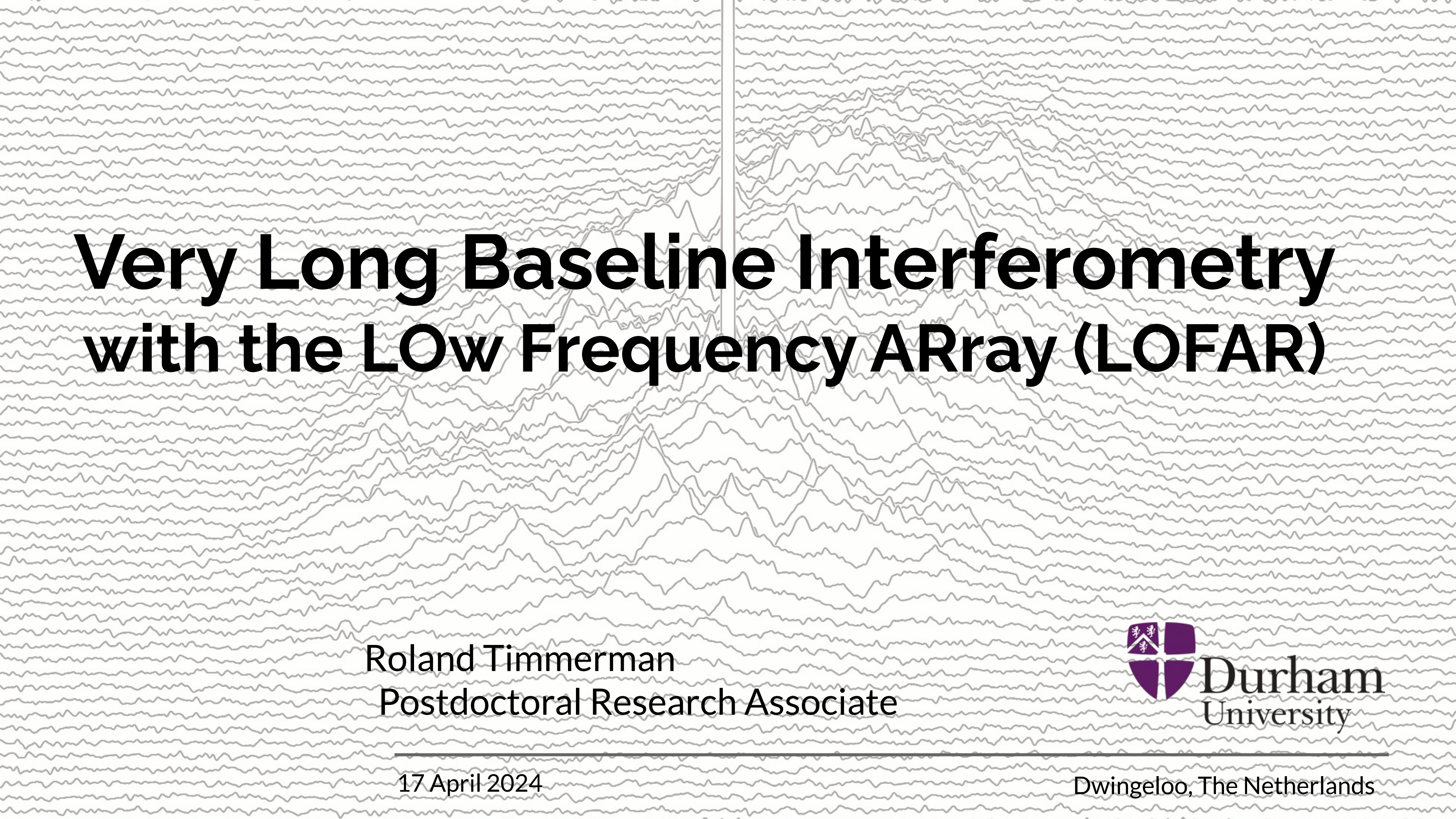


Very Long Baseline Interferometry with the LOw Frequency ARray (LOFAR)

Roland Timmerman
Postdoctoral Research Associate

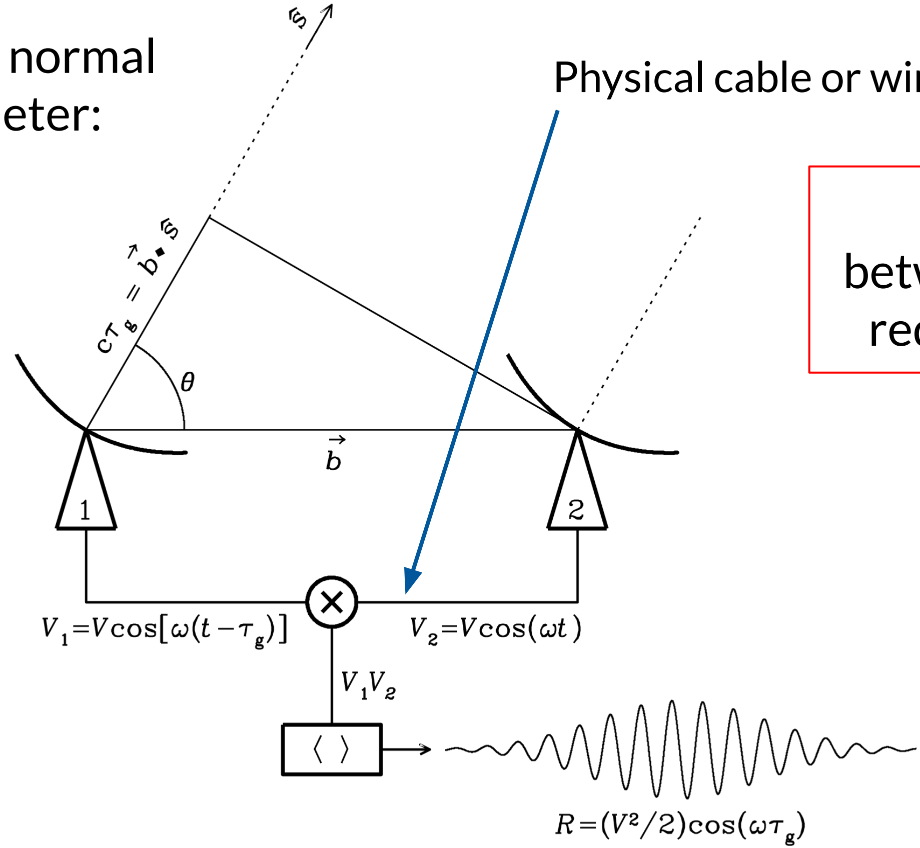
Durham University

17 April 2024

Dwingeloo, The Netherlands

What is Very Long Baseline Interferometry?

Consider a normal interferometer:



Physical cable or wireless transmission

In VLBI, the separations between antenna are too large, requiring separate recording

What is Very Long Baseline Interferometry?

Formally:

Interferometry where the signal of each antenna is recorded using **separate clocks**

Colloquially:

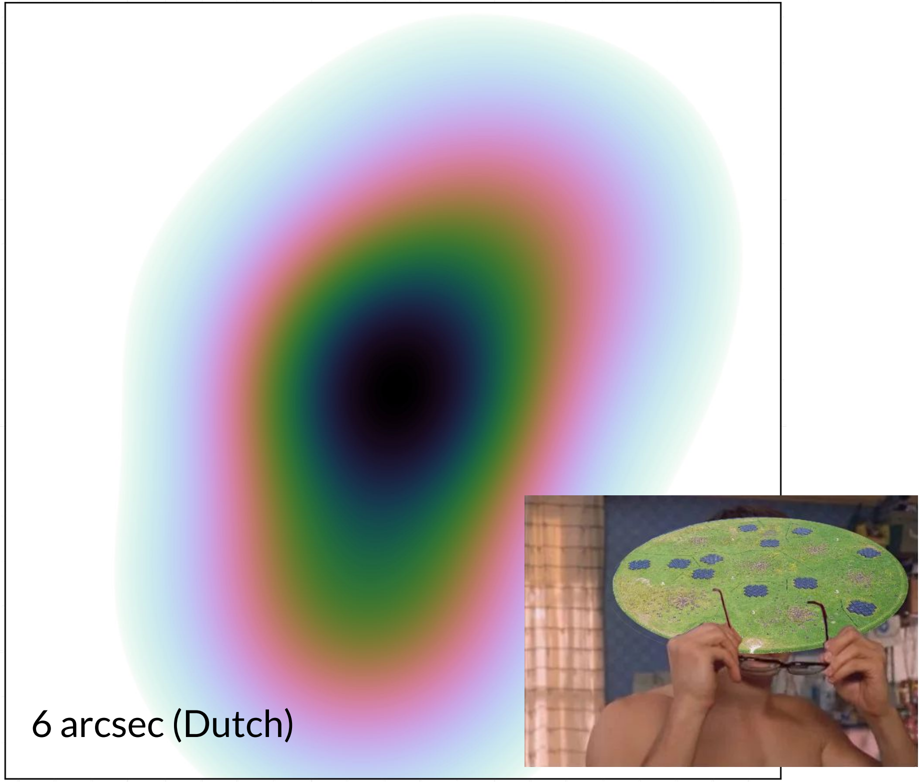
Interferometry with long enough baselines that you get the typical “VLBI problems”:

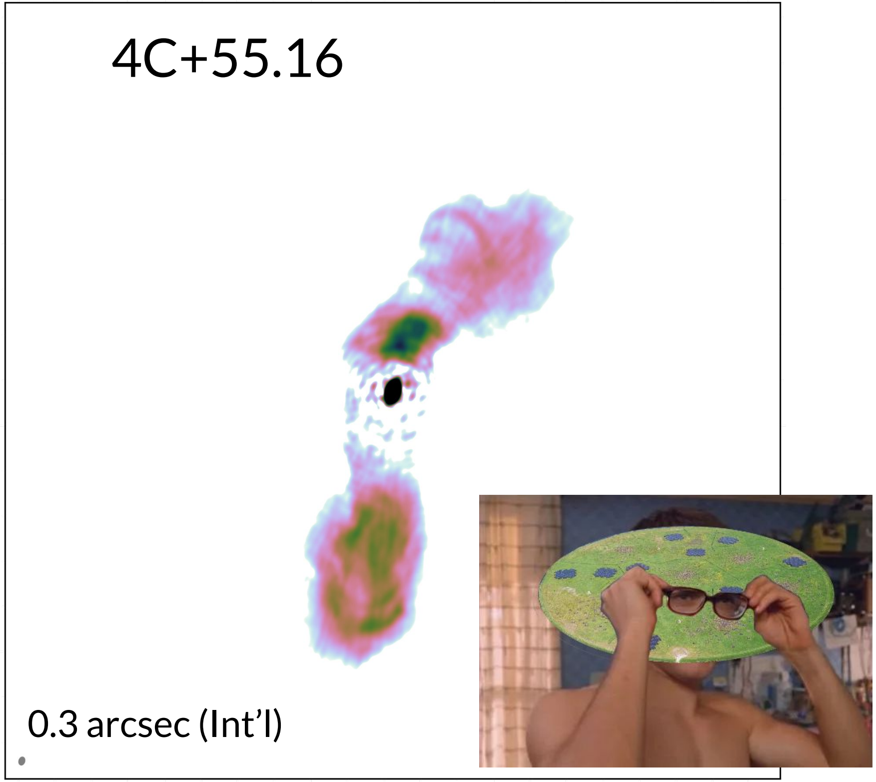
- Sparse array configurations with low degeneracy
- Decrease in S/N as flux decreases on longer baselines
- Strongly varying atmospheric conditions between the antennas
- Longer baselines increase requirements on imaging and data rates

But all of these problems are worth it, because of the spectacular **angular resolution!**

(Rule of thumb: if you Google for images and you get maps instead of photos, it’s probably VLBI)

Dutch LOFAR vs. International LOFAR Telescope


6 arcsec (Dutch)


4C+55.16

0.3 arcsec (Int'l)

VLBI: Sparse array configurations

General considerations:

- Large enough total effective area
- Cost-effective size per antenna
- Sampling of the uv-plane
- Wide variety in spacings
- Minimum baseline tends to be the size of the antenna
- Maximum baseline typically determined by target resolution or practical limits
- Side-lobes suppressed as much as possible
- All of the data also has to be processed…

Plus a lot of other details…

VLBI: Sparse array configurations


VLA


ALMA

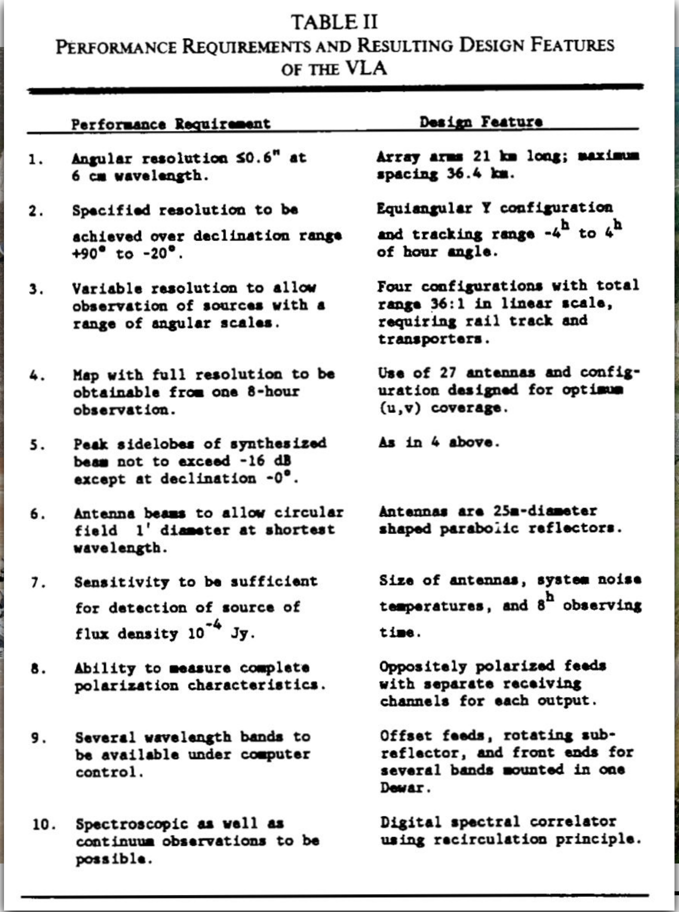


TABLE II
PERFORMANCE REQUIREMENTS AND RESULTING DESIGN FEATURES OF THE VLA

	Performance Requirement	Design Feature
1.	Angular resolution ≤0.6" at 6 cm wavelength.	Array arms 21 km long; maximum spacing 36.4 km.
2.	Specified resolution to be achieved over declination range +90° to -20°.	Equiangular Y configuration and tracking range -4^h to 4^h of hour angle.
3.	Variable resolution to allow observation of sources with a range of angular scales.	Four configurations with total range 36:1 in linear scale, requiring rail track and transporters.
4.	Map with full resolution to be obtainable from one 8-hour observation.	Use of 27 antennas and configuration designed for optimum (u,v) coverage.
5.	Peak sidelobes of synthesized beam not to exceed -16 dB except at declination -0°.	As in 4 above.
6.	Antenna beams to allow circular field 1' diameter at shortest wavelength.	Antennas are 25m-diameter shaped parabolic reflectors.
7.	Sensitivity to be sufficient for detection of source of flux density 10^{-4} Jy.	Size of antennas, system noise temperatures, and 8^h observing time.
8.	Ability to measure complete polarization characteristics.	Oppositely polarized feeds with separate receiving channels for each output.
9.	Several wavelength bands to be available under computer control.	Offset feeds, rotating subreflector, and front ends for several bands mounted in one Dewar.
10.	Spectroscopic as well as continuum observations to be possible.	Digital spectral correlator using recirculation principle.


