ExCALIBUR 10

Towards Learned Exascale Computational Imaging for the SKA

Jason McEwen
Mullard Space Science Laboratory (MSSL)
University College London (UCL)

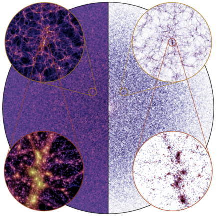
Towards Exascale-Ready Astrophysics (TERA) 2024

SciAI UCL · Met Office · UKRI UK Research and Innovation · UK Atomic Energy Authority

Exascale computational challenges

Big-Data

Big-AI



Big-Sims

$\Rightarrow$ All require **Big-Compute**.

Overview

1. SKA Exascale

2. Imaging Strategy

3. Exascale Algorithms

    Blocking for Distribution

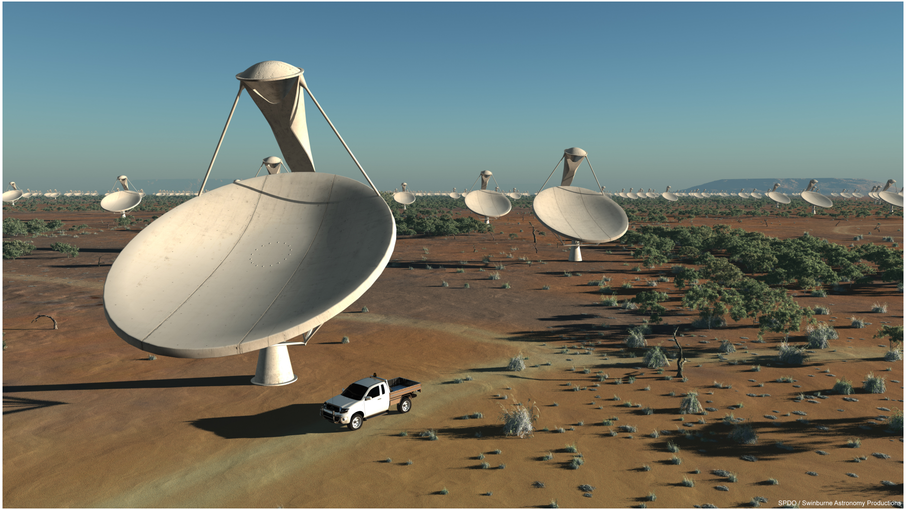
    Uncertainty Quantification

    AI Data-Driven Prior

4. Demonstrations

SKA Exascale

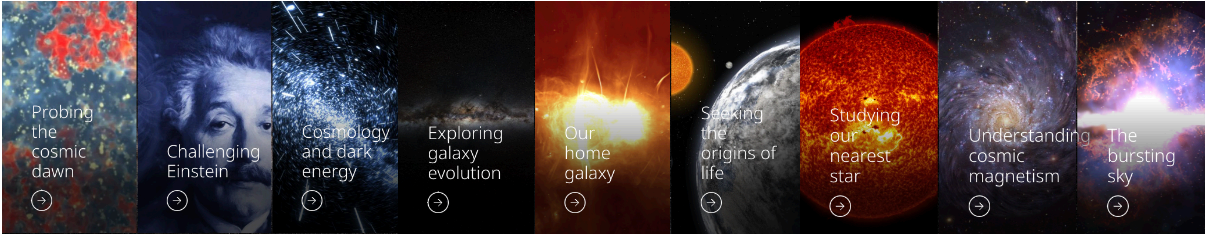
Square Kilometre Array (SKA): next-gen radio interferometric telescope



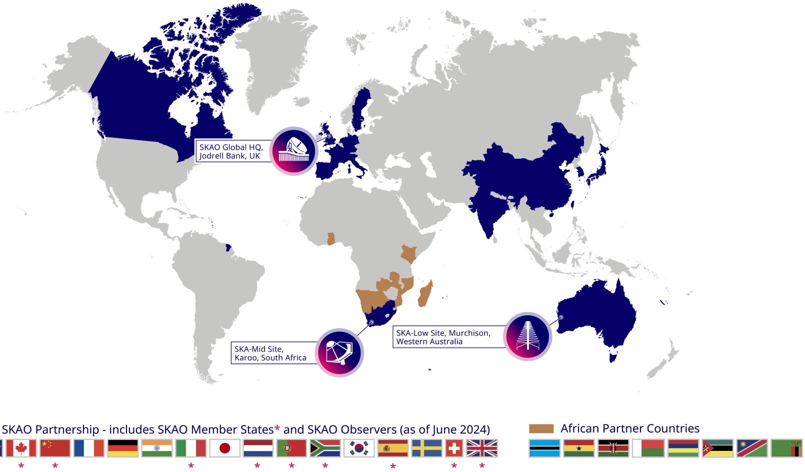
SKA science goals

Orders of magnitude improvement in sensitivity and resolution.

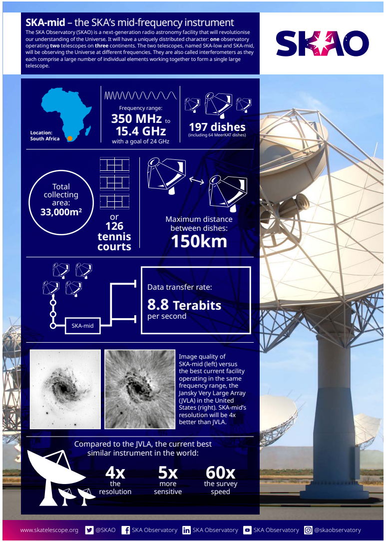
Unlock broad range of science goals.



