A Stochastic LBFGS Algorithm for Radio Interferometric Calibration

Sarod Yatawatta

ASTRON

The Netherlands Institute for Radio Astronomy,

Dwingeloo, The Netherlands

collaborators

Lukas De Clercq, Hanno Spreeuw and Faruk Diblen

Netherlands eScience Center,

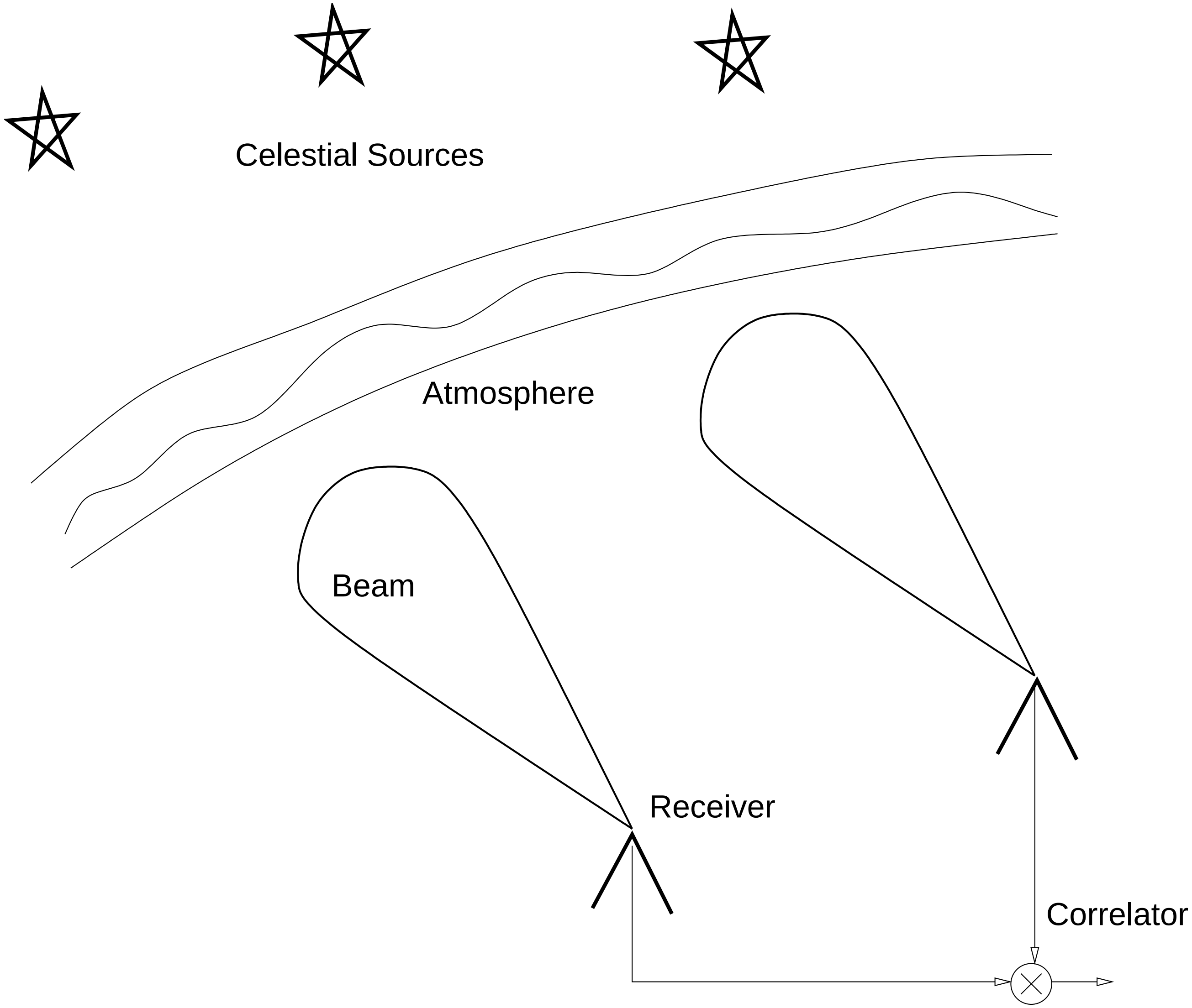
Science Park, Amsterdam, The Netherlands.