WSRT

General considerations:

- Large enough total effective area
- Cost-effective size per antenna
- Sampling of the uv-plane
- Wide variety in spacings
- Minimum baseline tends to be the size of the antenna
- Maximum baseline typically determined by target resolution or practical limits
- Side-lobes suppressed as much as possible
- All of the data also has to be processed…

Plus a lot of other details…

VLBI: Sparse array configurations

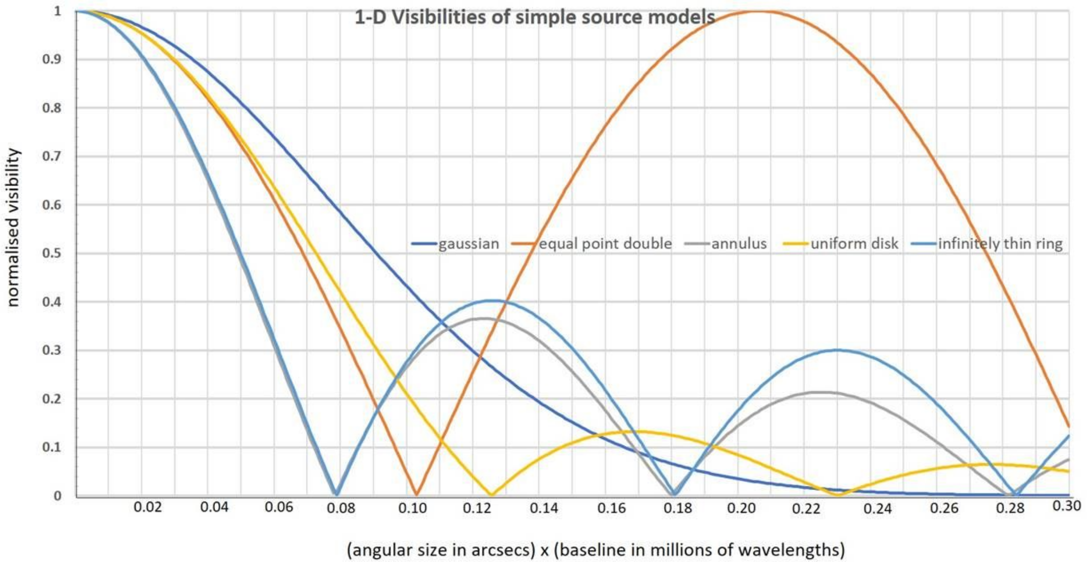
- Cost per antenna typically high (Separate clocks, infrastructure, etc…)
- Strong geographical constraints (Oceans, mountains, country borders, political and economic situations…)

- Maximum baseline determined by size of our planet

→ Typically fewer stations, less optimal configurations, sparse coverage

VLBI: Decreasing S/N at high resolutions

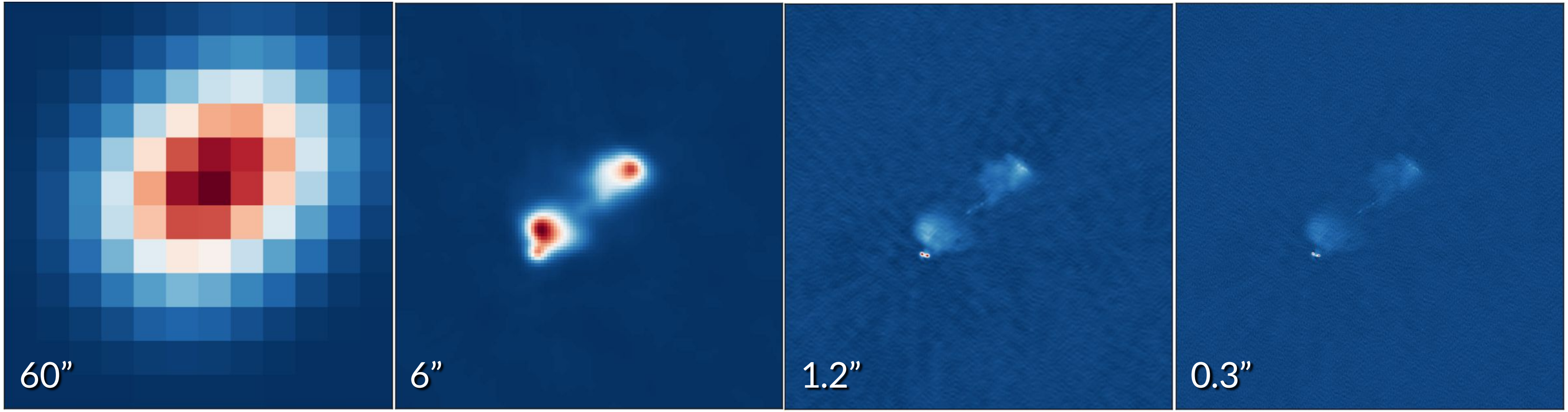
Remember: visibilities probe the **Fourier transform** of the sky brightness distribution



Only a perfect point source has as much flux on a long baseline as on a short baseline

VLBI: Decreasing S/N at high resolutions

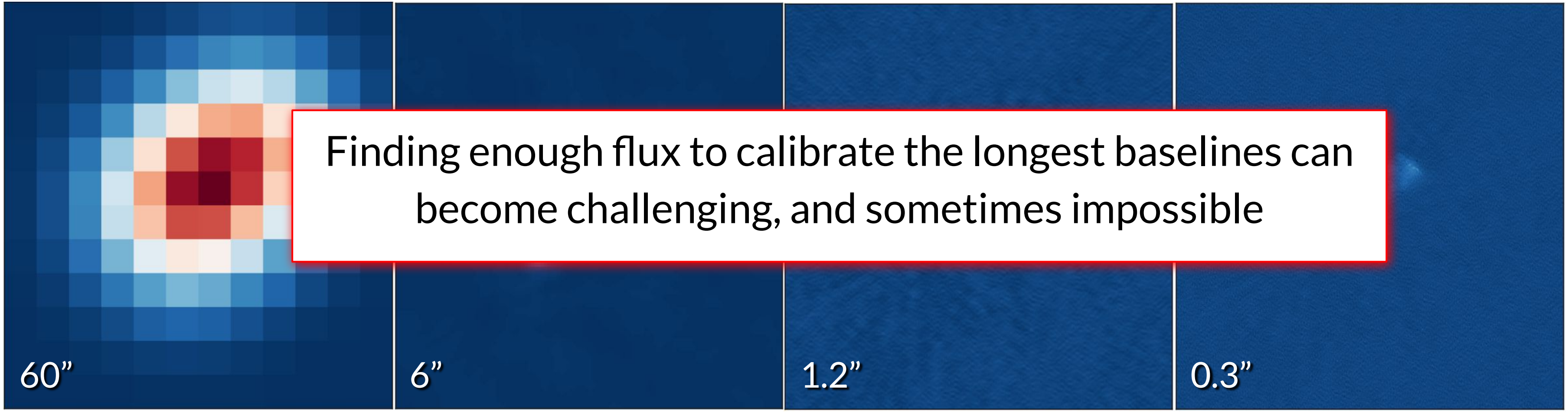
But wait… it's actually even worse!



Higher resolution → Smaller synthesized beam (i.e. resolution or pixels)
→ The same total intensity is distributed over more resolution elements
→ The intensity per resolution element is <u>reduced</u>

VLBI: Decreasing S/N at high resolutions

But wait… it’s actually even worse!



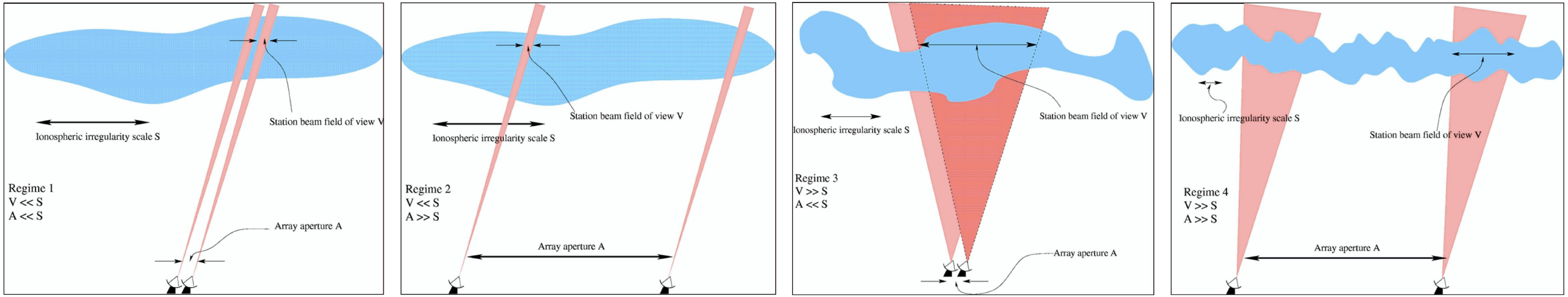
Finding enough flux to calibrate the longest baselines can become challenging, and sometimes impossible

Higher resolution → Smaller synthesized beam (i.e. resolution or pixels)
→ The same total intensity is distributed over more resolution elements
→ The intensity per resolution element is reduced

VLBI: Strongly varying atmospheric conditions

The four regimes:

(Credit: Wijnholds et al., 2010)



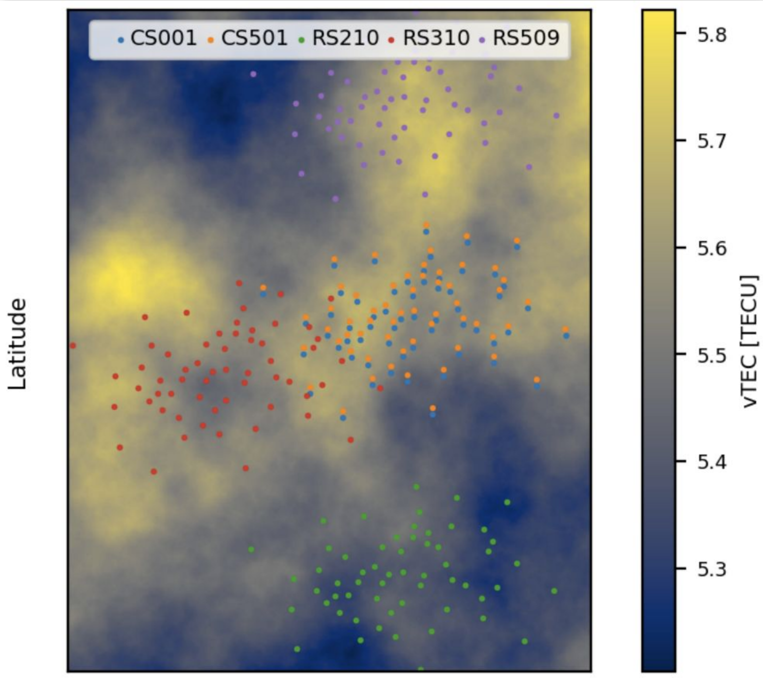
LOFAR core

Long-baseline LOFAR

↑

Also the most
difficult one...

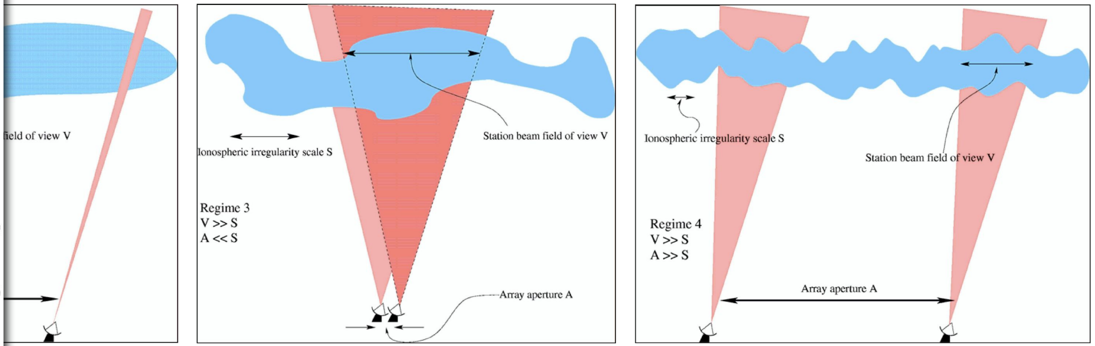
VLBI: Strongly varying atmospheric conditions



(Credit: Edler et al., 2021)

The four regimes:

(Credit: Wijnholds et al., 2010)



LOFAR core

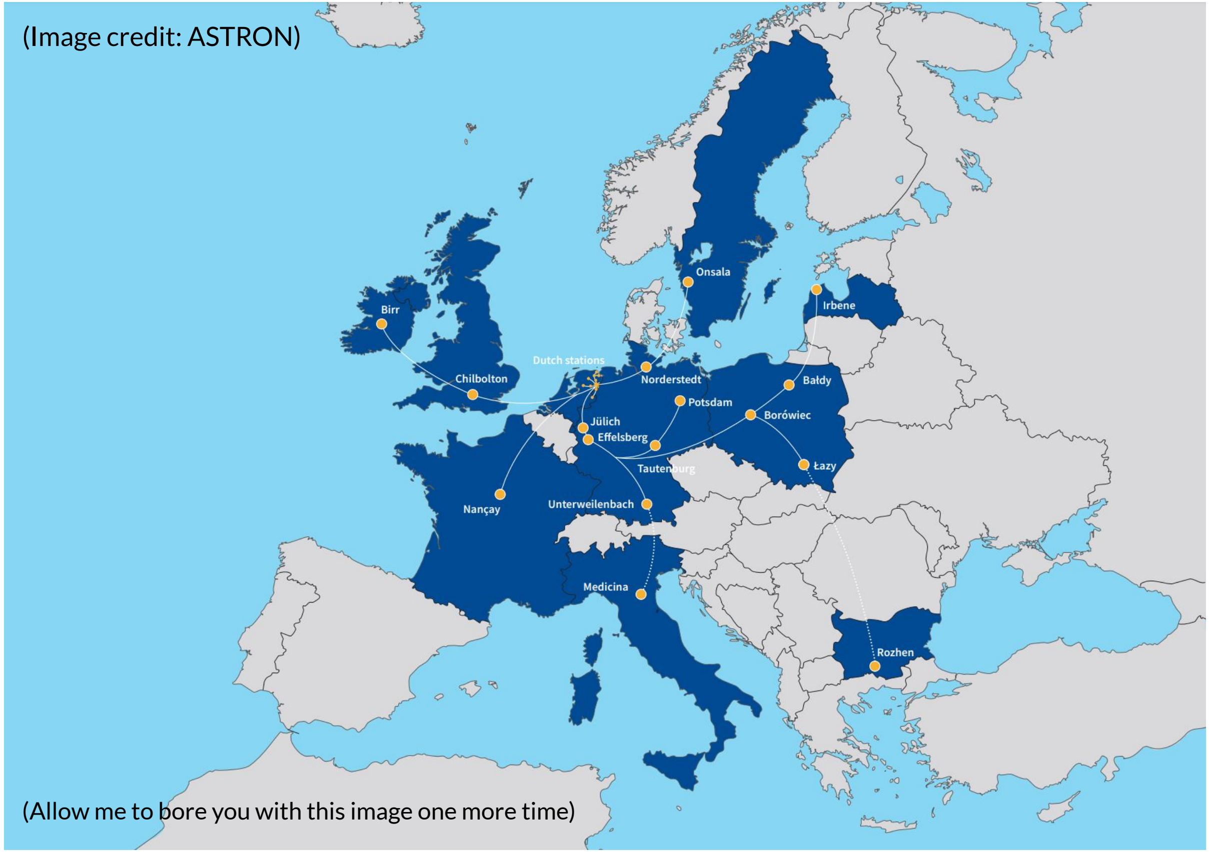
Long-baseline LOFAR

The ionosphere as seen by LOFAR strongly varies across the Field of View

Also the most difficult one...

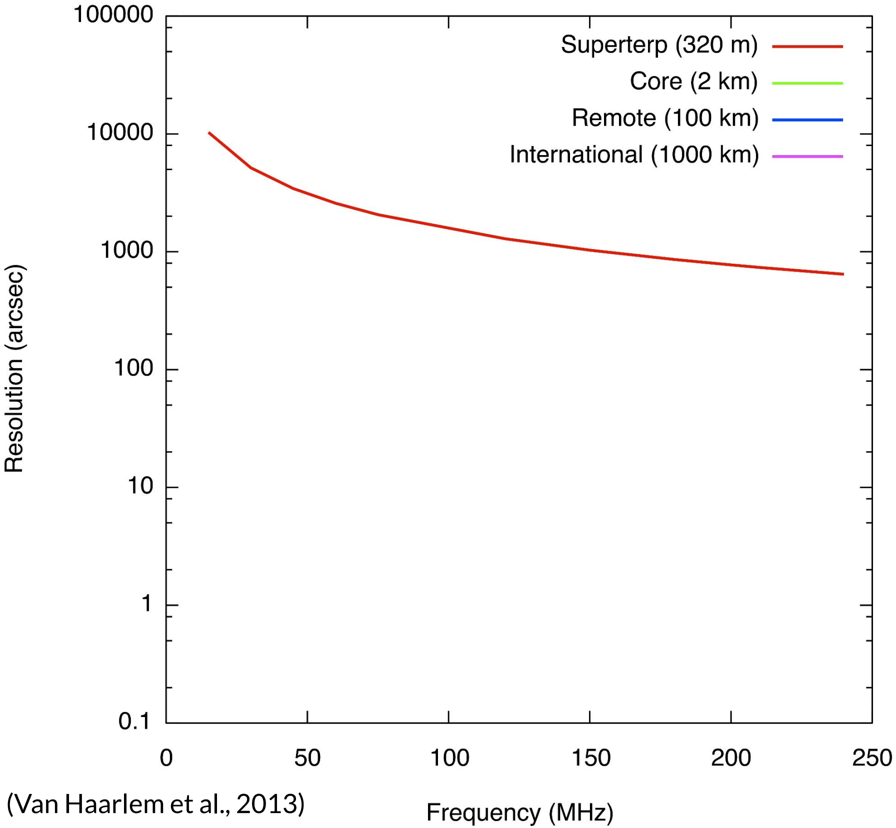
The International LOFAR Telescope

- Low-frequency phased array
    - LBA: 10-90 MHz
    - HBA: 110-240 MHz
- Baselines from 68 m to ~1890 km (soon even more!)
- Multi-beam observing capabilities
- Wide field of view
- LOFAR 2.0 upgrade incoming!


