccann, M. T., Klasky, M. L., and Ye, J. C. (2023), "Diffusion Posterior Sampling for General Noisy Inverse Problems," in *Proc. of Int. Conf. on Learning Representations*.

Coeurdoux, F., Dobigeon, N., and Chainais, P. (2023), "Learning Optimal Transport Between Two Empirical Distributions with Normalizing Flows," in *Machine Learning and Knowledge Discovery in Databases*, eds. Amini, M.-R., Canu, S., Fischer, A., Guns, T., Kralj Novak, P., and Tsoumakas, G., Cham: Springer Nature Switzerland, pp. 275–290.

Donoho, D. L. and Johnstone, J. M. (1994), "Ideal spatial adaptation by wavelet shrinkage," *Biometrika*, 81, 425–455.

Guo, X., Liu, F., and Tian, X. (2021), "Gaussian noise level estimation for color image denoising," *J. Opt. Soc. Am. A*, 38, 1150–1159.

Kawar, B., Elad, M., Ermon, S., and Song, J. (2022), "Denoising diffusion restoration models," *arXiv preprint arXiv:2201.11793*.

Li, Y., Liu, C., You, X., and Liu, J. (2022), "A Single-Image Noise Estimation Algorithm Based on Pixel-Level Low-Rank Low-Texture Patch and Principal Component Analysis," *Sensors*, 22, 8899.

Song, Y., Shen, L., Xing, L., and Ermon, S. (2022), "Solving Inverse Problems in Medical Imaging with Score-Based Generative Models," in *Proc. of Int. Conf. on Learning Representations*.

Thouvenin, P.-A., Repetti, A., and Chainais, P. (2023), "A Distributed Block-Split Gibbs Sampler with Hypergraph Structure for High-Dimensional Inverse Problems," *Journal of Computational and Graphical Statistics*, 0, 1–35.

Vono, M., Dobigeon, N., and Chainais, P. (2019a), "Split-and-augmented Gibbs sampler - Application to large-scale inference problems," *IEEE Transactions on Signal Processing*, 67, 1648–1661.

— (2019b), "Split-and-augmented Gibbs sampler—Application to large-scale inference problems," *IEEE Transactions on Signal Processing*, 67, 1648–1661.

Zhang, K., Li, Y., Zuo, W., Zhang, L., Van Gool, L., and Timofte, R. (2021), "Plug-and-play image restoration with deep denoiser prior," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44, 6360–6376.

AXDA : comparing SPA & ADMM

Connection between MAP and ADMM

By replacing Gibbs sampling steps by optimizations, ADMM appears:

$$\mathbf{x}^{(t)} \in \arg\min_{\mathbf{x}} -\log p\left(\mathbf{x}|\mathbf{z}^{(t-1)}, \mathbf{u}^{(t-1)}; \rho\right)$$

$$\mathbf{z}^{(t)} \in \arg\min_{\mathbf{z}} -\log p\left(\mathbf{z}|\mathbf{x}^{(t)}, \mathbf{u}^{(t-1)}; \rho\right)$$

$$\mathbf{u}^{(t)} = \mathbf{u}^{(t-1)} + \mathbf{x}^{(t)} - \mathbf{z}^{(t)}$$