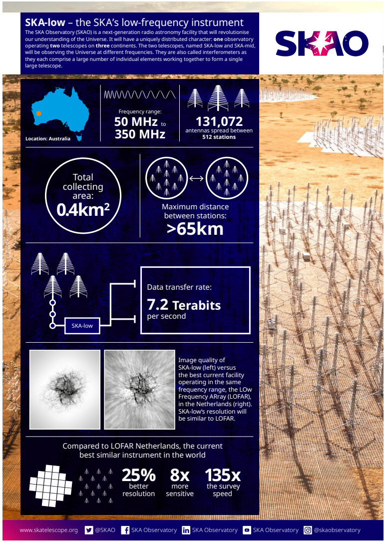
SKA partners



SKA sites

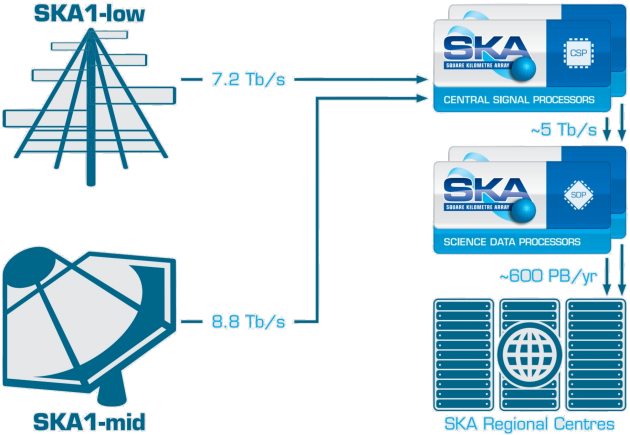


SKA-mid – the SKA's mid-frequency instrument
The SKA Observatory (SKAO) is a next-generation radio astronomy facility that will revolutionise our understanding of the Universe. It will have a uniquely distributed character: one observatory operating two telescopes on three continents. The two telescopes, named SKA-low and SKA-mid, will be observing the Universe at different frequencies. They are also called interferometers as they each comprise a large number of individual elements working together to form a single large telescope.
SKAO
Location: South Africa
Frequency range: 350 MHz to 15.4 GHz with a goal of 24 GHz
197 dishes (including 64 MeerKAT dishes)
Total collecting area: 33,000m²
or 126 tennis courts
Maximum distance between dishes: 150km
SKA-mid
Data transfer rate: 8.8 Terabits per second
Image quality of SKA-mid (left) versus the best current facility operating in the same frequency range, the Jansky Very Large Array (JVLA) in the United States (right). SKA-mid's resolution will be 4x better than JVLA.
Compared to the JVLA, the current best similar instrument in the world:
4x the resolution
5x more sensitive
60x the survey speed
www.skatelescope.org @SKAO SKA Observatory SKA Observatory SKA Observatory @skaobservatory



SKA-low – the SKA's low-frequency instrument
The SKA Observatory (SKAO) is a next-generation radio astronomy facility that will revolutionise our understanding of the Universe. It will have a uniquely distributed character: one observatory operating two telescopes on three continents. The two telescopes, named SKA-low and SKA-mid, will be observing the Universe at different frequencies. They are also called interferometers as they each comprise a large number of individual elements working together to form a single large telescope.
SKAO
Location: Australia
Frequency range: 50 MHz to 350 MHz
131,072 antennas spread between 512 stations
Total collecting area: 0.4km²
Maximum distance between stations: >65km
SKA-low
Data transfer rate: 7.2 Terabits per second
Image quality of SKA-low (left) versus the best current facility operating in the same frequency range, the LOw Frequency ARray (LOFAR), in the Netherlands (right). SKA-low's resolution will be similar to LOFAR.
Compared to LOFAR Netherlands, the current best similar instrument in the world
25% better resolution
8x more sensitive
135x the survey speed
www.skatelescope.org @SKAO SKA Observatory SKA Observatory SKA Observatory @skaobservatory

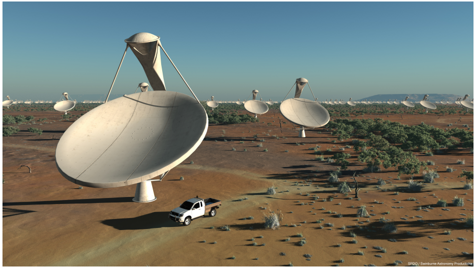
SKA data rates



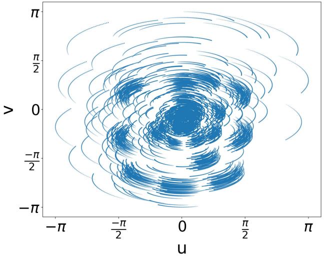
⇒ **8.5 Exabytes** over the 15-year lifetime of initial high-priority science programmes (Scaife 2020).

Imaging Strategy

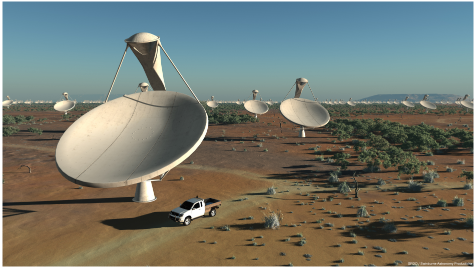
Radio interferometric telescopes acquire "Fourier" measurements



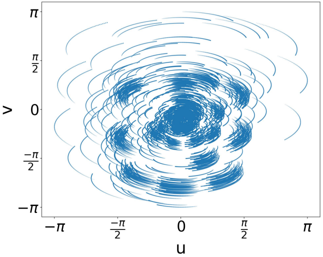
"Fourier"
Measurements
$\Rightarrow$



Radio interferometric telescopes acquire “Fourier” measurements



“Fourier”
Measurements
$\Rightarrow$



Interferometric imaging is an **exascale computational inverse imaging problem**.