(Image credit: ASTRON)

(Allow me to bore you with this image one more time)

The International LOFAR Telescope

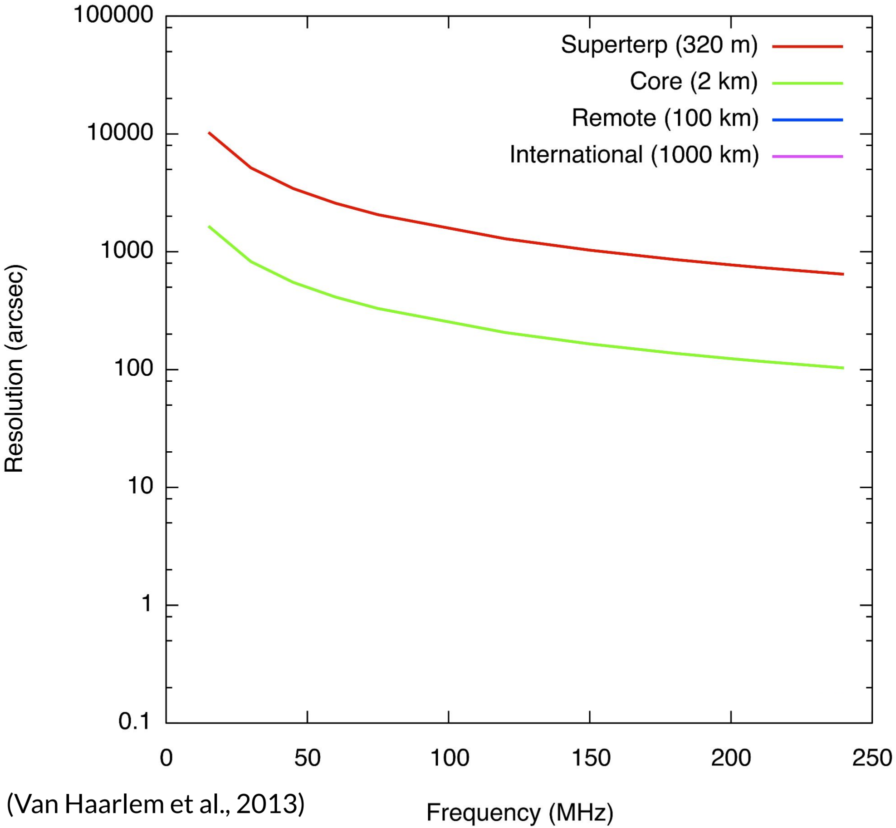


(Van Haarlem et al., 2013)

17 April 2024

Dwingeloo, The Netherlands

The International LOFAR Telescope

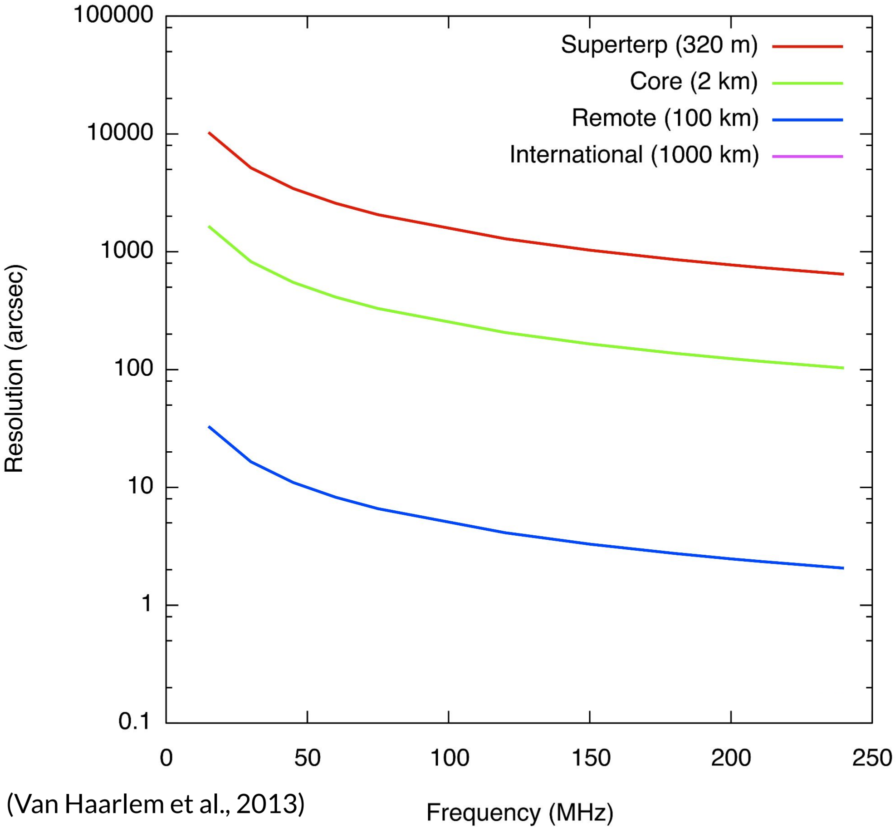


(Van Haarlem et al., 2013)

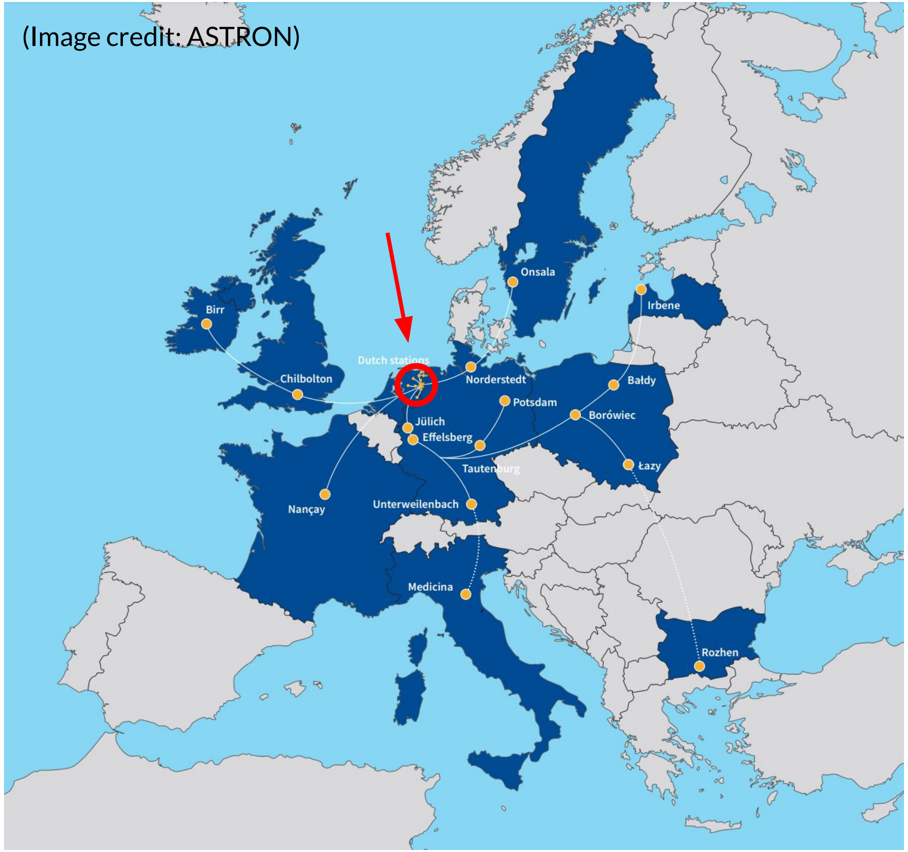
17 April 2024

Dwingeloo, The Netherlands

The International LOFAR Telescope

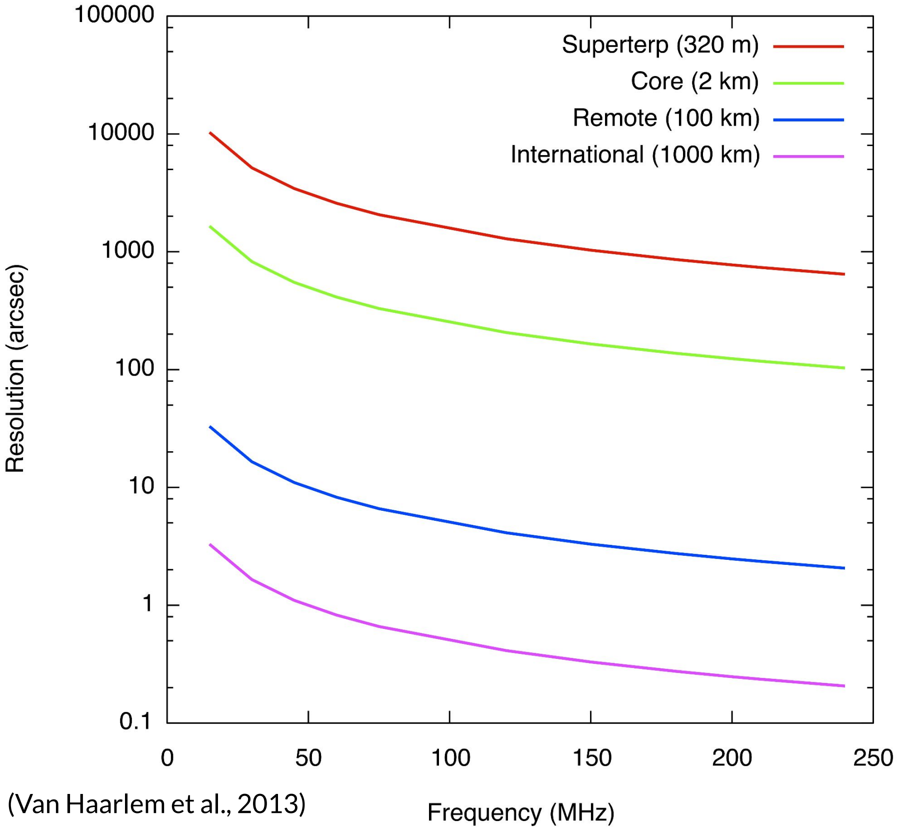


(Van Haarlem et al., 2013)

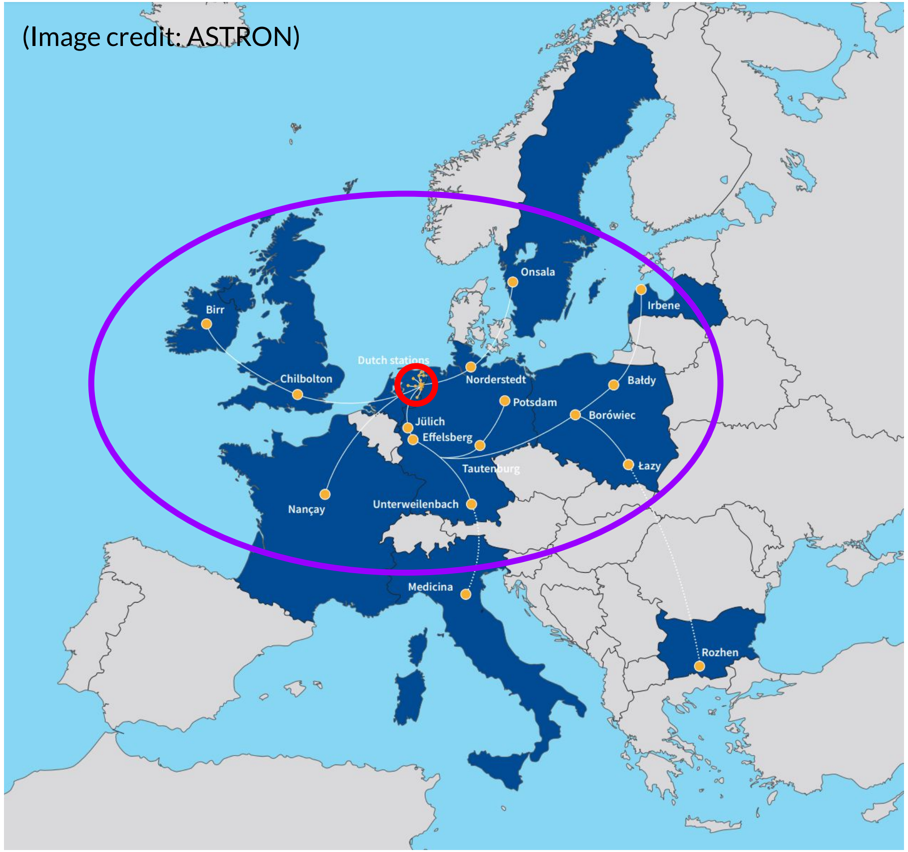


(Image credit: ASTRON)

The International LOFAR Telescope

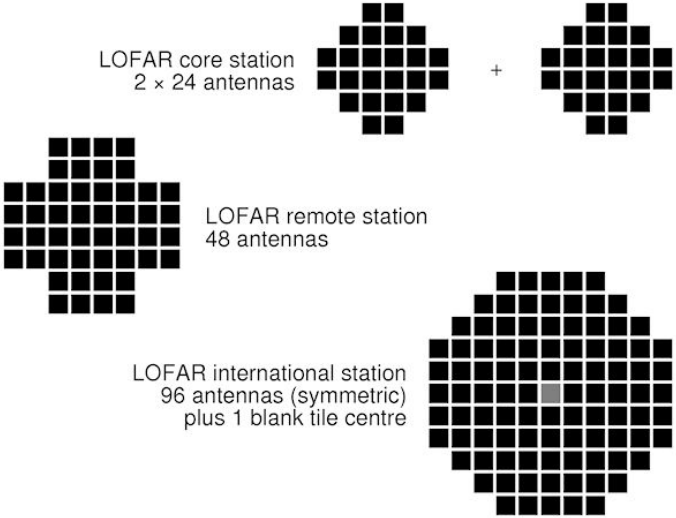


(Van Haarlem et al., 2013)

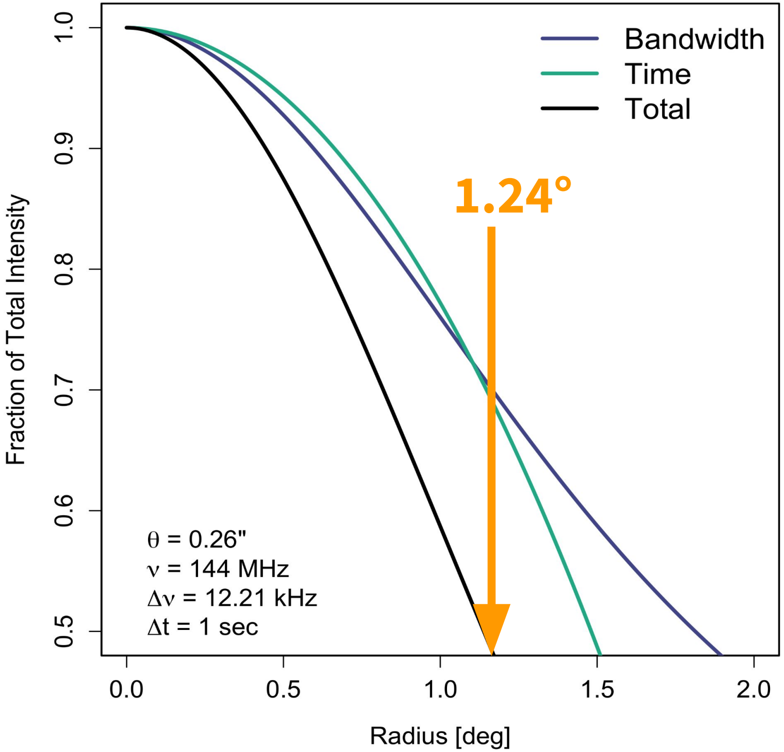


(Image credit: ASTRON)

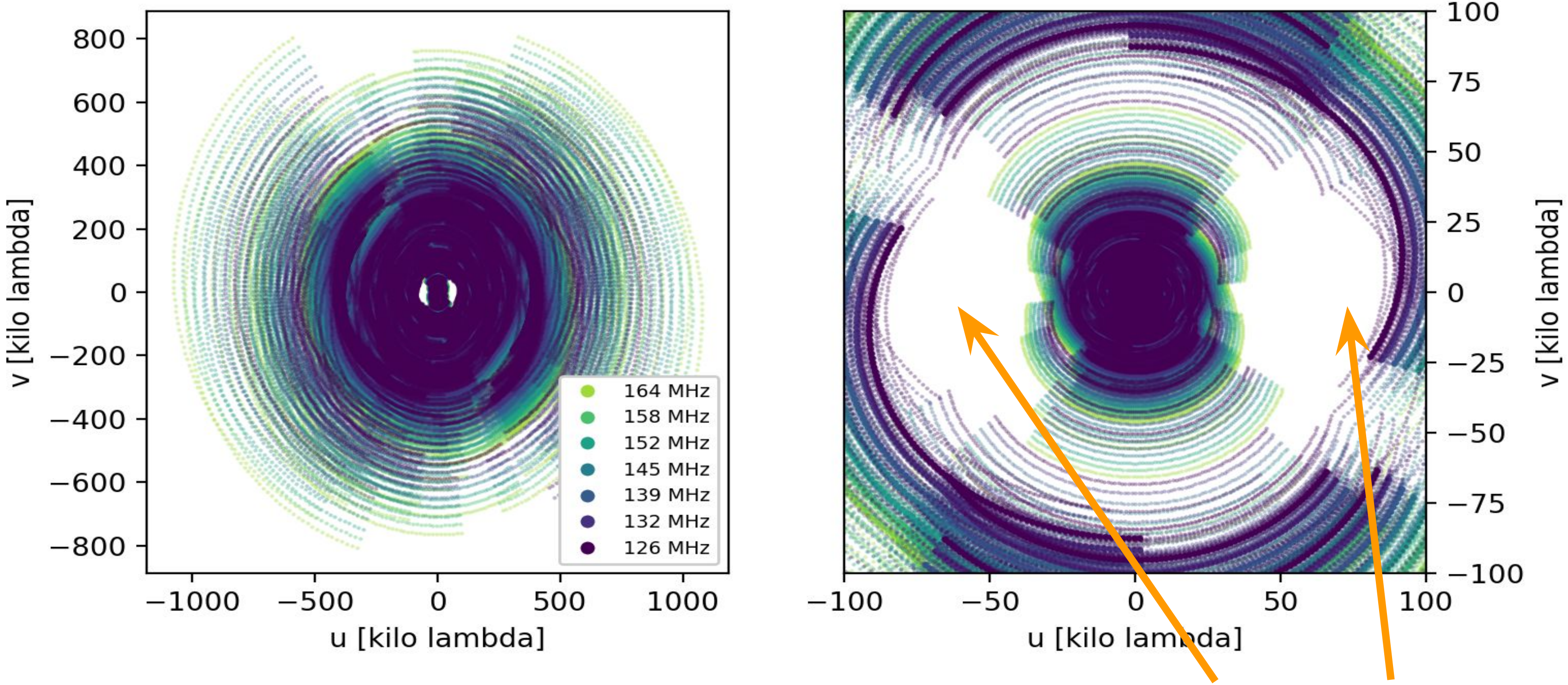
The International LOFAR Telescope



The Field of View (FoV) of the international stations is narrower than that of the Dutch stations!