This work was supported by

Netherlands eScience Center

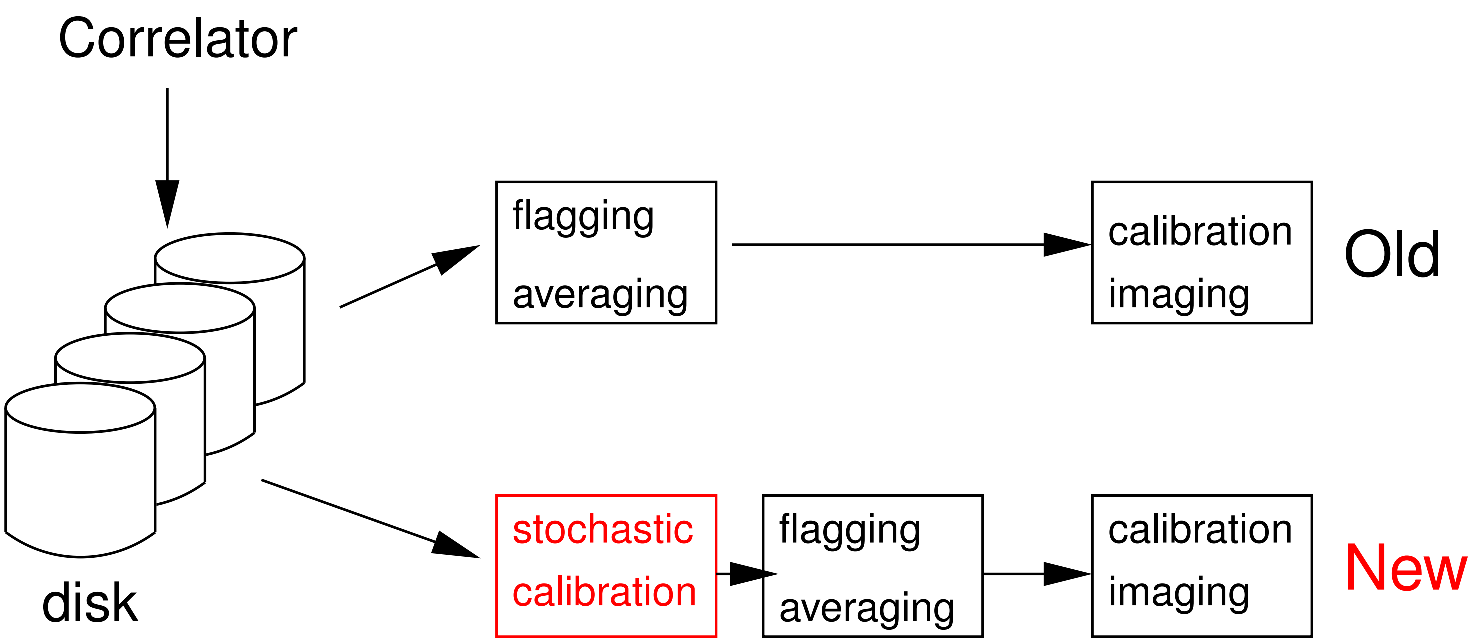
(project DIRAC, grant 27016G05).

Radio interferometers



Two receivers after correlation = interferometer

Data processing



- Removal of interference: flagging, Size reduction: averaging.
- Estimate and correct data for systematic errors: calibration.
- Image formation from Fourier space data: imaging.
- Stochastic calibration: calibration of very large volumes of data, that cannot fit into computer memory.