Radio interferometric inverse problem

Radio interferometric imaging **ill-posed inverse problem**:

$$y = \boldsymbol{\Phi}(x) + n$$

for data (visibilities) y, telescope model $\boldsymbol{\Phi}$, underlying image x and noise n.

Radio interferometric inverse problem

Radio interferometric imaging **ill-posed inverse problem**:

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

$$y \xleftarrow{\text{forward model}} x$$

for data (visibilities) y, telescope model $\mathbf{\Phi}$, underlying image x and noise n.

Radio interferometric inverse problem

Radio interferometric imaging **ill-posed inverse problem**:

$$y = \boldsymbol{\Phi}(x) + n$$

$$y \xleftarrow{\text{forward model}} x$$

$$y \xrightarrow[\text{inverse inference}]{} x$$

for data (visibilities) y, telescope model $\boldsymbol{\Phi}$, underlying image x and noise n.

Radio interferometric inverse problem

Radio interferometric imaging **ill-posed inverse problem**:

$$y = \boldsymbol{\Phi}(x) + n$$

$$y \xleftarrow{\text{forward model}} x$$

$$y \xrightarrow[\text{inverse inference}]{} x$$

for data (visibilities) y, telescope model $\boldsymbol{\Phi}$, underlying image x and noise n.

Highly realistic wide-field telescope model

(Pratley, Johnston-Hollitt & McEwen 2019; Pratley, Johnston-Hollitt & McEwen 2020).

Radio interferometric inverse problem

Radio interferometric imaging **ill-posed inverse problem**:

$$y = \boldsymbol{\Phi}(x) + n$$

$$y \xleftarrow{\text{forward model}} x$$

$$y \xrightarrow[\text{inverse inference}]{} x$$

for data (visibilities) y, telescope model $\boldsymbol{\Phi}$, underlying image x and noise n.

Highly realistic wide-field telescope model

(Pratley, Johnston-Hollitt & McEwen 2019; Pratley, Johnston-Hollitt & McEwen 2020).

Big-Data $\Rightarrow$ Big-Compute

since compute scales as $\mathcal{O}(M)$ for M data measurements.

Statistical framework

Inverse problem is ill-posed so inject regularising prior information.

Statistical framework

Inverse problem is ill-posed so **inject regularising prior information**.

Bayes Theorem:

$$p(\mathbf{x} \mid \mathbf{y}) \propto p(\mathbf{y} \mid \mathbf{x}) p(\mathbf{x}) \, , \quad \textit{i.e.} \text{ posterior} \propto \text{likelihood} \times \text{prior}$$

Define likelihood (assuming Gaussian noise) and prior:

$$p(\mathbf{y} \mid \mathbf{x}) = \mathcal{L}(\mathbf{x}) \propto \exp\left(-\|\mathbf{y} - \mathbf{\Phi}\mathbf{x}\|_2^2 / (2\sigma^2)\right) \qquad p(\mathbf{x}) = \pi(\mathbf{x}) \propto \exp\left(-R(\mathbf{x})\right)$$

likelihood prior

Optimisation vs sampling

MAP estimation

- \+ Based on optimisation so computationally efficient.
- – **No uncertainties** (traditionally).
- – **Hand-crafted priors** (traditionally).

MCMC sampling

- – Based on sampling so **computationally demanding**.
- \+ Uncertatinties encoded in posterior.
- – **Hand-crafted priors** (traditionally).

Computational imaging strategy

Goals:

+ Computationally efficient (optimisation + distribution).
+ Quantifies uncertainties.
+ Data-driven AI priors (enhance reconstruction fidelity).

Computational imaging strategy

Goals:

+ Computationally efficient (optimisation + distribution).
+ Quantifies uncertainties.
+ Data-driven AI priors (enhance reconstruction fidelity).

Achieve by combining:

1. Statistical framework: Bayesian inference and MAP estimation.
2. Mathematical theory: probability concentration theorem for log-convex distributions.
3. Constrained AI model: convex AI model with explicit potential.

Solve optimisation problem

Solve optimisation problem (MAP estimation by variation regularisation):

$$x_{\text{map}} = \arg\max_{x} \Big[\log p(y \mid x) \Big] = \arg\min_{x} \Big[\| y - \Phi x \|_2^2 + \lambda R(x) \Big]$$

regulariser

Solve optimisation problem

Solve optimisation problem (MAP estimation by variation regularisation):

$$x_{\text{map}} = \arg\max_x \left[\log p(y \mid x) \right] = \arg\min_x \left[\|y - \Phi x\|_2^2 + \lambda R(x) \right]$$

regulariser

Traditionally, **hand-crafted regularisers** used
(*e.g.* $R(x) = \|\Psi^\dagger x\|_1$ to promote sparsity in some (wavelet) dictionary Ψ).

Instead, adopt data-driven AI prior for regulariser (**small-AI**) trained on simulations (**big-sims**).