The International LOFAR Telescope



The “Belgian gap”

(It’s actually in North Holland and Germany…)

Low radio frequencies

Standard (high freq.) VLBI uses **fringe fitting**

$$\Delta\phi_{\nu,f} = \phi_0 + \left(\frac{\delta\phi}{\delta\nu}\Delta\nu + \frac{\delta\phi}{\delta t}\Delta t\right)$$

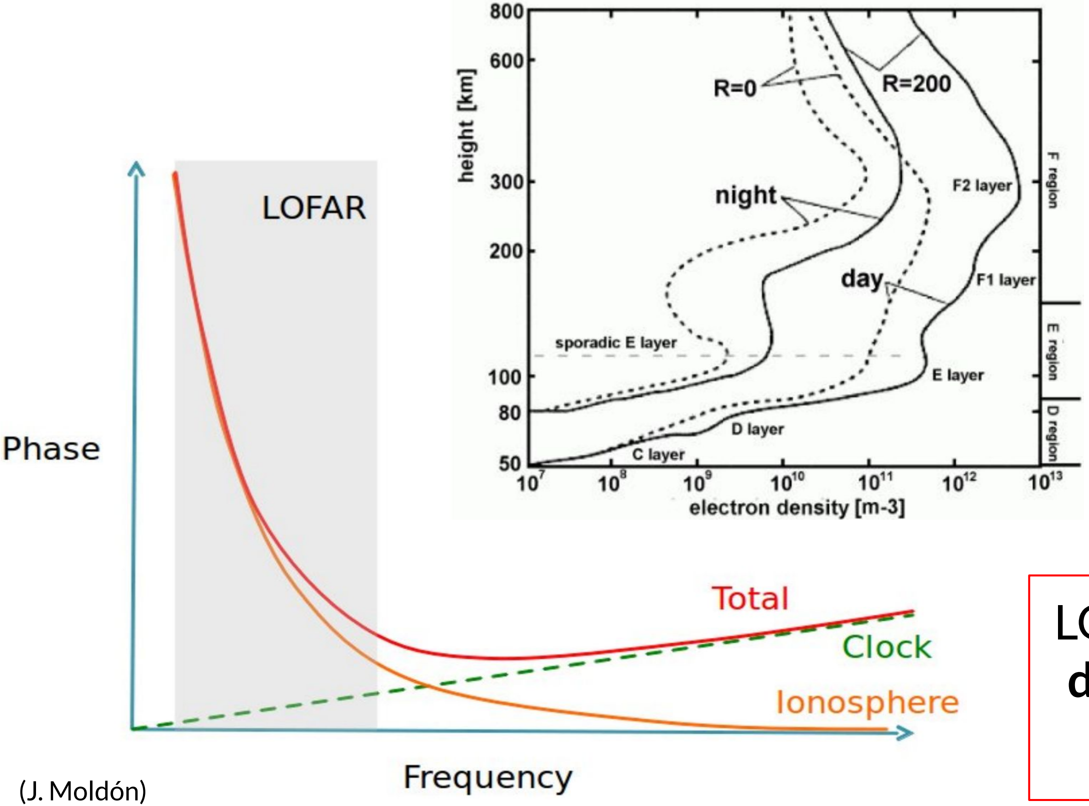
$\phi_0 \equiv$ the **phase** error at t_0, ν_0

$\frac{\delta\phi}{\delta\nu} \equiv$ **delay** or delay residual

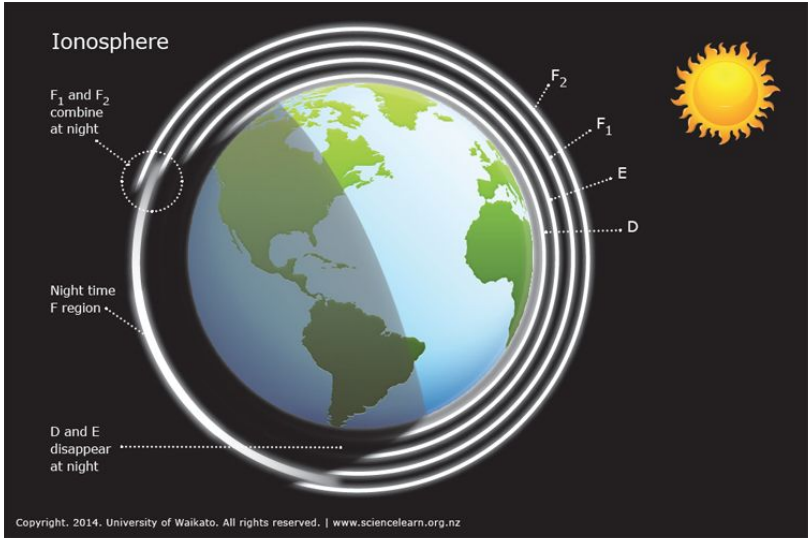
- **Dispersive** (e.g. ionosphere)
- **Non-dispersive** (e.g. clock errors)

$\frac{\delta\phi}{\delta t} \equiv$ **rate**, delay rate, or delay residual

Low radio frequencies



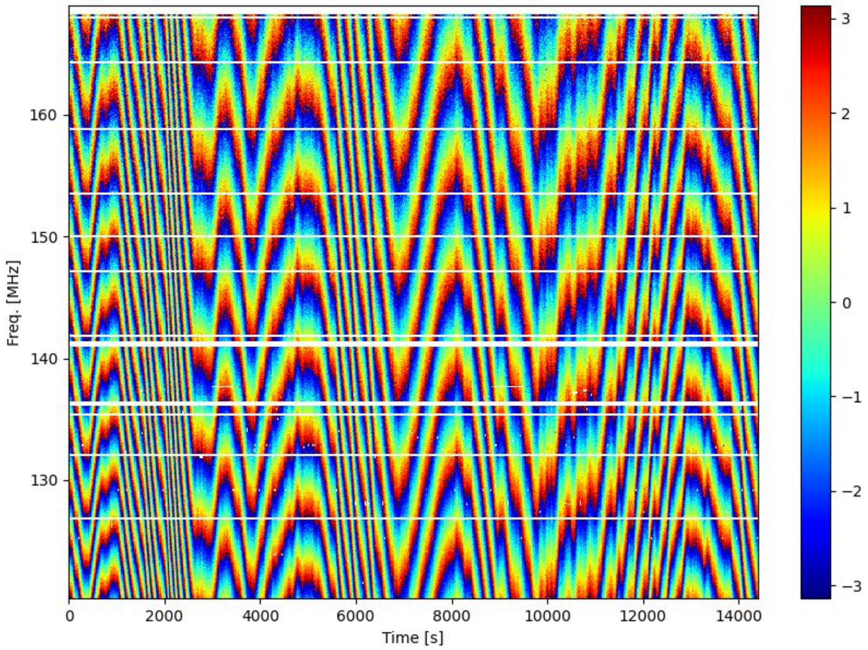
(J. Moldón)

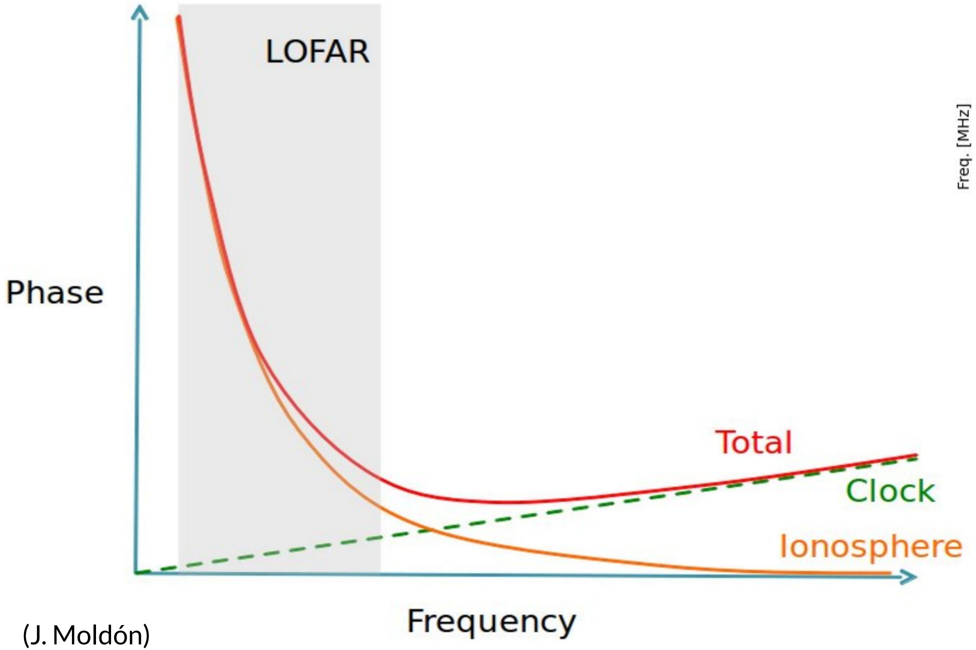


LOFAR is dominated by **dispersive delays** caused by the ionosphere, which scale as $\varphi \propto \nu^{-1}$

Low radio frequencies

Okay

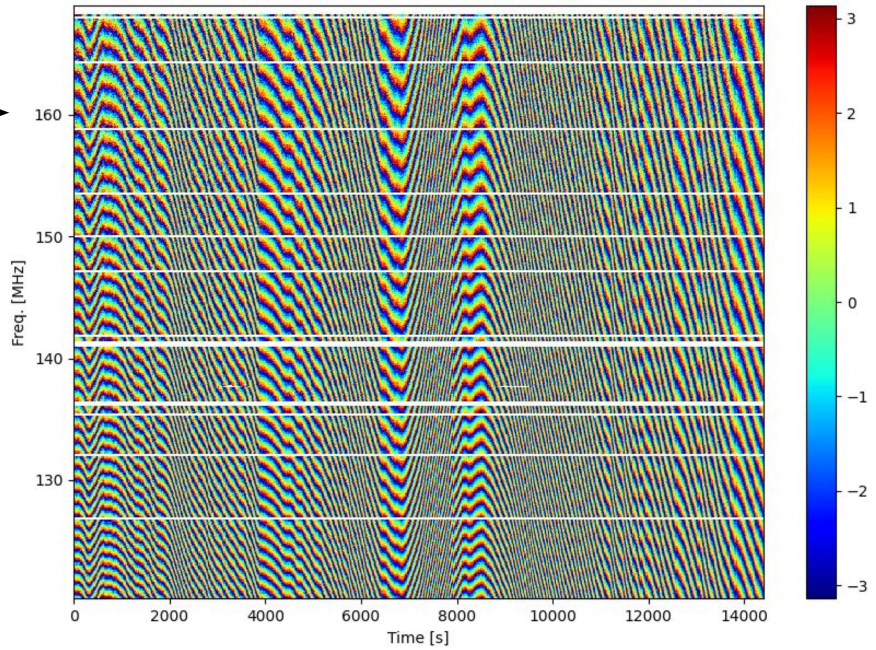


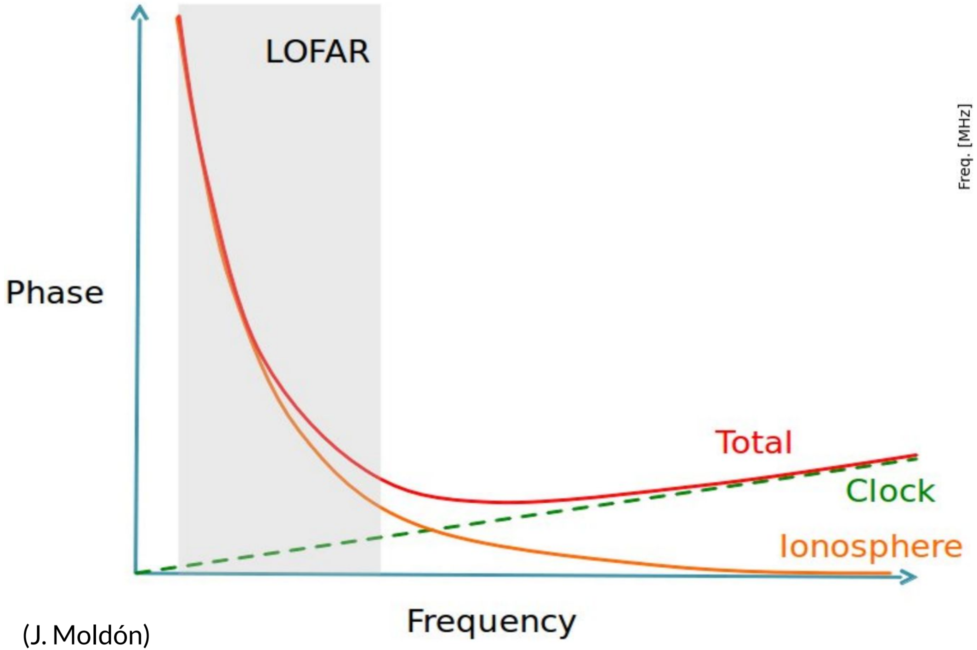


(J. Moldón)

Low radio frequencies

Bad

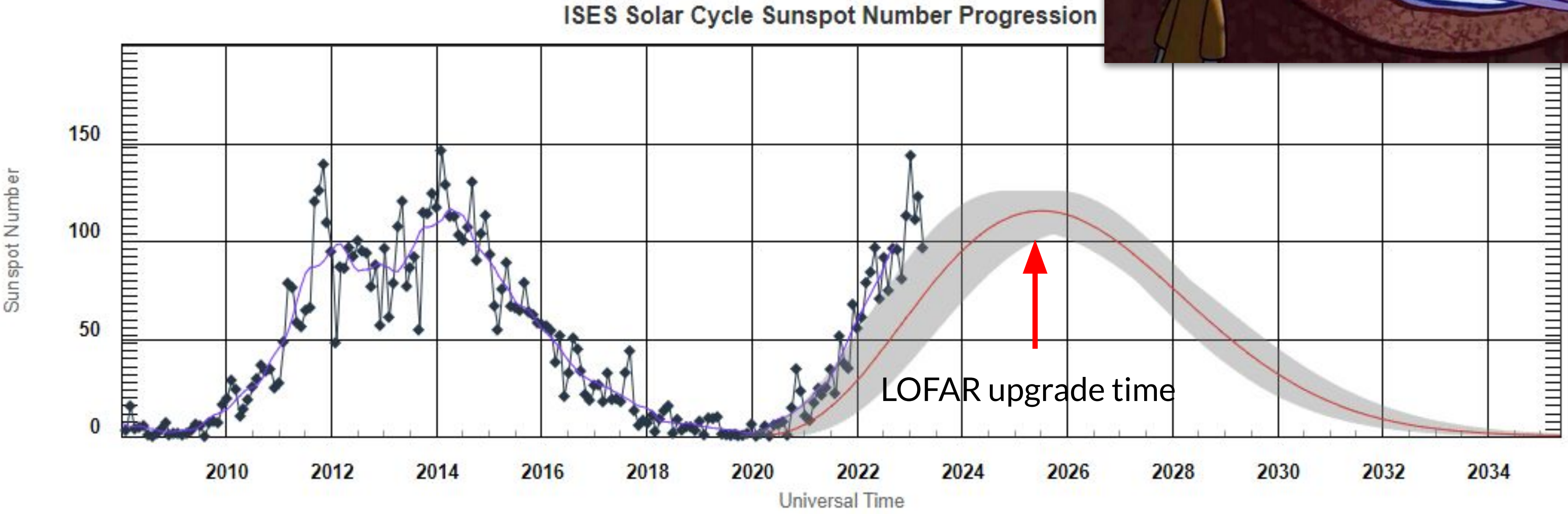




(J. Moldón)

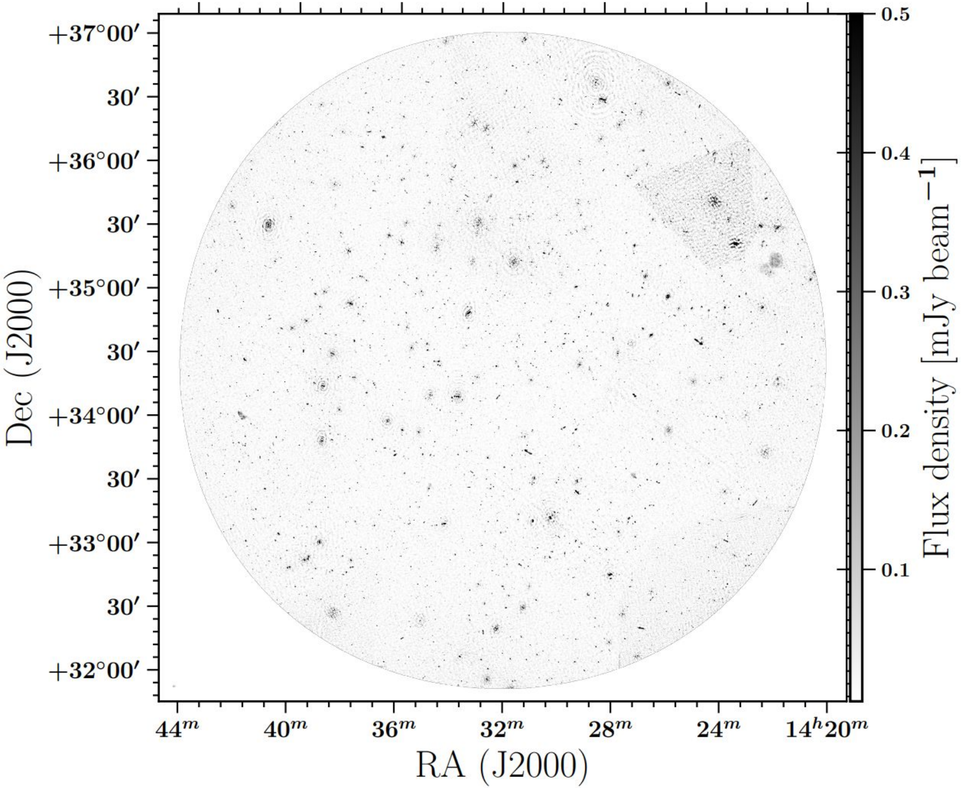
The solar cycle

More active Sun
→ more active ionosphere
→ higher calibration difficulty



LOFAR upgrade time

Low radio frequencies



(Retana-Montenegro et al., 2018)

Large Field of Views
→ Strong interference from nearby sources

(even up to ~30 degrees away!)