Modern radio telescopes

LOFAR core in The Netherlands

Modern radio telescopes

Closeup view of LOFAR

Calibration

Observed data at interferometer p-q

$$\mathbf{V}_{pq} = \sum_{k=1}^{K} \mathbf{J}_{pk}\mathbf{C}_{pqk}\mathbf{J}_{qk}^{H} + \mathbf{N}_{pq}$$

$\mathbf{V}_{pq}, \mathbf{J}_{pk}, \mathbf{C}_{pqk}, \mathbf{J}_{qk} \in \mathbb{C}^{2\times 2}$.

Calibration along K directions in the sky.

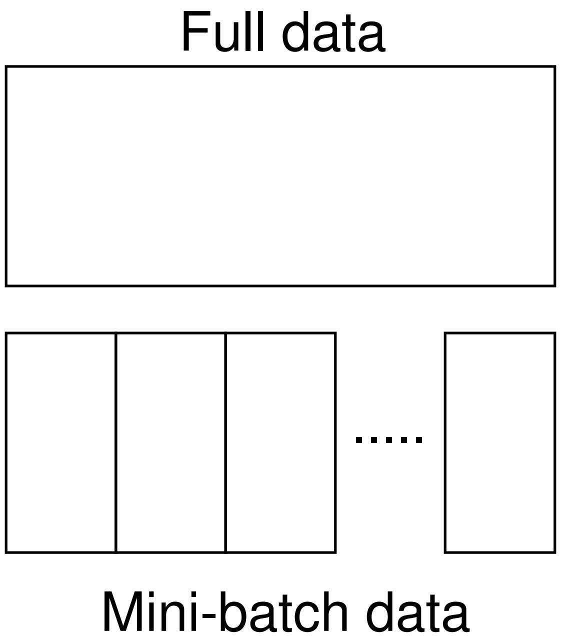
Systematic errors: $\mathbf{J}_{pk}, \mathbf{J}_{qk}$ Model: $\mathbf{C}_{pqk}$ Noise: $\mathbf{N}_{pq}$

For N stations, there are $N(N-1)/2$ pairs of p-q. Data taken at many thousands of frequencies and every few seconds. Total data that are used to estimate $\mathbf{J}_{pk}, \mathbf{J}_{qk}$ can easily run into several hundred GBs.

Minimize a robust cost function

$$f(\boldsymbol{\theta}) = \sum_{i=1}^{4TFN(N-1)} \log\left(1 + \frac{(\mathbf{x}[i] - \mathbf{m}(\boldsymbol{\theta})[i])^2}{\nu}\right)$$