Solve optimisation problem

Solve optimisation problem (MAP estimation by variation regularisation):

$$x_{\text{map}} = \arg\max_x \left[\log p(y \mid x) \right] = \arg\min_x \left[\, \| y - \boldsymbol{\Phi} x \|_2^2 + \underbrace{\lambda \, R(x)}_{\text{regulariser}} \, \right]$$

Traditionally, **hand-crafted regularisers** used
(*e.g.* $R(x) = \|\boldsymbol{\Psi}^\dagger x\|_1$ to promote sparsity in some (wavelet) dictionary $\boldsymbol{\Psi}$).

Instead, adopt data-driven AI prior for regulariser (**small-AI**) trained on simulations (**big-sims**).

$\Rightarrow$ **Highly distributed and parallelised** optimisation algorithms, with **low communication** overhead.

Exascale Algorithms

Exascale Algorithms

Blocking for Distribution

Block distribution

Solve resulting convex optimisation problem by **proximal splitting**.

Block algorithm to **distribute data and compute** (telescope model):

(Carrillo, McEwen & Wiaux 2014; Onose *et al.* (inc. McEwen) 2016; Pratley, Johnston-Hollitt & McEwen 2019; Pratley, McEwen *et al.* 2019; Pratley, Johnston-Hollitt & McEwen 2020)

$$\boldsymbol{y} = \begin{bmatrix} \boldsymbol{y}_1 \\ \vdots \\ \boldsymbol{y}_{n_\mathrm{d}} \end{bmatrix}, \quad \boldsymbol{\Phi} = \begin{bmatrix} \boldsymbol{\Phi}_1 \\ \vdots \\ \boldsymbol{\Phi}_{n_\mathrm{d}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{G}_1 \boldsymbol{M}_1 \\ \vdots \\ \boldsymbol{G}_{n_\mathrm{d}} \boldsymbol{M}_{n_\mathrm{d}} \end{bmatrix} \boldsymbol{F} \boldsymbol{Z} .$$

Block distribution

Solve resulting convex optimisation problem by **proximal splitting**.

Block algorithm to **distribute data and compute** (telescope model):

(Carrillo, McEwen & Wiaux 2014; Onose *et al.* (inc. McEwen) 2016; Pratley, Johnston-Hollitt & McEwen 2019; Pratley, McEwen *et al.* 2019; Pratley, Johnston-Hollitt & McEwen 2020)

$$\boldsymbol{y} = \begin{bmatrix} \boldsymbol{y}_1 \\ \vdots \\ \boldsymbol{y}_{n_\mathrm{d}} \end{bmatrix}, \quad \boldsymbol{\Phi} = \begin{bmatrix} \boldsymbol{\Phi}_1 \\ \vdots \\ \boldsymbol{\Phi}_{n_\mathrm{d}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{G}_1 \boldsymbol{M}_1 \\ \vdots \\ \boldsymbol{G}_{n_\mathrm{d}} \boldsymbol{M}_{n_\mathrm{d}} \end{bmatrix} \boldsymbol{F} \boldsymbol{Z} .$$

- Stochastic updates to support big-data.
- Two internal distribution strategies:
  1. Distribute image (*i.e.* distribute $\boldsymbol{\Phi}_i$)
  2. Distribute Fourier grid (*i.e.* distribute $\boldsymbol{G}_i \boldsymbol{M}_i$)

Block distribution

Solve resulting convex optimisation problem by **proximal splitting**.

Block algorithm to **distribute data and compute** (telescope model):

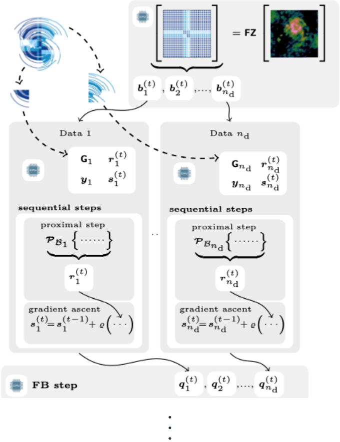
(Carrillo, McEwen & Wiaux 2014; Onose *et al.* (inc. McEwen) 2016; Pratley, Johnston-Hollitt & McEwen 2019; Pratley, McEwen *et al.* 2019; Pratley, Johnston-Hollitt & McEwen 2020)

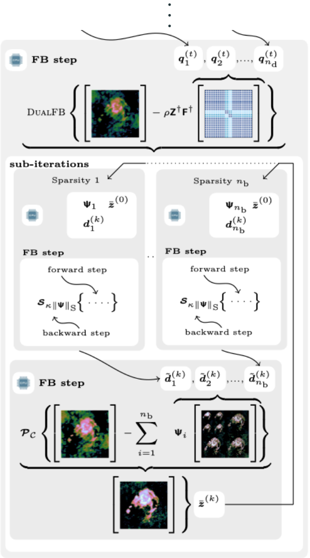
$$\boldsymbol{y} = \begin{bmatrix} \boldsymbol{y}_1 \\ \vdots \\ \boldsymbol{y}_{n_\mathrm{d}} \end{bmatrix}, \quad \boldsymbol{\Phi} = \begin{bmatrix} \boldsymbol{\Phi}_1 \\ \vdots \\ \boldsymbol{\Phi}_{n_\mathrm{d}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{G}_1 \boldsymbol{M}_1 \\ \vdots \\ \boldsymbol{G}_{n_\mathrm{d}} \boldsymbol{M}_{n_\mathrm{d}} \end{bmatrix} \boldsymbol{F}\boldsymbol{Z}.$$