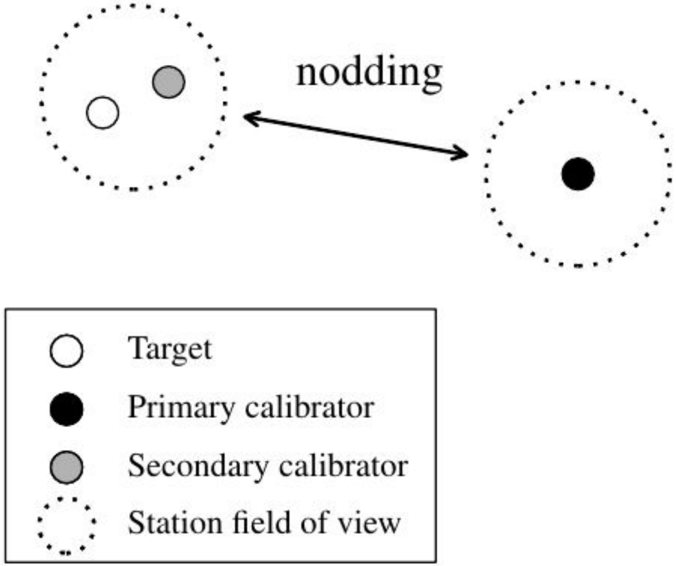
To make it worse: the heterogeneous array has different FoVs!

Dutch array: ~2.5 degree radius
Int’l array: ~1.2 degree radius

The Long Baseline Calibrator Survey

LOFAR can observe both its science target and a calibrator at the same time

LBCS source (in-field or delay calibrator)



3C48
3C147
3C196
3C295

LBCS source selection:

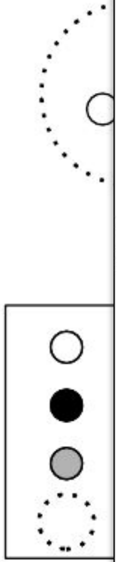
- Compact in WENSS (above 30° N)
- Compact in NVSS (below 30° N)
- Also detected by VLSSr, MSSS or TGSS
- The VLBA calibrators

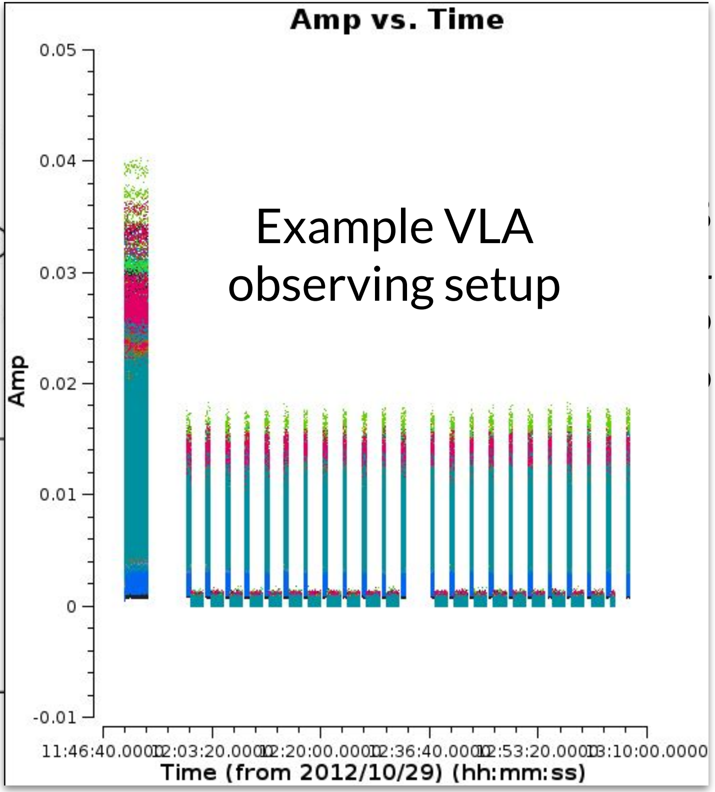
A good in-field calibrator is both **bright** at low radio frequencies (≳0.2 Jy) and **compact** at angular scales of 0.3 arcseconds.

The Long Baseline Calibrator Survey

LOFAR can observe both its science target and a calibrator at the same time

LBCS source (i





Example VLA observing setup

LBCS source selection:

- Compact in WENSS (above 30° N)
- Compact in NVSS (below 30° N)
- Also detected by VLSSr, MSSS or TGSS
- The VLBA calibrators

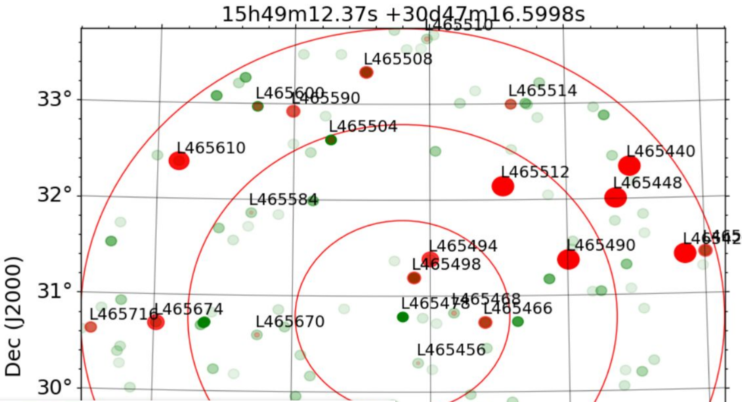
A good in-field calibrator is both **bright** at low radio frequencies (≳0.2 Jy) and **compact** at angular scales of 0.3 arcseconds.

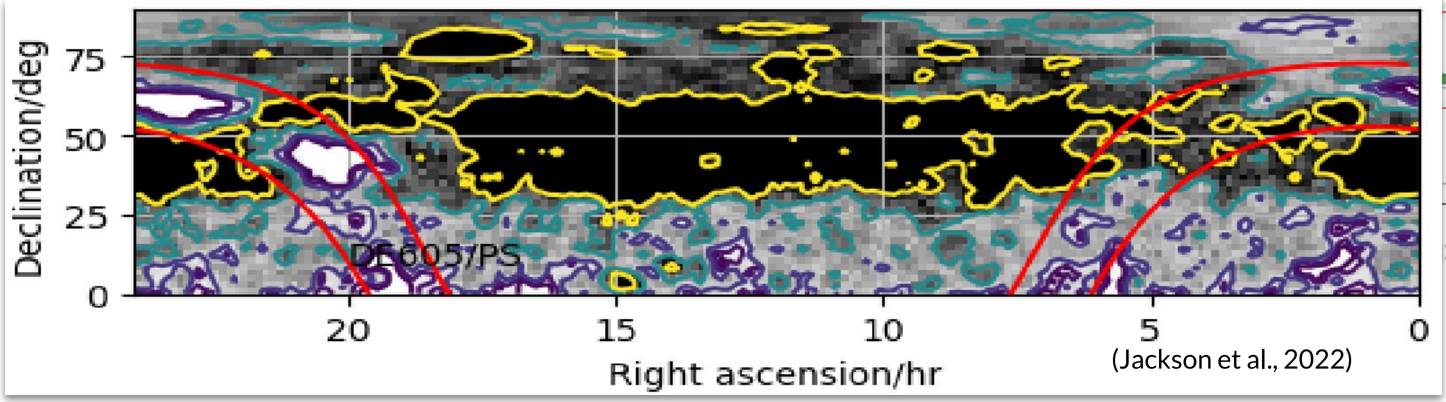
The Long Baseline Calibrator Survey

Typically, you will find ~1 in-field calibrator per square degree

Though sometimes you can be unlucky…

(especially towards low Dec. and the Galactic Plane)

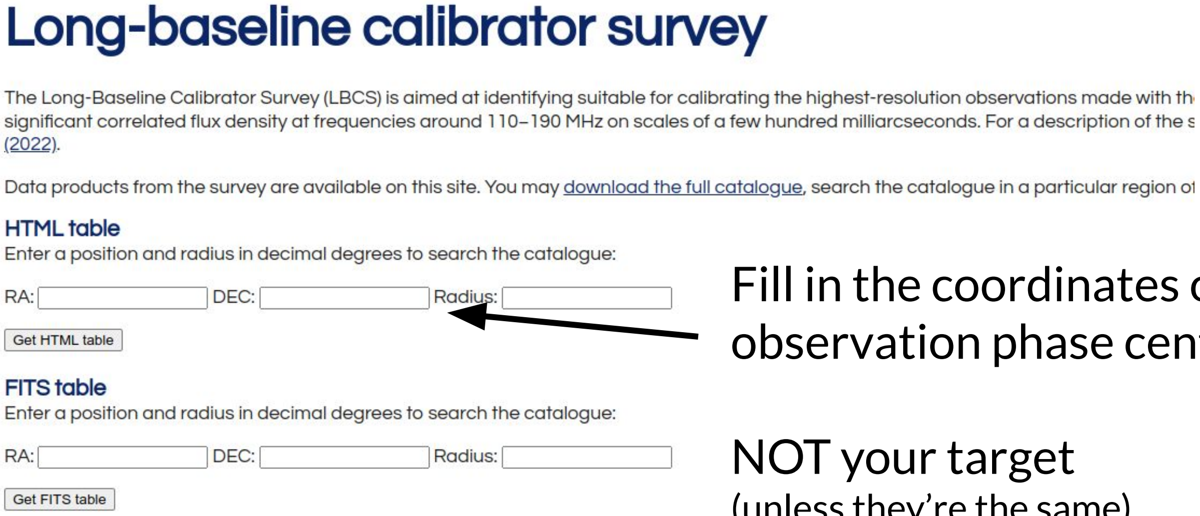




(Jackson et al., 2022)

Intermezzo: some practicalities...

lofar-surveys.org/lbcs.html



Long-baseline calibrator survey

The Long-Baseline Calibrator Survey (LBCS) is aimed at identifying suitable for calibrating the highest-resolution observations made with th significant correlated flux density at frequencies around 110–190 MHz on scales of a few hundred milliarcseconds. For a description of the s (2022).

Data products from the survey are available on this site. You may download the full catalogue, search the catalogue in a particular region of

HTML table

Enter a position and radius in decimal degrees to search the catalogue:

RA: DEC: Radius:

FITS table

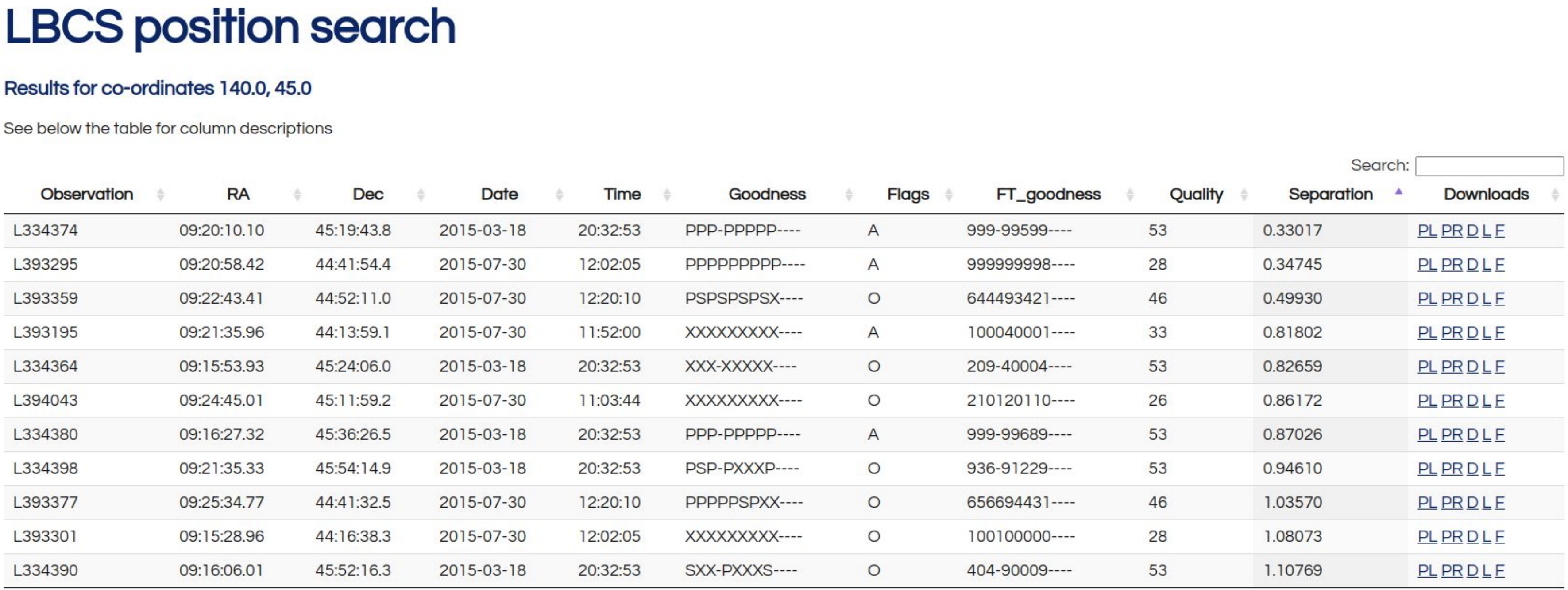
Enter a position and radius in decimal degrees to search the catalogue:

RA: DEC: Radius:

Fill in the coordinates of your observation phase center

NOT your target
(unless they're the same)

Intermezzo: some practicalities…



LBCS position search

Results for co-ordinates 140.0, 45.0

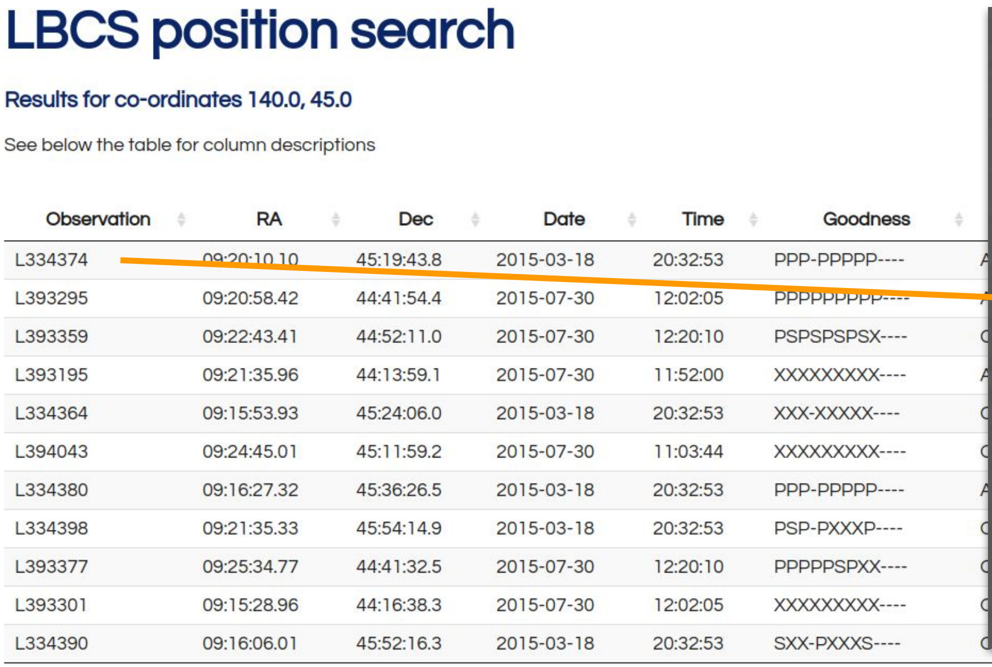
See below the table for column descriptions

Observation	RA	Dec	Date	Time	Goodness	Flags	FT_goodness	Quality	Separation	Downloads
L334374	09:20:10.10	45:19:43.8	2015-03-18	20:32:53	PPP-PPPPP----	A	999-99599----	53	0.33017	PL PR D L F
L393295	09:20:58.42	44:41:54.4	2015-07-30	12:02:05	PPPPPPPPP----	A	999999998----	28	0.34745	PL PR D L F
L393359	09:22:43.41	44:52:11.0	2015-07-30	12:20:10	PSPSPSPSX----	O	644493421----	46	0.49930	PL PR D L F
L393195	09:21:35.96	44:13:59.1	2015-07-30	11:52:00	XXXXXXXXX----	A	100040001----	33	0.81802	PL PR D L F
L334364	09:15:53.93	45:24:06.0	2015-03-18	20:32:53	XXX-XXXXX----	O	209-40004----	53	0.82659	PL PR D L F
L394043	09:24:45.01	45:11:59.2	2015-07-30	11:03:44	XXXXXXXXX----	O	210120110----	26	0.86172	PL PR D L F
L334380	09:16:27.32	45:36:26.5	2015-03-18	20:32:53	PPP-PPPPP----	A	999-99689----	53	0.87026	PL PR D L F
L334398	09:21:35.33	45:54:14.9	2015-03-18	20:32:53	PSP-PXXXP----	O	936-91229----	53	0.94610	PL PR D L F
L393377	09:25:34.77	44:41:32.5	2015-07-30	12:20:10	PPPPPSPXX----	O	656694431----	46	1.03570	PL PR D L F
L393301	09:15:28.96	44:16:38.3	2015-07-30	12:02:05	XXXXXXXXX----	O	100100000----	28	1.08073	PL PR D L F
L334390	09:16:06.01	45:52:16.3	2015-03-18	20:32:53	SXX-PXXXS----	O	404-90009----	53	1.10769	PL PR D L F

P: good; S: marginal; X: bad

FT goodness: higher is better

Intermezzo: some practicalities...