Stochastic mode



$f(\boldsymbol{\theta}) \qquad \nabla f(\boldsymbol{\theta})$

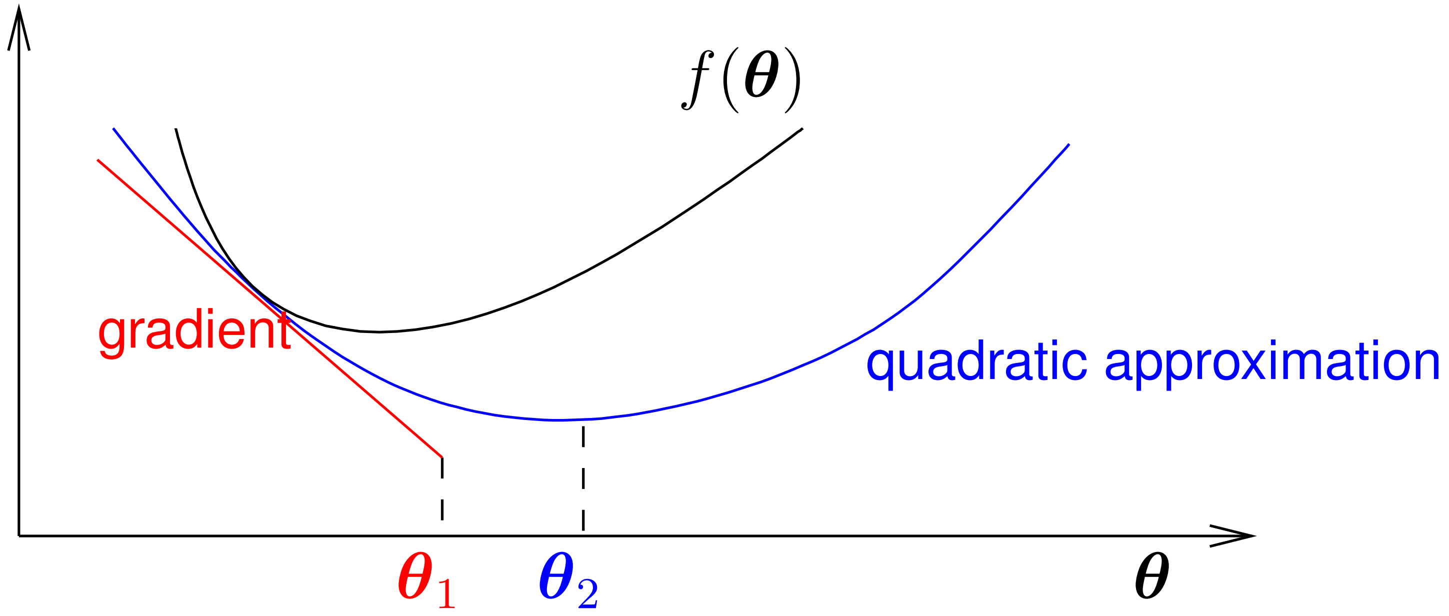
$f_1(\boldsymbol{\theta}), f_2(\boldsymbol{\theta}), ... \quad \nabla f_1(\boldsymbol{\theta}), \nabla f_2(\boldsymbol{\theta}); \cdots$

- Cost and gradient are different for each mini-batch.
- Gradient of each mini-batch is noisy.

$$\nabla f_i(\boldsymbol{\theta}) \approx \nabla f(\boldsymbol{\theta})$$

- Parameters $\boldsymbol{\theta}$ are valid for the full dataset.
- For a long observation, same setup repeats.

Nonlinear optimization



Common minimization strategy:

- ☐ Find a descent direction
- ☐ Find a descent amount (step size)
- ☐ Update parameters and repeat above

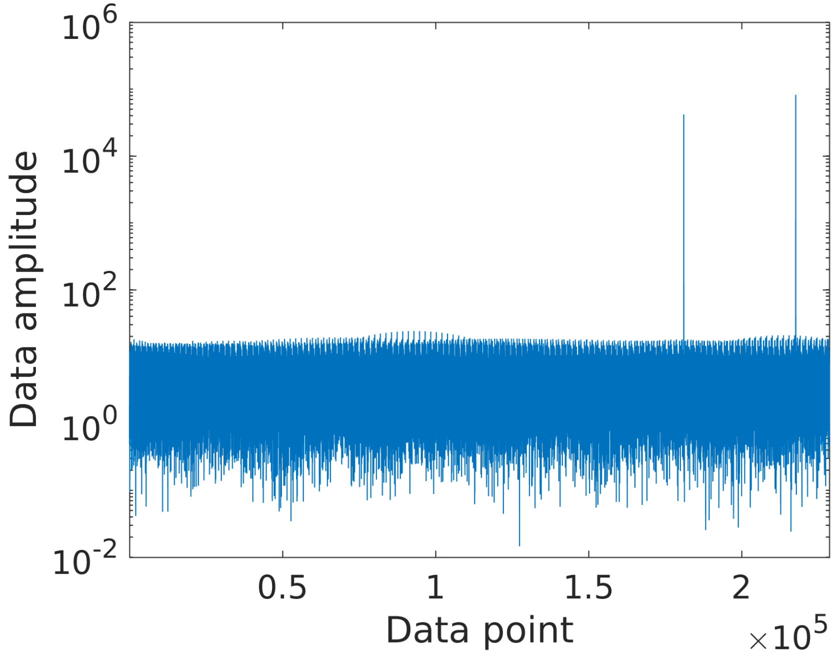
Trust region methods: no step size needed, just the trust region radius.