- Stochastic updates to support big-data.
- Two internal distribution strategies:
    1. Distribute image (*i.e.* distribute $\boldsymbol{\Phi}_i$)
    2. Distribute Fourier grid (*i.e.* distribute $\boldsymbol{G}_i \boldsymbol{M}_i$)

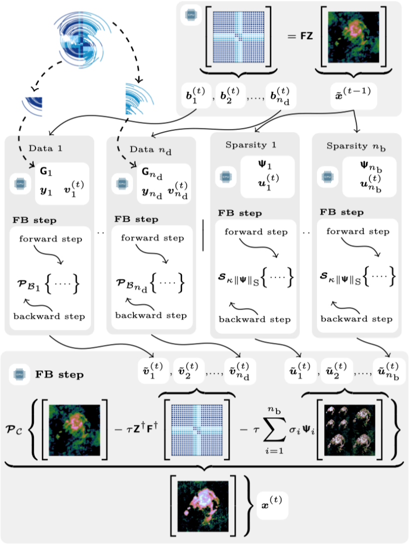
Benchmarking performed in Pratley, McEwen *et al.* 2019 (although out of date).

Block distributed alternating direction method of multipliers (ADMM) algorithm

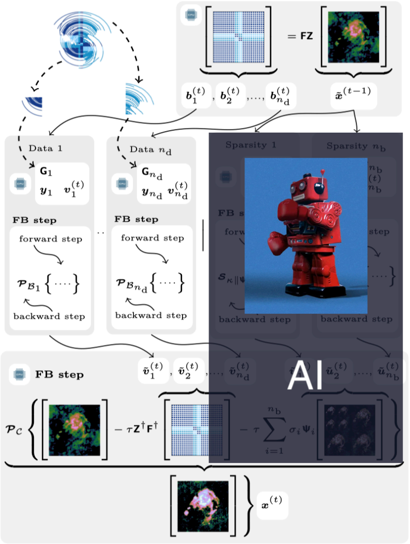




Block distributed primal dual algorithm



Block distributed primal dual algorithm with AI prior



Exascale Algorithms

Uncertainty Quantification

Convex probability concentration for uncertainty quantification

Posterior credible region:

$$p(x \in C_\alpha | y) = \int_{x \in \mathbb{R}^N} p(x|y) \mathbb{1}_{C_\alpha} \mathrm{d}x = 1 - \alpha.$$

Consider the **highest posterior density (HPD) region**

$$C^*_\alpha = \left\{ x : -\log p(x) \leq \gamma_\alpha \right\}, \quad \text{with } \gamma_\alpha \in \mathbb{R}, \quad \text{and } p(x \in C^*_\alpha | y) = 1 - \alpha \text{ holds.}$$

Convex probability concentration for uncertainty quantification

Posterior credible region:

$$p(x \in C_\alpha | y) = \int_{x \in \mathbb{R}^N} p(x|y) \mathbb{1}_{C_\alpha} \mathrm{d}x = 1 - \alpha.$$

Consider the **highest posterior density (HPD) region**

$$C^*_\alpha = \{x : -\log p(x) \leq \gamma_\alpha\}, \quad \text{with } \gamma_\alpha \in \mathbb{R}, \quad \text{and } p(x \in C^*_\alpha | y) = 1 - \alpha \text{ holds.}$$

Theorem 3.1 (Pereyra 2017)

Suppose the posterior $\log p(x|y) \propto \log \mathcal{L}(x) + \log \pi(x)$ is log-concave on $\mathbb{R}^N$. Then, for any $\alpha \in (4\mathrm{e}^[(-N/3)], 1)$, the HPD region C^*_α is contained by

$$\hat{C}_\alpha = \left\{x : \log \mathcal{L}(x) + \log \pi(x) \leq \hat{\gamma}_\alpha = \log \mathcal{L}(\hat{x}_{\text{MAP}}) + \log \pi(\hat{x}_{\text{MAP}}) + \sqrt{N}\tau_\alpha + N\right\},$$

with a positive constant $\tau_\alpha = \sqrt{16 \log(3/\alpha)}$ independent of $p(x|y)$.

Convex probability concentration for uncertainty quantification

Posterior credible region:

$$p(x \in C_\alpha | y) = \int_{x \in \mathbb{R}^N} p(x|y) \mathbb{1}_{C_\alpha} \mathrm{d}x = 1 - \alpha.$$

Consider the **highest posterior density (HPD) region**

$$C^*_\alpha = \{x : -\log p(x) \leq \gamma_\alpha\}, \quad \text{with } \gamma_\alpha \in \mathbb{R}, \quad \text{and } p(x \in C^*_\alpha | y) = 1 - \alpha \text{ holds.}$$

Theorem 3.1 (Pereyra 2017)

Suppose the posterior $\log p(x|y) \propto \log \mathcal{L}(x) + \log \pi(x)$ is log-concave on $\mathbb{R}^N$. Then, for any $\alpha \in (4\mathrm{e}^[(-N/3)], 1)$, the HPD region C^*_α is contained by

$$\hat{C}_\alpha = \left\{x : \log \mathcal{L}(x) + \log \pi(x) \leq \hat{\gamma}_\alpha = \log \mathcal{L}(\hat{x}_{\text{MAP}}) + \log \pi(\hat{x}_{\text{MAP}}) + \sqrt{N}\tau_\alpha + N\right\},$$

with a positive constant $\tau_\alpha = \sqrt{16 \log(3/\alpha)}$ independent of $p(x|y)$.

Need **only evaluate** $\log \mathcal{L} + \log \pi$ **for the MAP estimate** x_{MAP}!