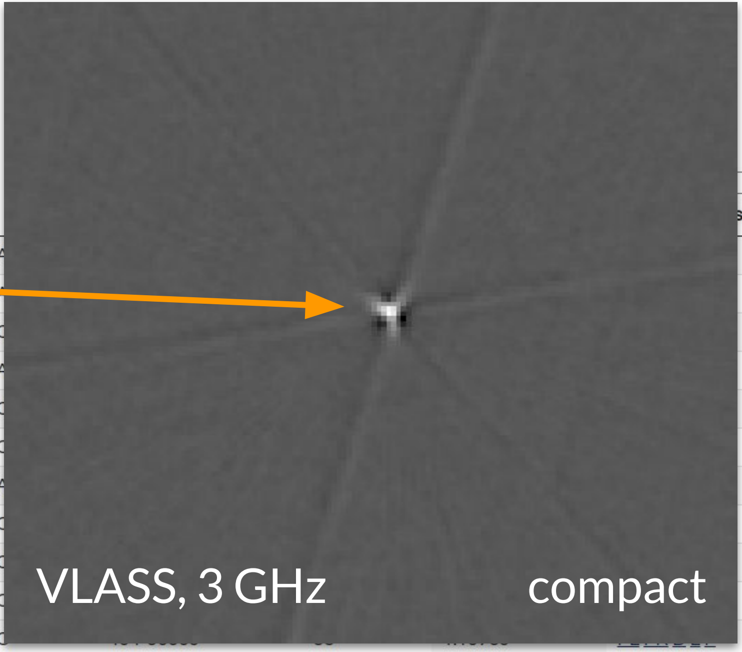


LBCS position search

Results for co-ordinates 140.0, 45.0

See below the table for column descriptions

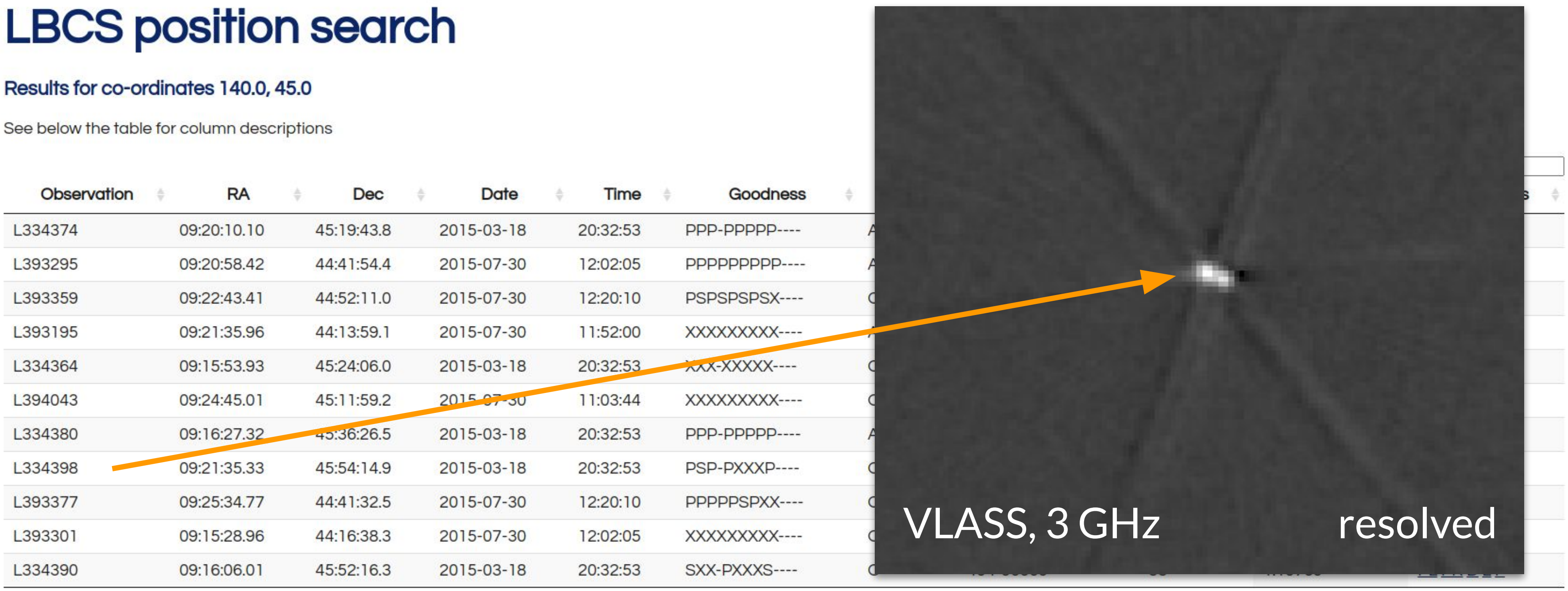
Observation	RA	Dec	Date	Time	Goodness
L334374	09:20:10.10	45:19:43.8	2015-03-18	20:32:53	PPP-PPPPP----
L393295	09:20:58.42	44:41:54.4	2015-07-30	12:02:05	PPPPPPPPP----
L393359	09:22:43.41	44:52:11.0	2015-07-30	12:20:10	PSPSPSPSX----
L393195	09:21:35.96	44:13:59.1	2015-07-30	11:52:00	XXXXXXXXX----
L334364	09:15:53.93	45:24:06.0	2015-03-18	20:32:53	XXX-XXXXX----
L394043	09:24:45.01	45:11:59.2	2015-07-30	11:03:44	XXXXXXXXX----
L334380	09:16:27.32	45:36:26.5	2015-03-18	20:32:53	PPP-PPPPP----
L334398	09:21:35.33	45:54:14.9	2015-03-18	20:32:53	PSP-PXXXP----
L393377	09:25:34.77	44:41:32.5	2015-07-30	12:20:10	PPPPPSPXX----
L393301	09:15:28.96	44:16:38.3	2015-07-30	12:02:05	XXXXXXXXX----
L334390	09:16:06.01	45:52:16.3	2015-03-18	20:32:53	SXX-PXXXS----



P: good; S: marginal; X: bad

FT goodness: higher is better

Intermezzo: some practicalities...



LBCS position search

Results for co-ordinates 140.0, 45.0

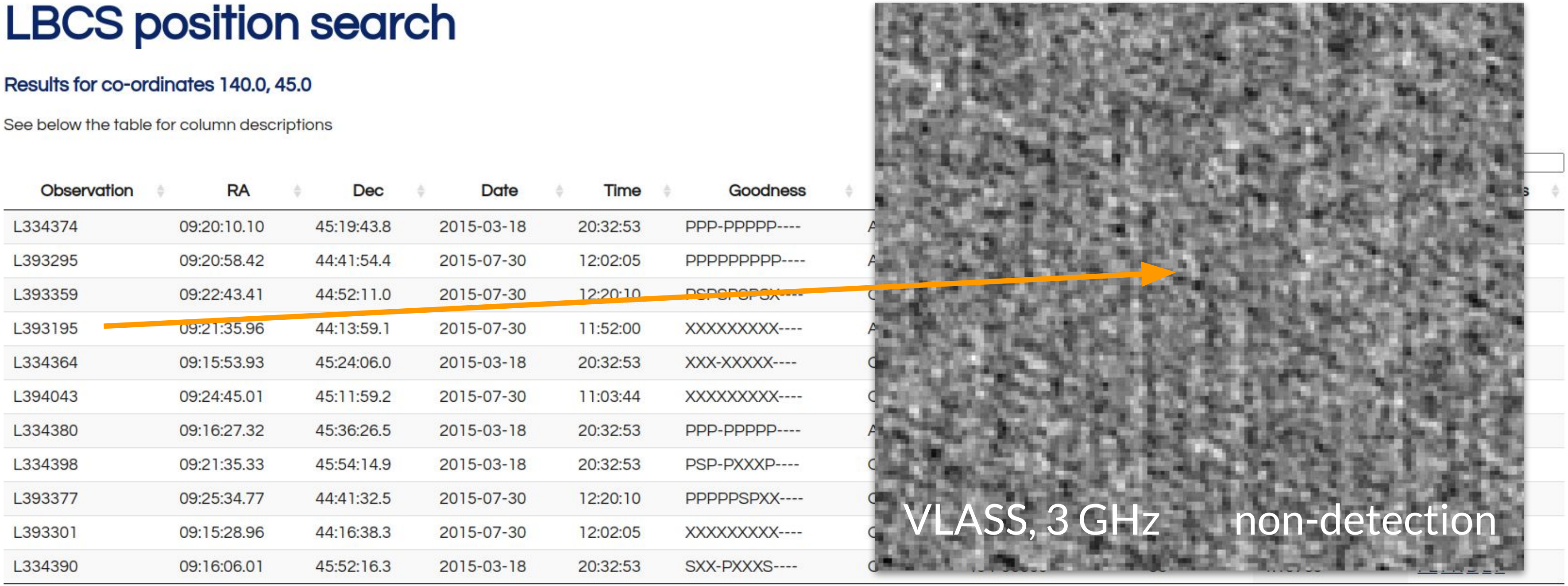
See below the table for column descriptions

Observation	RA	Dec	Date	Time	Goodness
L334374	09:20:10.10	45:19:43.8	2015-03-18	20:32:53	PPP-PPPPP----
L393295	09:20:58.42	44:41:54.4	2015-07-30	12:02:05	PPPPPPPPP----
L393359	09:22:43.41	44:52:11.0	2015-07-30	12:20:10	PSPSPSPSX----
L393195	09:21:35.96	44:13:59.1	2015-07-30	11:52:00	XXXXXXXXX----
L334364	09:15:53.93	45:24:06.0	2015-03-18	20:32:53	XXX-XXXXX----
L394043	09:24:45.01	45:11:59.2	2015-07-30	11:03:44	XXXXXXXXX----
L334380	09:16:27.32	45:36:26.5	2015-03-18	20:32:53	PPP-PPPPP----
L334398	09:21:35.33	45:54:14.9	2015-03-18	20:32:53	PSP-PXXXP----
L393377	09:25:34.77	44:41:32.5	2015-07-30	12:20:10	PPPPPSPXX----
L393301	09:15:28.96	44:16:38.3	2015-07-30	12:02:05	XXXXXXXXX----
L334390	09:16:06.01	45:52:16.3	2015-03-18	20:32:53	SXX-PXXXS----

P: good; S: marginal; X: bad

FT goodness: higher is better

Intermezzo: some practicalities...



LBCS position search

Results for co-ordinates 140.0, 45.0

See below the table for column descriptions

Observation	RA	Dec	Date	Time	Goodness
L334374	09:20:10.10	45:19:43.8	2015-03-18	20:32:53	PPP-PPPPP----
L393295	09:20:58.42	44:41:54.4	2015-07-30	12:02:05	PPPPPPPPP----
L393359	09:22:43.41	44:52:11.0	2015-07-30	12:20:10	PSPSPSPSX----
L393195	09:21:35.96	44:13:59.1	2015-07-30	11:52:00	XXXXXXXXX----
L334364	09:15:53.93	45:24:06.0	2015-03-18	20:32:53	XXX-XXXXX----
L394043	09:24:45.01	45:11:59.2	2015-07-30	11:03:44	XXXXXXXXX----
L334380	09:16:27.32	45:36:26.5	2015-03-18	20:32:53	PPP-PPPPP----
L334398	09:21:35.33	45:54:14.9	2015-03-18	20:32:53	PSP-PXXXP----
L393377	09:25:34.77	44:41:32.5	2015-07-30	12:20:10	PPPPPSPXX----
L393301	09:15:28.96	44:16:38.3	2015-07-30	12:02:05	XXXXXXXXX----
L334390	09:16:06.01	45:52:16.3	2015-03-18	20:32:53	SXX-PXXXS----

P: good; S: marginal; X: bad

FT goodness: higher is better

The in-field calibrator

Performing the calibration consists of a few ingredients:

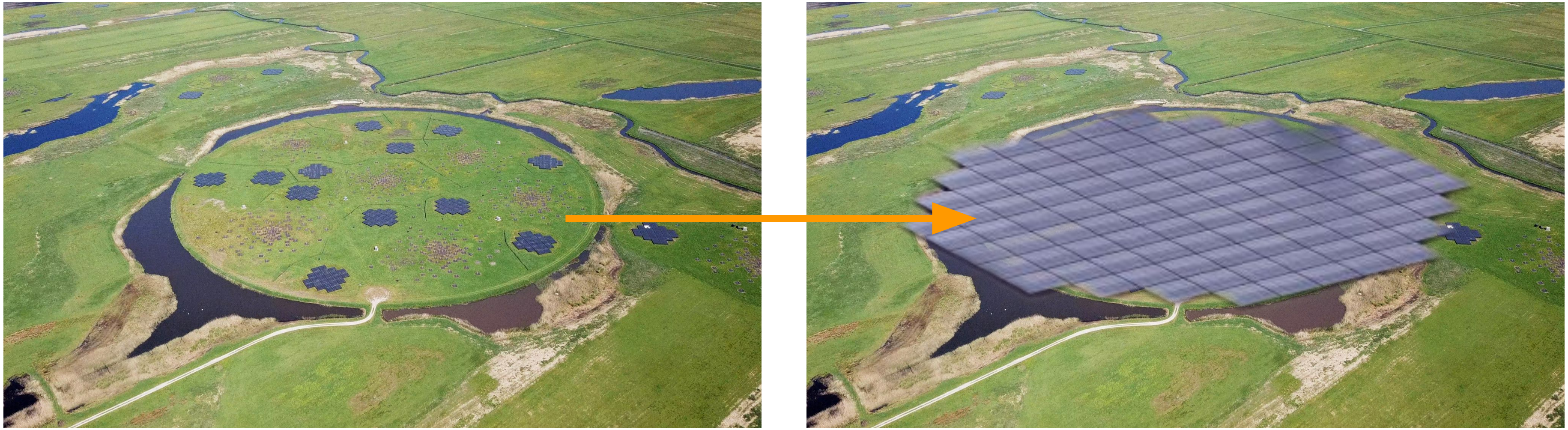
- **Reducing** the **interference** from nearby sources as much as possible
    - Clean fields are prefered
    - Narrow down the FoV
- Providing a **good starting model**
    - Point source
    - VLASS model (in facetselfcal)
    - Custom model
- Configuring the **best calibration steps**
    - Not too much, not too little…

Note that your target could also be the in-field!

The in-field calibrator

Reducing the interference from nearby sources:

- Coherently phase up the core stations to form a single superstation:

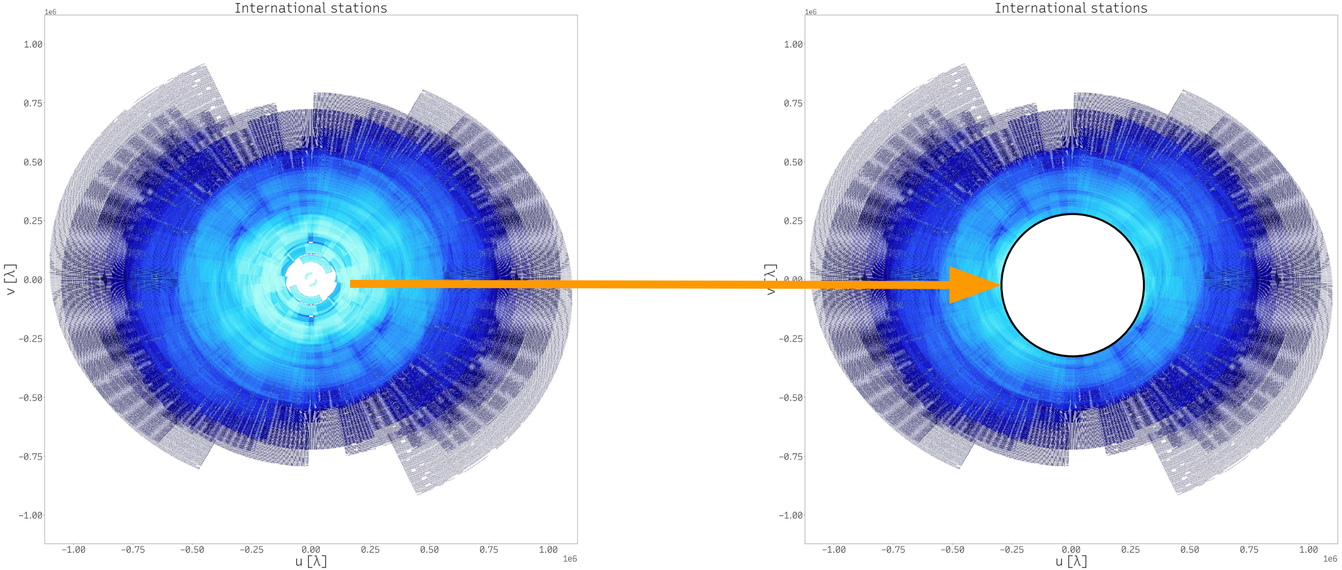


*actually not just the **superterp** but all of the **core** stations

The in-field calibrator

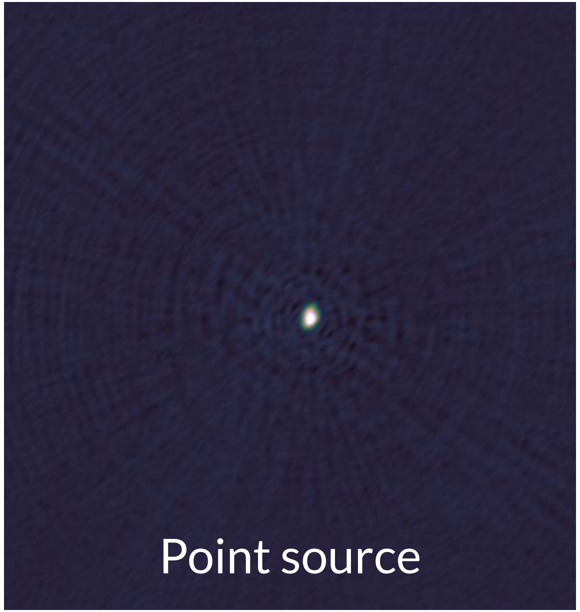
Reducing the interference from nearby sources::