New stochastic LBFGS

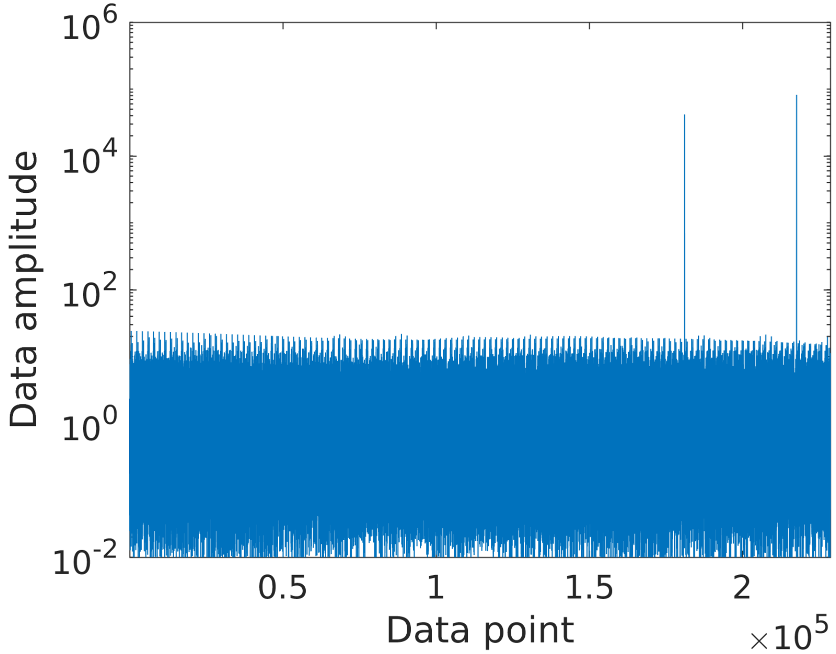
- LBFGS: Limited memory Broyden Fletcher Goldfarb Shanno algorithm, a very popular quasi-Newton method.
- No full Hessian is stored, only a subset of curvature pairs, i.e., difference of gradient $\nabla f(\boldsymbol{\theta}_1) - \nabla f(\boldsymbol{\theta}_2)$ and difference of parameters $\boldsymbol{\theta}_1 - \boldsymbol{\theta}_2$ are stored. Memory efficient, suitable for large number of parameters.
- New method: Uses online estimate of variance of $\|\nabla f(\boldsymbol{\theta})\|$ using [Welford 1962] to adjust line search. The mini-batch size is fixed.
- Robust cost function in calibration: Student's T noise model.

Simulation

Calibration along $K = 4$ directions of 10 minutes of LOFAR data.