Local Bayesian credible intervals for sparse reconstruction

(Cai, Pereyra & McEwen 2018b)

Let Ω define the area (or pixel) over which to compute the credible interval $(\tilde{\xi}_-, \tilde{\xi}_+)$ and $\boldsymbol{\zeta}$ be an index vector describing Ω (*i.e.* $\zeta_i = 1$ if $i \in \Omega$ and 0 otherwise).

Consider the test image with the Ω region replaced by constant value ξ:

$$x' = x^\star(\mathcal{I} - \zeta) + \xi\zeta \; .$$

Given $\tilde{\gamma}_\alpha$ and $x^\star$, compute the credible interval by

$$\tilde{\xi}_- = \min_\xi \left\{ \xi \mid \log \mathcal{L}(x') + \log \pi(x') \leq \tilde{\gamma}_\alpha, \; \forall \xi \in [-\infty, +\infty) \right\} ,$$

$$\tilde{\xi}_+ = \max_\xi \left\{ \xi \mid \log \mathcal{L}(x') + \log \pi(x') \leq \tilde{\gamma}_\alpha, \; \forall \xi \in [-\infty, +\infty) \right\} .$$

Hypothesis testing

Hypothesis testing of physical structure

(Pereyra 2017; Cai, Pereyra & McEwen 2018a)

1. Remove structure of interest from recovered image $\boldsymbol{x}^\star$.
2. Inpaint background (noise) into region, yielding surrogate image $\boldsymbol{x}'$.
3. Test whether $\boldsymbol{x}' \in C_\alpha$:
   - If $\boldsymbol{x}' \notin C_\alpha$ then reject hypothesis that structure is an artifact with confidence $(1-\alpha)$%, *i.e.* **structure most likely physical**.
   - If $\boldsymbol{x}' \in C_\alpha$ uncertainly too high to draw strong conclusions about the physical nature of the structure.

Exascale Algorithms

AI Data-Driven Prior

Convex AI prior

Adopt **neural-network-based convex regulariser** R
(Goujon *et al.* 2022; Liaudat *et al.* McEwen 2024):

$$R(\boldsymbol{x}) = \sum_{n=1}^{N_C} \sum_{k} \psi_n \left((\boldsymbol{h}_n * \boldsymbol{x}) [k] \right) ,$$

- ψ_n are learned convex profile functions with Lipschitz continuous derivative;
- N_C learned convolutional filters $\boldsymbol{h}_n$.

Adopt **neural-network-based convex regulariser** R
(Goujon *et al.* 2022; Liaudat *et al.* McEwen 2024):

$$R(x) = \sum_{n=1}^{N_C} \sum_{k} \psi_n \left((\boldsymbol{h}_n * x) [k] \right),$$

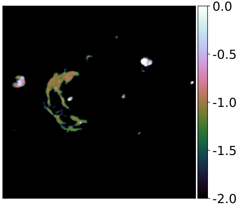
- ψ_n are learned convex profile functions with Lipschitz continuous derivative;
- N_C learned convolutional filters $\boldsymbol{h}_n$.

Properties:

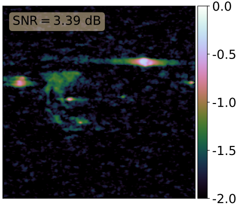
1. Convex + explicit $\Rightarrow$ leverage convex UQ theory.
2. Smooth regulariser with known Lipschitz constant $\Rightarrow$ theoretical convergence guarantees.

Demonstrations

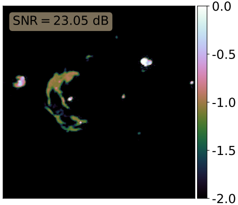
Reconstructed images



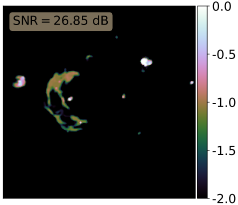
Ground truth



Dirty image
SNR=3.39 dB

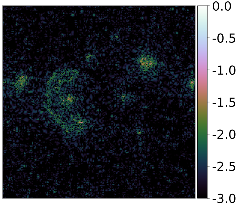


Reconstruction (classical)
SNR=23.05 dB

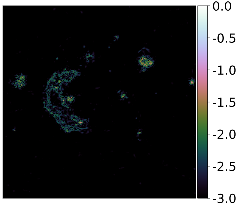


Reconstruction (learned)
SNR= 26.85 dB

(Liaudat *et al.* McEwen 2024)

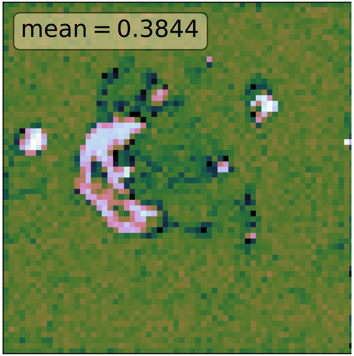


Error (classical)

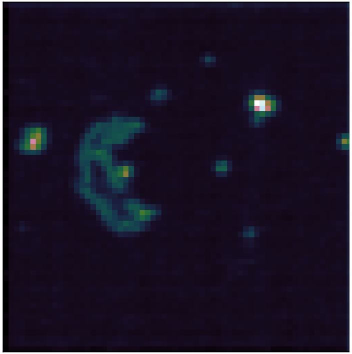


Error (learned)