- Coherently phase up the core stations to form a single superstation:
- Performing additional averaging and apply a uv-cut to suppress environment

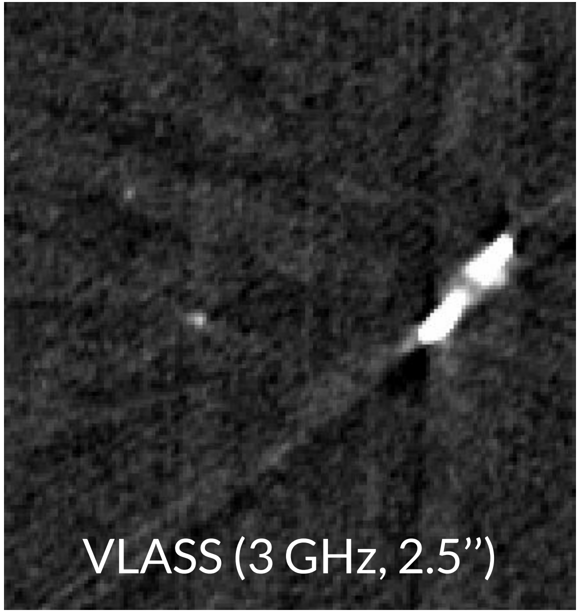


The in-field calibrator

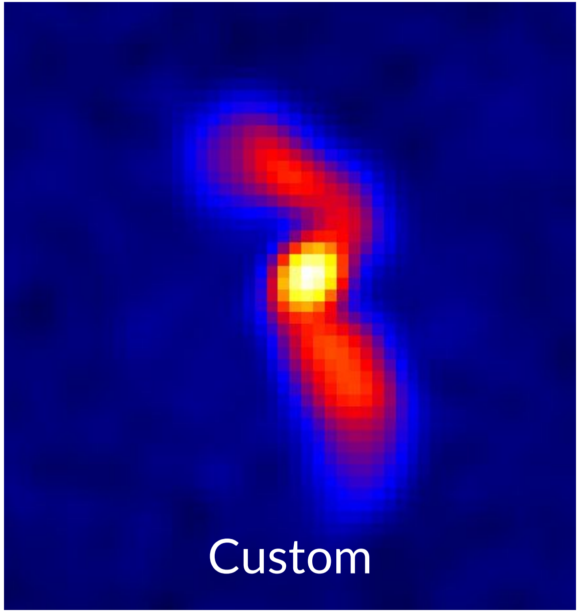
Providing a good starting model:



Point source



VLASS (3 GHz, 2.5”)

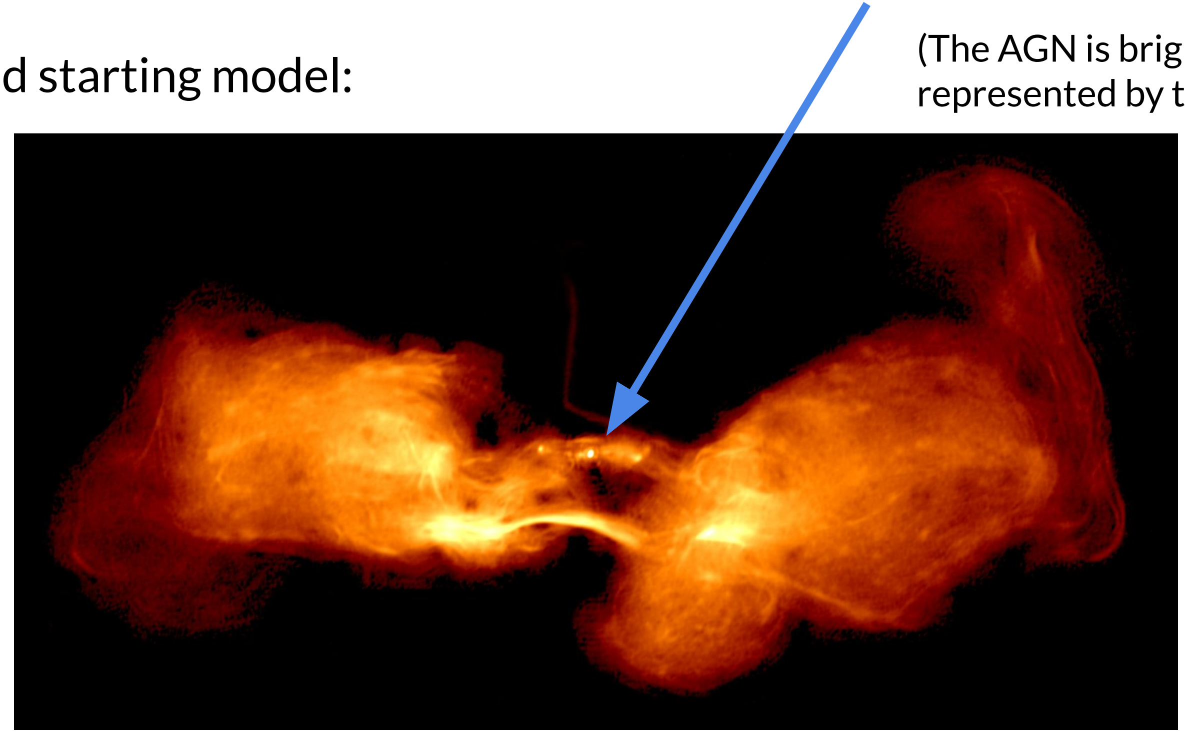


Custom

The point source model is often more successful than you might expect

The in-field calibrator

Providing a good starting model:



There's the AGN = point source

(The AGN is brighter than represented by this color map…)

The point source model is often more successful than you might expect

The in-field calibrator

Configuring the right calibration steps:

The strategy needs to be motivated by what **physically affects the data!**
Real but messy > Fake but pretty

The standard recipe

Nr	Step	Time interval	Freq. interval
1	scalarphasediff	Minutes	>10 MHz
2	scalarphase (Int'l stations)	1 integration	~2 MHz
3	scalarphase (Dutch stations)	1-4 integrations	>10 MHz
4	scalarcomplexgain	~hours	>10 MHz

*with the complexgain step typically starting only after a few cycles

Tutorial prep: facetselfcal overview

How to actually run the script:

```
python3 facetselfcal.py --imsize=1024 -i selfcal_run0 --pixelscale=0.075
--robust=-0.5 --skymodelpointsource=1 --uvmin=40000
--soltype-list="['scalarphasediff','scalarphase','scalarphase','scalarcom
plexgain']" --soltypecycles-list="[0,0,0,2]"
--solint-list="['2min','32s','32s','1h']" --nchan-list="[1,1,1,1]"
--smoothnessconstraint-list="[20.0,1.0,10.0,20.0]" --docircular
--antennaconstraint-list="['alldutch',None,None,None]"
--resetsols-list="[None,'alldutch',None,None]" --stop=12 --avgfreqstep=4
--avgtimestep=4 --phaseupstations='core' your_dataset.ms
```

Tutorial prep: facetselfcal overview

How to actually run the script:

```
python3 facetselfcal.py --imsize=1024 -i selfcal_run0 --pixelscale=0.075
--robust=-0.5 --skymodelpointsource=1 --uvmin=40000
--soltype-list="['scalarphasediff','scalarphase','scalarphase','scalarcom
plexgain']" --soltypecycles-list="[0,0,0,2]"
--solint-list="['2min','32s','32s','1h']" --nchan-list="[1,1,1,1]"
--smoothnessconstraint-list="[20.0,1.0,10.0,20.0]" --docircular
--antennaconstraint-list="['alldutch',None,None,None]"
--resetsols-list="[None,'alldutch',None,None]" --stop=12 --avgfreqstep=4
--avgtimestep=4 --phaseupstations='core' your_dataset.ms
```

Image size

Tutorial prep: facetselfcal overview

How to actually run the script:

```
python3 facetselfcal.py --imsize=1024 -i selfcal_run0 --pixelscale=0.075
--robust=-0.5 --skymodelpointsource=1 --uvmin=40000
--soltype-list="['scalarphasediff','scalarphase','scalarphase','scalarcom
plexgain']" --soltypecycles-list="[0,0,0,2]"
--solint-list="['2min','32s','32s','1h']" --nchan-list="[1,1,1,1]"
--smoothnessconstraint-list="[20.0,1.0,10.0,20.0]" --docircular
--antennaconstraint-list="['alldutch',None,None,None]"
--resetsols-list="[None,'alldutch',None,None]" --stop=12 --avgfreqstep=4
--avgtimestep=4 --phaseupstations='core' your_dataset.ms
```

Image name

Tutorial prep: facetselfcal overview

How to actually run the script:

```
python3 facetselfcal.py --imsize=1024 -i selfcal_run0 --pixelscale=0.075
--robust=-0.5 --skymodelpointsource=1 --uvmin=40000
--soltype-list="['scalarphasediff','scalarphase','scalarphase','scalarcom
plexgain']" --soltypecycles-list="[0,0,0,2]"
--solint-list="['2min','32s','32s','1h']" --nchan-list="[1,1,1,1]"
--smoothnessconstraint-list="[20.0,1.0,10.0,20.0]" --docircular
--antennaconstraint-list="['alldutch',None,None,None]"
--resetsols-list="[None,'alldutch',None,None]" --stop=12 --avgfreqstep=4
--avgtimestep=4 --phaseupstations='core' your_dataset.ms
```

Pixel scale (in arcseconds)

Tutorial prep: facetselfcal overview

How to actually run the script:

```
python3 facetselfcal.py --imsize=1024 -i selfcal_run0 --pixelscale=0.075
--robust=-0.5 --skymodelpointsource=1 --uvmin=40000
--soltype-list="['scalarphasediff','scalarphase','scalarphase','scalarcom
plexgain']" --soltypecycles-list="[0,0,0,2]"
--solint-list="['2min','32s','32s','1h']" --nchan-list="[1,1,1,1]"
--smoothnessconstraint-list="[20.0,1.0,10.0,20.0]" --docircular
--antennaconstraint-list="['alldutch',None,None,None]"
--resetsols-list="[None,'alldutch',None,None]" --stop=12 --avgfreqstep=4
--avgtimestep=4 --phaseupstations='core' your_dataset.ms
```

uv weighting scheme

Tutorial prep: facetselfcal overview

How to actually run the script:

```
python3 facetselfcal.py --imsize=1024 -i selfcal_run0 --pixelscale=0.075
--robust=-0.5 --skymodelpointsource=1 --uvmin=40000
--soltype-list="['scalarphasediff','scalarphase','scalarphase','scalarcom
plexgain']" --soltypecycles-list="[0,0,0,2]"
--solint-list="['2min','32s','32s','1h']" --nchan-list="[1,1,1,1]"
--smoothnessconstraint-list="[20.0,1.0,10.0,20.0]" --docircular
--antennaconstraint-list="['alldutch',None,None,None]"
--resetsols-list="[None,'alldutch',None,None]" --stop=12 --avgfreqstep=4
--avgtimestep=4 --phaseupstations='core' your_dataset.ms
```

Starting sky model

Tutorial prep: facetselfcal overview

How to actually run the script:

```
python3 facetselfcal.py --imsize=1024 -i selfcal_run0 --pixelscale=0.075
--robust=-0.5 --skymodelpointsource=1 --uvmin=40000
--soltype-list="['scalarphasediff','scalarphase','scalarphase','scalarcom
plexgain']" --soltypecycles-list="[0,0,0,2]"
--solint-list="['2min','32s','32s','1h']" --nchan-list="[1,1,1,1]"
--smoothnessconstraint-list="[20.0,1.0,10.0,20.0]" --docircular
--antennaconstraint-list="['alldutch',None,None,None]"
--resetsols-list="[None,'alldutch',None,None]" --stop=12 --avgfreqstep=4
--avgtimestep=4 --phaseupstations='core' your_dataset.ms
```

Minimum baseline length (in λ)

Tutorial prep: facetselfcal overview

How to actually run the script:

```
python3 facetselfcal.py --imsize=1024 -i selfcal_run0 --pixelscale=0.075
--robust=-0.5 --skymodelpointsource=1 --uvmin=40000
--soltype-list="['scalarphasediff','scalarphase','scalarphase','scalarcom
plexgain']" --soltypecycles-list="[0,0,0,2]"
--solint-list="['2min','32s','32s','1h']" --nchan-list="[1,1,1,1]"
--smoothnessconstraint-list="[20.0,1.0,10.0,20.0]" --docircular
--antennaconstraint-list="['alldutch',None,None,None]"
--resetsols-list="[None,'alldutch',None,None]" --stop=12 --avgfreqstep=4
--avgtimestep=4 --phaseupstations='core' your_dataset.ms
```

Steps per self-cal cycle

Tutorial prep: facetselfcal overview

How to actually run the script:

```
python3 facetselfcal.py --imsize=1024 -i selfcal_run0 --pixelscale=0.075
--robust=-0.5 --skymodelpointsource=1 --uvmin=40000
--soltype-list="['scalarphasediff','scalarphase','scalarphase','scalarcom
plexgain']" --soltypecycles-list="[0,0,0,2]"
--solint-list="['2min','32s','32s','1h']" --nchan-list="[1,1,1,1]"
--smoothnessconstraint-list="[20.0,1.0,10.0,20.0]" --docircular
--antennaconstraint-list="['alldutch',None,None,None]"
--resetsols-list="[None,'alldutch',None,None]" --stop=12 --avgfreqstep=4
--avgtimestep=4 --phaseupstations='core' your_dataset.ms
```

Which cycle each step is enabled

Tutorial prep: facetselfcal overview

How to actually run the script:

```
python3 facetselfcal.py --imsize=1024 -i selfcal_run0 --pixelscale=0.075
--robust=-0.5 --skymodelpointsource=1 --uvmin=40000
--soltype-list="['scalarphasediff','scalarphase','scalarphase','scalarcom
plexgain']" --soltypecycles-list="[0,0,0,2]"
--solint-list="['2min','32s','32s','1h']" --nchan-list="[1,1,1,1]"
--smoothnessconstraint-list="[20.0,1.0,10.0,20.0]" --docircular
--antennaconstraint-list="['alldutch',None,None,None]"
--resetsols-list="[None,'alldutch',None,None]" --stop=12 --avgfreqstep=4
--avgtimestep=4 --phaseupstations='core' your_dataset.ms
```

Solution interval for each calibration step

Tutorial prep: facetselfcal overview

How to actually run the script:

```
python3 facetselfcal.py --imsize=1024 -i selfcal_run0 --pixelscale=0.075
--robust=-0.5 --skymodelpointsource=1 --uvmin=40000
--soltype-list="['scalarphasediff','scalarphase','scalarphase','scalarcom
plexgain']" --soltypecycles-list="[0,0,0,2]"
--solint-list="['2min','32s','32s','1h']" --nchan-list="[1,1,1,1]"
--smoothnessconstraint-list="[20.0,1.0,10.0,20.0]" --docircular
--antennaconstraint-list="['alldutch',None,None,None]"
--resetsols-list="[None,'alldutch',None,None]" --stop=12 --avgfreqstep=4
--avgtimestep=4 --phaseupstations='core' your_dataset.ms
```

Freq. channel interval for each calibration step

Tutorial prep: facetselfcal overview

How to actually run the script:

```
python3 facetselfcal.py --imsize=1024 -i selfcal_run0 --pixelscale=0.075
--robust=-0.5 --skymodelpointsource=1 --uvmin=40000
--soltype-list="['scalarphasediff','scalarphase','scalarphase','scalarcom
plexgain']" --soltypecycles-list="[0,0,0,2]"
--solint-list="['2min','32s','32s','1h']" --nchan-list="[1,1,1,1]"
--smoothnessconstraint-list="[20.0,1.0,10.0,20.0]" --docircular
--antennaconstraint-list="['alldutch',None,None,None]"
--resetsols-list="[None,'alldutch',None,None]" --stop=12 --avgfreqstep=4
--avgtimestep=4 --phaseupstations='core' your_dataset.ms
```

Frequency smoothing of solutions per step

Tutorial prep: facetselfcal overview

How to actually run the script:

```
python3 facetselfcal.py --imsize=1024 -i selfcal_run0 --pixelscale=0.075
--robust=-0.5 --skymodelpointsource=1 --uvmin=40000
--soltype-list="['scalarphasediff','scalarphase','scalarphase','scalarcom
plexgain']" --soltypecycles-list="[0,0,0,2]"
--solint-list="['2min','32s','32s','1h']" --nchan-list="[1,1,1,1]"
--smoothnessconstraint-list="[20.0,1.0,10.0,20.0]" --docircular
--antennaconstraint-list="['alldutch',None,None,None]"
--resetsols-list="[None,'alldutch',None,None]" --stop=12 --avgfreqstep=4
--avgtimestep=4 --phaseupstations='core' your_dataset.ms
```

To convert linear to circular polarization

Tutorial prep: facetselfcal overview

How to actually run the script:

```
python3 facetselfcal.py --imsize=1024 -i selfcal_run0 --pixelscale=0.075
--robust=-0.5 --skymodelpointsource=1 --uvmin=40000
--soltype-list="['scalarphasediff','scalarphase','scalarphase','scalarcom
plexgain']" --soltypecycles-list="[0,0,0,2]"
--solint-list="['2min','32s','32s','1h']" --nchan-list="[1,1,1,1]"
--smoothnessconstraint-list="[20.0,1.0,10.0,20.0]" --docircular
--antennaconstraint-list="['alldutch',None,None,None]"
--resetsols-list="[None,'alldutch',None,None]" --stop=12 --avgfreqstep=4
--avgtimestep=4 --phaseupstations='core' your_dataset.ms
```