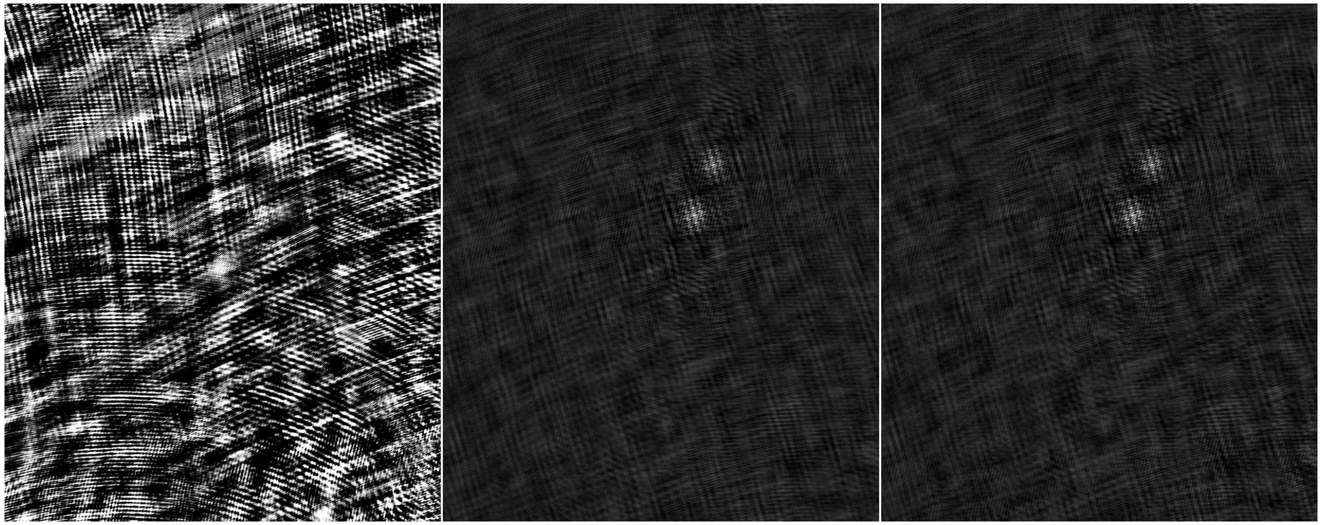
Raw data



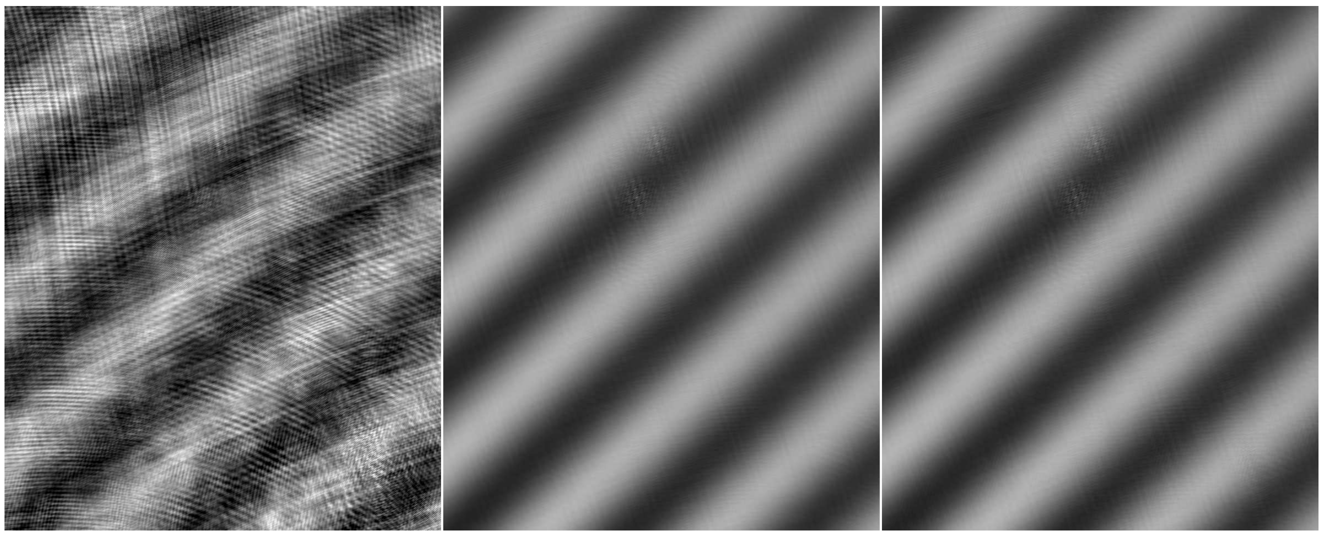
Residual after calibration

Stochastic calibration: (i) should NOT be affected by strong interference, (ii) should reveal weak interference, (iii) should reveal weak transient signals (fast radio bursts).