Approximate local Bayesian credible intervals



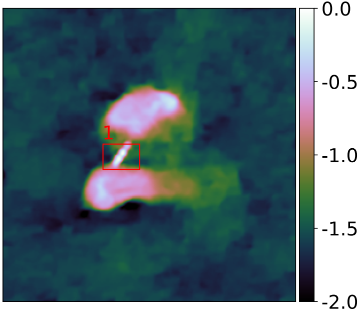
LCI
(super-pixel size 4×4)



MCMC standard deviation
(super-pixel size 4×4)

(Liaudat *et al.* McEwen 2024)

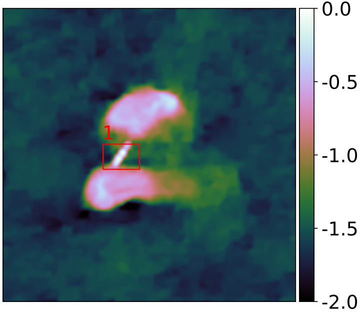
Hypothesis testing of structure



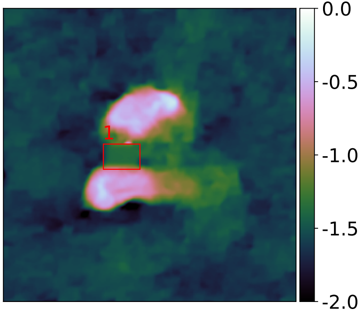
Reconstructed image

(Liaudat *et al.* McEwen 2024)

Hypothesis testing of structure

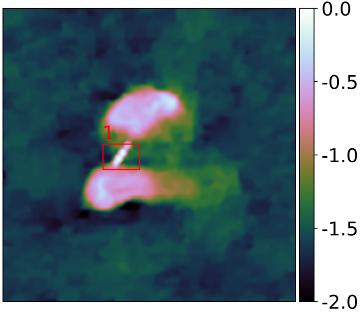


Reconstructed image



Surrogate test image (region removed)

(Liaudat *et al.* McEwen 2024)



Reconstructed image

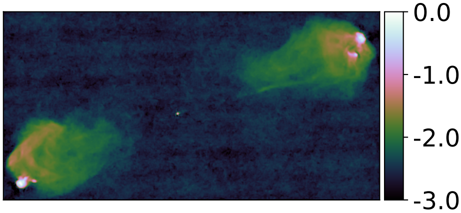
Surrogate test image (region removed)

Reject null hypothesis

$\Rightarrow$ **structure physical**

(Liaudat *et al.* McEwen 2024)

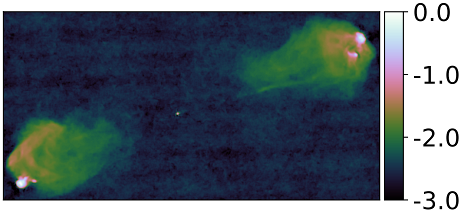
Hypothesis testing of substructure



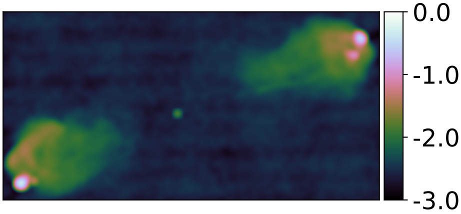
Reconstructed image

(Liaudat *et al.* McEwen 2024)

Hypothesis testing of substructure



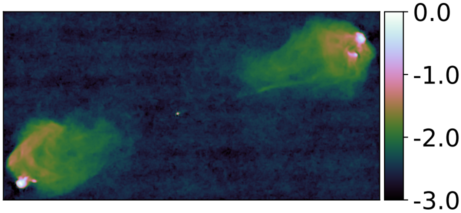
Reconstructed image



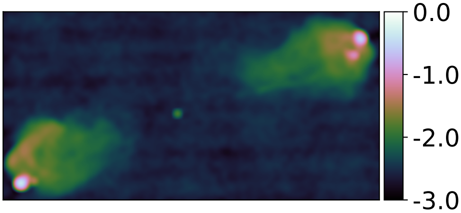
Surrogate test image (blurred)

(Liaudat *et al.* McEwen 2024)

Hypothesis testing of substructure



Reconstructed image

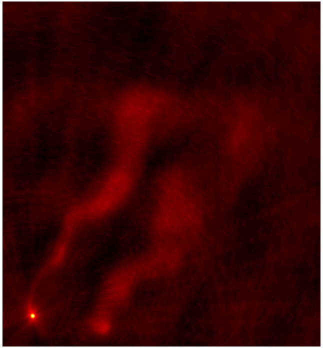


Surrogate test image (blurred)

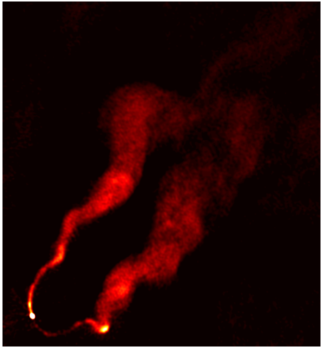
Reject null hypothesis ⇒ **substructure physical**

(Liaudat *et al.* McEwen 2024)

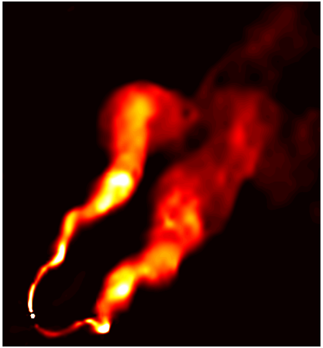
Imaging 3C128 with VLA



Dirty image

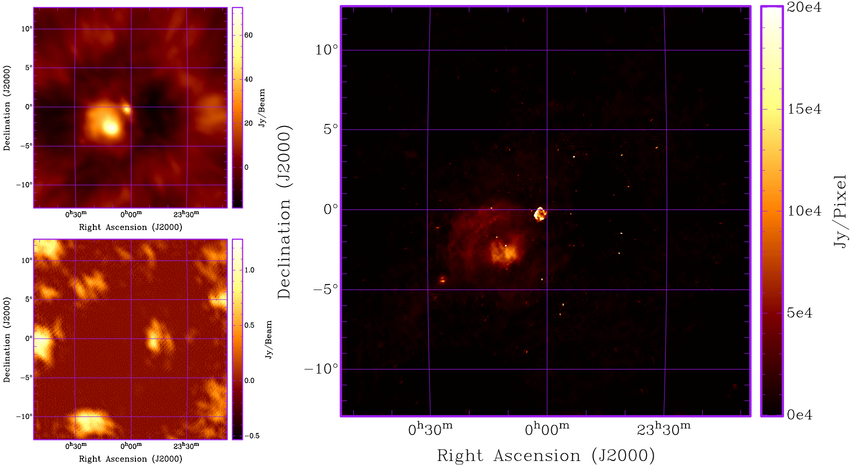


CLEAN



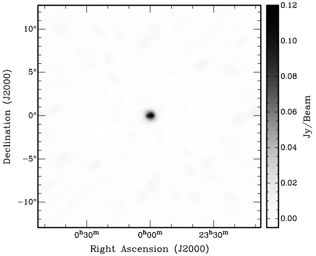
PURIFY (Ours)

(Pratley, McEwen *et al.* 2018)

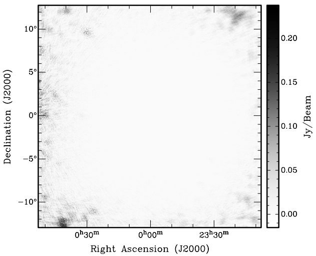


(Pratley, Johnston-Hollitt & McEwen 2019)

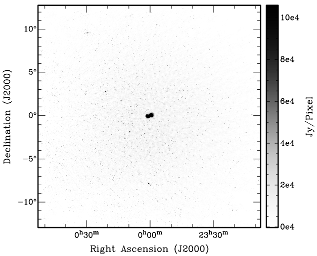
Imaging Fornax A with MWA



Dirty image



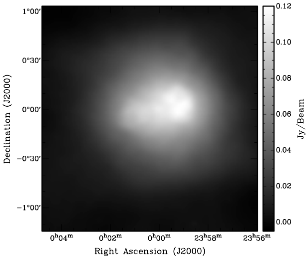
Residuals



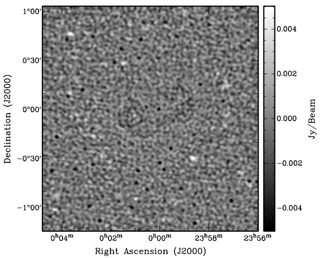
Reconstruction

(Pratley, Johnston-Hollitt & McEwen 2020)

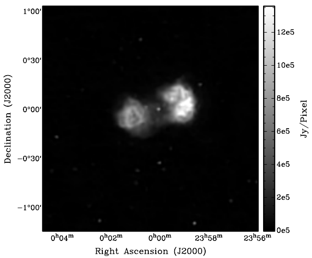
Imaging Fornax A with MWA



Dirty image



Residuals



Reconstruction

(Pratley, Johnston-Hollitt & McEwen 2020)

Open-source codes

PURIFY code

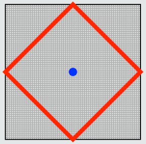
https://github.com/astro-informatics/purify

Next-generation radio interferometric imaging

PURIFY is a highly distributed and parallelized open-source C++ code for radio interferometric imaging, leveraging recent developments in the field of variational regularization, convex optimisation, and learned imaging.

SOPT code

https://github.com/astro-informatics/sopt



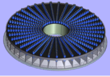
Sparse OPTimisation

SOPT is a highly distributed and parallelized open-source C++ code for variational regularization and convex optimisation, with learned data-driven priors.

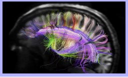
Learned Exascale Computational Imaging (LEXCI)

Radio Interferometry
e.g. MWA, SKA

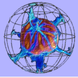
Optical Interferometry
e.g. SPIDER



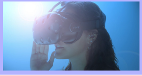
Medical Imaging
e.g. Diffusion MRI



Seismic Imaging
e.g. Mantle Plumes



Computer Graphics
e.g. Inverse Rendering



LEXCI project

Astronomical Imaging | **Planetary & Climate Science** | **Molecular Biology** | **Optical Photography** | **Geophysics**

Acoustics | **Computer Vision** | **Many Others...**

Extended application areas

Summary

- SKA is an exascale experiment.
- Learned exascale computational inverse imaging (LEXCI) framework
  1. **Highly distributed and parallelised**
  2. Highly **realistic telescope modelling** (exact wide-field corrections)
  3. **Superior reconstruction quality** by using learned AI data-driven priors
  4. **Uncertainty quantification for exascale imaging** with learned priors for the first time.
  5. **Validated** by MCMC sampling (for low-dimensional setting)
- Next steps
  1. **Integrating AI priors and uncertainty quantification** into PURIFY and SOPT
  2. **Benchmark** computational performance
  3. Apply full framework to **real observations**