Images



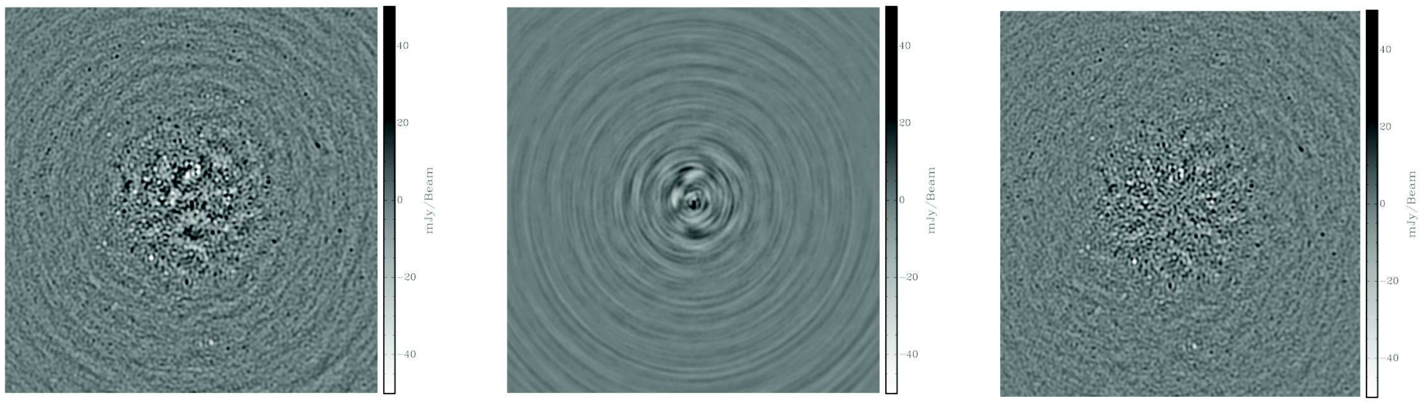
No interference



With interference

(left) raw data (middle) full batch (right) stochastic

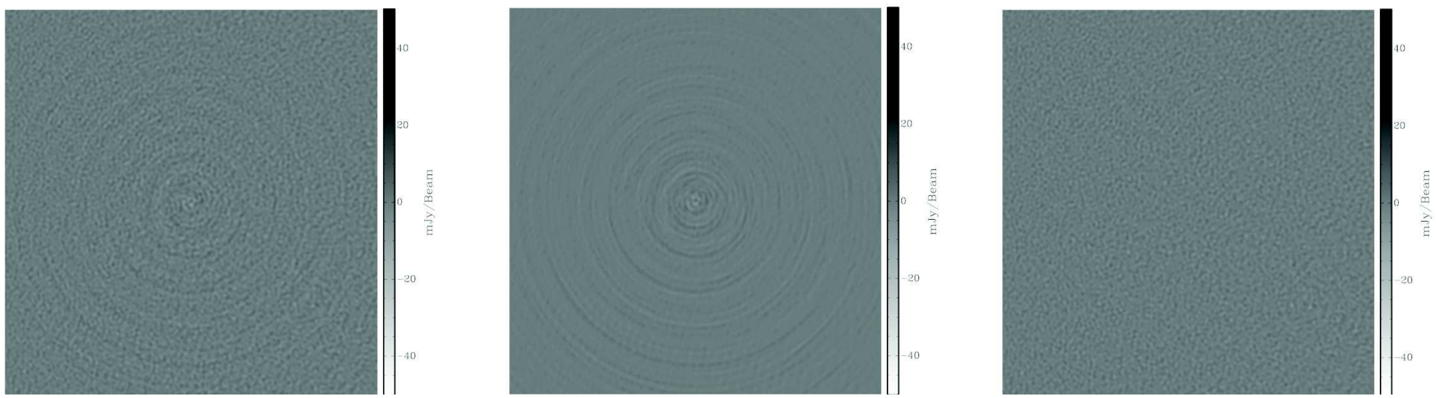
Real data example



Stokes I: before, difference, after
Stokes V: before , difference, after

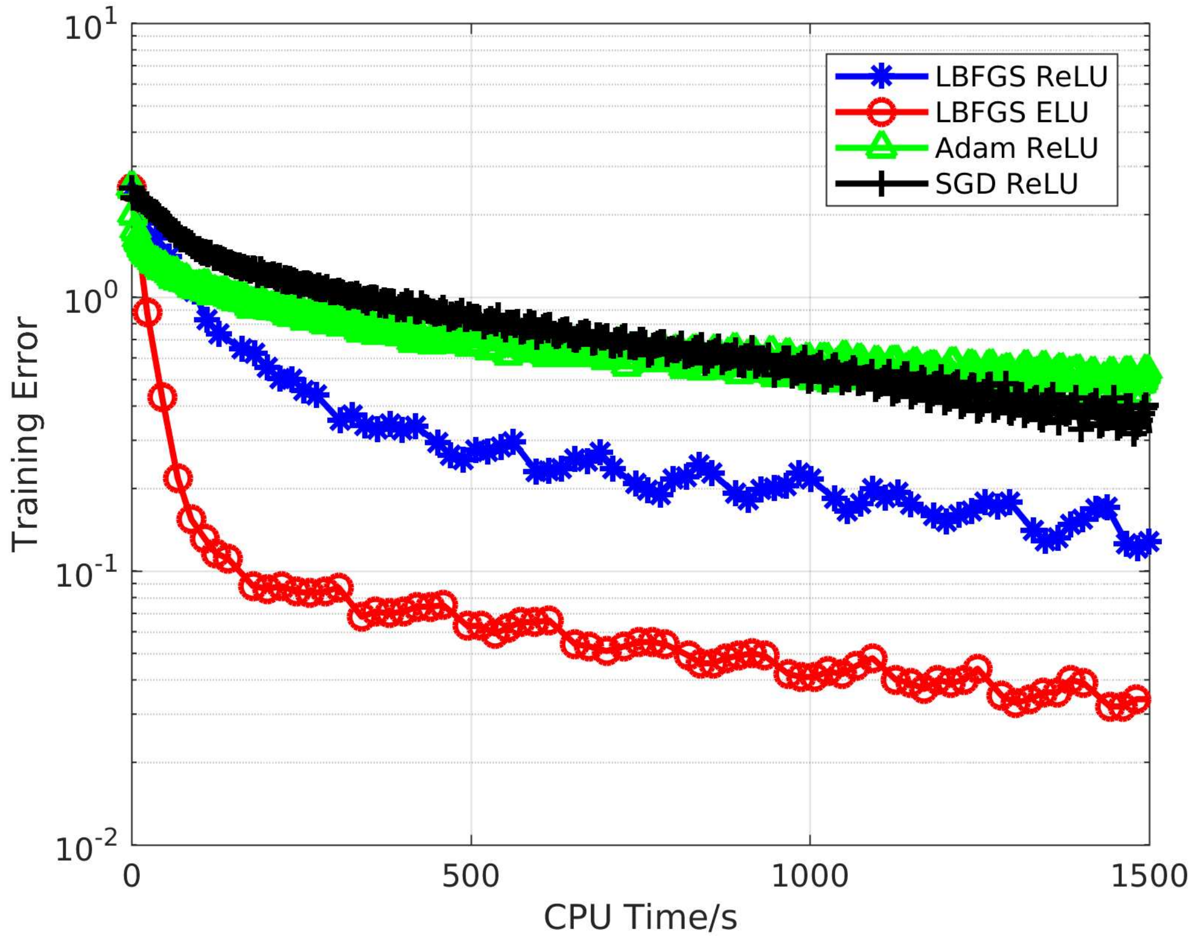
L412370 (Jan '16)
sb090-099
unif30-250

NCP



Residual images after one night of observation

Why not use SGD?



CIFAR 10 dataset

- ☐ Mini-batch size 32 (total 50000).
- ☐ LBFGS with `ReLU` and `ELU` activation.

Conclusions

- New stochastic LBFGS algorithm for stochastic calibration.
- Applications in other machine learning problems, code available in PyTorch.
- To do: Add consensus optimization for distributed stochastic calibration.
- To do: Integrate interference detection/filtering with stochastic calibration.
- To do: Apply to LOFAR raw data with weak interference.
- Code: https://github.com/nlesc-dirac