Antenna constraints to apply to each step

Tutorial prep: facetselfcal overview

How to actually run the script:

```
python3 facetselfcal.py --imsize=1024 -i selfcal_run0 --pixelscale=0.075
--robust=-0.5 --skymodelpointsource=1 --uvmin=40000
--soltype-list="['scalarphasediff','scalarphase','scalarphase','scalarcom
plexgain']" --soltypecycles-list="[0,0,0,2]"
--solint-list="['2min','32s','32s','1h']" --nchan-list="[1,1,1,1]"
--smoothnessconstraint-list="[20.0,1.0,10.0,20.0]" --docircular
--antennaconstraint-list="['alldutch',None,None,None]"
--resetsols-list="[None,'alldutch',None,None]" --stop=12 --avgfreqstep=4
--avgtimestep=4 --phaseupstations='core' your_dataset.ms
```

Which solutions to reset after derivation

Tutorial prep: facetselfcal overview

How to actually run the script:

```
python3 facetselfcal.py --imsize=1024 -i selfcal_run0 --pixelscale=0.075
--robust=-0.5 --skymodelpointsource=1 --uvmin=40000
--soltype-list="['scalarphasediff','scalarphase','scalarphase','scalarcom
plexgain']" --soltypecycles-list="[0,0,0,2]"
--solint-list="['2min','32s','32s','1h']" --nchan-list="[1,1,1,1]"
--smoothnessconstraint-list="[20.0,1.0,10.0,20.0]" --docircular
--antennaconstraint-list="['alldutch',None,None,None]"
--resetsols-list="[None,'alldutch',None,None]" --stop=12 --avgfreqstep=4
--avgtimestep=4 --phaseupstations='core' your_dataset.ms
```

After how many cycles to stop

Tutorial prep: facetselfcal overview

How to actually run the script:

```
python3 facetselfcal.py --imsize=1024 -i selfcal_run0 --pixelscale=0.075
--robust=-0.5 --skymodelpointsource=1 --uvmin=40000
--soltype-list="['scalarphasediff','scalarphase','scalarphase','scalarcom
plexgain']" --soltypecycles-list="[0,0,0,2]"
--solint-list="['2min','32s','32s','1h']" --nchan-list="[1,1,1,1]"
--smoothnessconstraint-list="[20.0,1.0,10.0,20.0]" --docircular
--antennaconstraint-list="['alldutch',None,None,None]"
--resetsols-list="[None,'alldutch',None,None]" --stop=12 --avgfreqstep=4
--avgtimestep=4 --phaseupstations='core' your_dataset.ms
```

How much freq. averaging to perform

Tutorial prep: facetselfcal overview

How to actually run the script:

```
python3 facetselfcal.py --imsize=1024 -i selfcal_run0 --pixelscale=0.075
--robust=-0.5 --skymodelpointsource=1 --uvmin=40000
--soltype-list="['scalarphasediff','scalarphase','scalarphase','scalarcom
plexgain']" --soltypecycles-list="[0,0,0,2]"
--solint-list="['2min','32s','32s','1h']" --nchan-list="[1,1,1,1]"
--smoothnessconstraint-list="[20.0,1.0,10.0,20.0]" --docircular
--antennaconstraint-list="['alldutch',None,None,None]"
--resetsols-list="[None,'alldutch',None,None]" --stop=12 --avgfreqstep=4
--avgtimestep=4 --phaseupstations='core' your_dataset.ms
```

How much time averaging to perform

Tutorial prep: facetselfcal overview

How to actually run the script:

```
python3 facetselfcal.py --imsize=1024 -i selfcal_run0 --pixelscale=0.075
--robust=-0.5 --skymodelpointsource=1 --uvmin=40000
--soltype-list="['scalarphasediff','scalarphase','scalarphase','scalarcomplexgain']" --soltypecycles-list="[0,0,0,2]"
--solint-list="['2min','32s','32s','1h']" --nchan-list="[1,1,1,1]"
--smoothnessconstraint-list="[20.0,1.0,10.0,20.0]" --docircular
--antennaconstraint-list="['alldutch',None,None,None]"
--resetsols-list="[None,'alldutch',None,None]" --stop=12 --avgfreqstep=4
--avgtimestep=4 --phaseupstations='core' your_dataset.ms
```

To phase-up a set of antennas

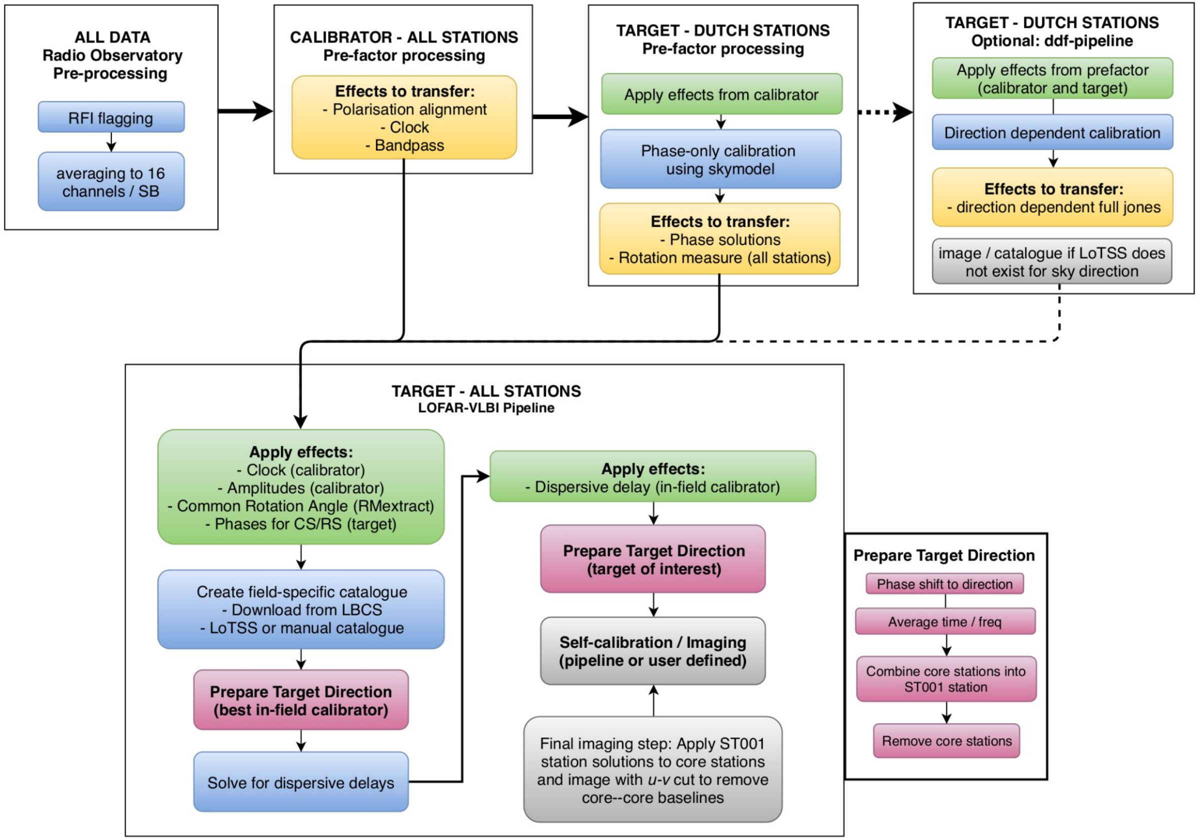
Tutorial prep: facetselfcal overview

How to actually run the script:

```
python3 facetselfcal.py --imsize=1024 -i selfcal_run0 --pixelscale=0.075
--robust=-0.5 --skymodelpointsource=1 --uvmin=40000
--soltype-list="['scalarphasediff','scalarphase','scalarphase','scalarcom
plexgain']" --soltypecycles-list="[0,0,0,2]"
--solint-list="['2min','32s','32s','1h']" --nchan-list="[1,1,1,1]"
--smoothnessconstraint-list="[20.0,1.0,10.0,20.0]" --docircular
--antennaconstraint-list="['alldutch',None,None,None]"
--resetsols-list="[None,'alldutch',None,None]" --stop=12 --avgfreqstep=4
--avgtimestep=4 --phaseupstations='core' your_dataset.ms
```

Your Measurement Set

The LOFAR-VLBI pipeline



ALL DATA
Radio Observatory
Pre-processing

RFI flagging

averaging to 16 channels / SB

CALIBRATOR - ALL STATIONS
Pre-factor processing

Effects to transfer:
- Polarisation alignment
- Clock
- Bandpass

TARGET - DUTCH STATIONS
Pre-factor processing

Apply effects from calibrator

Phase-only calibration using skymodel

Effects to transfer:
- Phase solutions
- Rotation measure (all stations)

TARGET - DUTCH STATIONS
Optional: ddf-pipeline

Apply effects from prefactor (calibrator and target)

Direction dependent calibration

Effects to transfer:
- direction dependent full jones

image / catalogue if LoTSS does not exist for sky direction

TARGET - ALL STATIONS
LOFAR-VLBI Pipeline

Apply effects:
- Clock (calibrator)
- Amplitudes (calibrator)
- Common Rotation Angle (RMextract)
- Phases for CS/RS (target)

Create field-specific catalogue
- Download from LBCS
- LoTSS or manual catalogue

Prepare Target Direction (best in-field calibrator)

Solve for dispersive delays

Apply effects:
- Dispersive delay (in-field calibrator)

Prepare Target Direction (target of interest)

Self-calibration / Imaging (pipeline or user defined)

Final imaging step: Apply ST001 station solutions to core stations and image with *u-v* cut to remove core--core baselines

Prepare Target Direction

Phase shift to direction

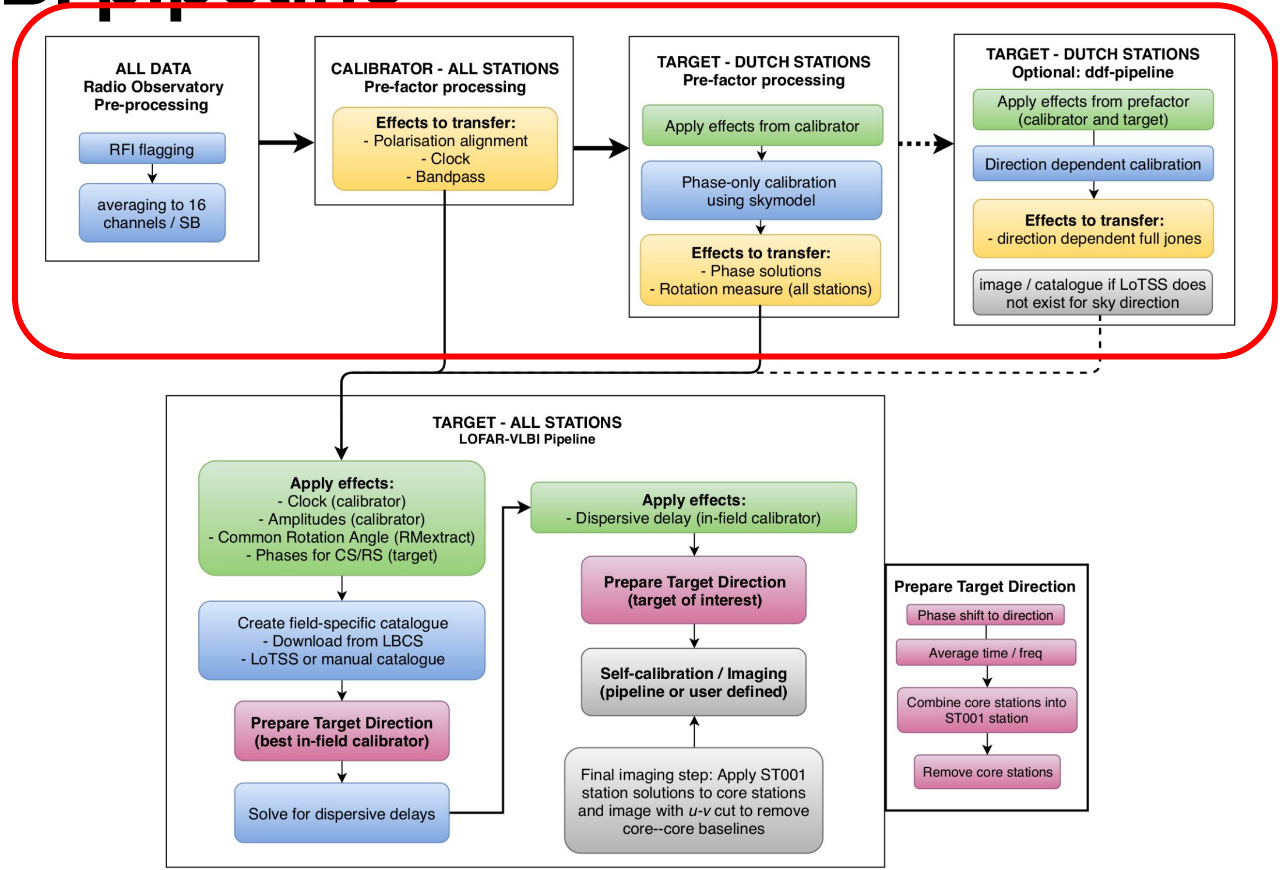
Average time / freq

Combine core stations into ST001 station

Remove core stations

The LOFAR-VLBI pipeline

Standard Dutch
array processing



ALL DATA
Radio Observatory
Pre-processing

RFI flagging

averaging to 16
channels / SB

CALIBRATOR - ALL STATIONS
Pre-factor processing

Effects to transfer:
- Polarisation alignment
- Clock
- Bandpass

TARGET - DUTCH STATIONS
Pre-factor processing

Apply effects from calibrator

Phase-only calibration
using skymodel

Effects to transfer:
- Phase solutions
- Rotation measure (all stations)

TARGET - DUTCH STATIONS
Optional: ddf-pipeline

Apply effects from prefactor
(calibrator and target)

Direction dependent calibration

Effects to transfer:
- direction dependent full jones

image / catalogue if LoTSS does
not exist for sky direction

TARGET - ALL STATIONS
LOFAR-VLBI Pipeline

Apply effects:
- Clock (calibrator)
- Amplitudes (calibrator)
- Common Rotation Angle (RMextract)
- Phases for CS/RS (target)

Create field-specific catalogue
- Download from LBCS
- LoTSS or manual catalogue

Prepare Target Direction
(best in-field calibrator)

Solve for dispersive delays

Apply effects:
- Dispersive delay (in-field calibrator)

Prepare Target Direction
(target of interest)

Self-calibration / Imaging
(pipeline or user defined)

Final imaging step: Apply ST001
station solutions to core stations
and image with *u-v* cut to remove
core--core baselines

Prepare Target Direction

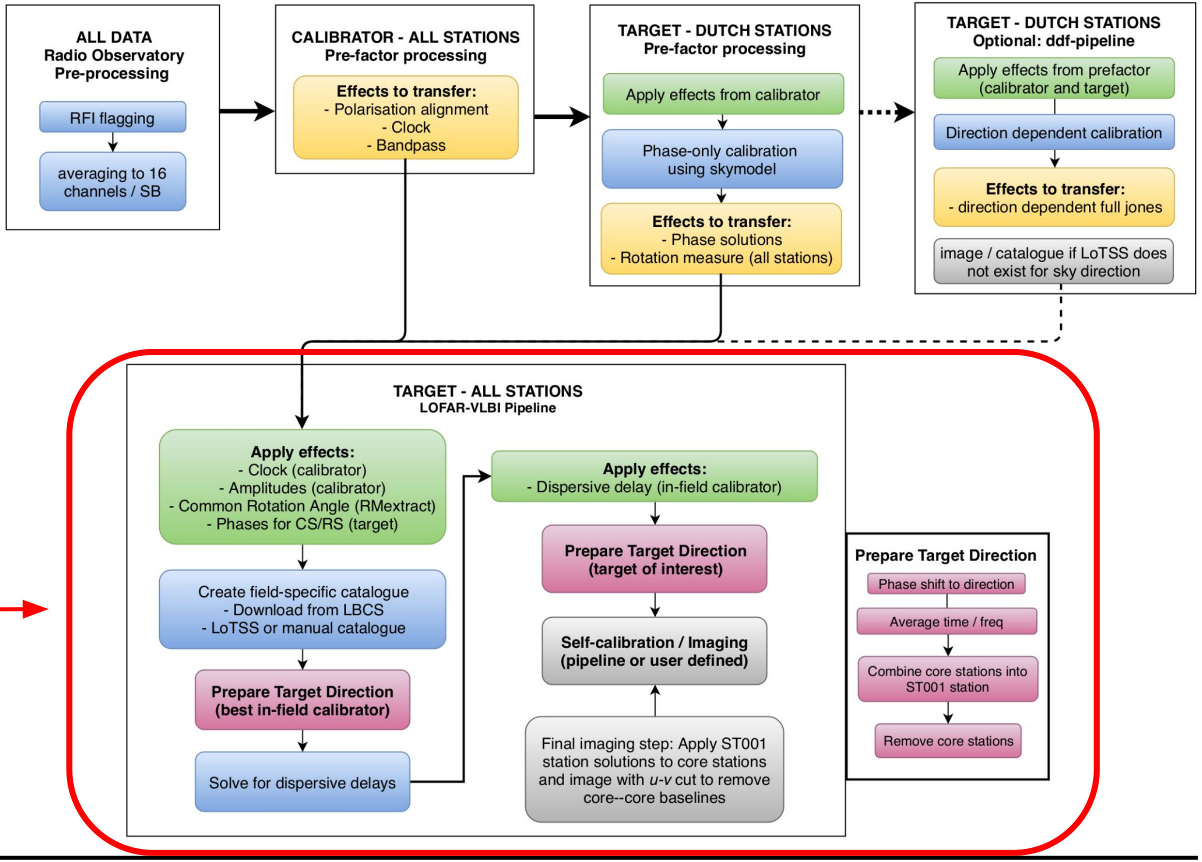
Phase shift to direction

Average time / freq

Combine core stations into
ST001 station

Remove core stations

The LOFAR-VLBI pipeline



ALL DATA
Radio Observatory
Pre-processing

RFI flagging

averaging to 16 channels / SB

CALIBRATOR - ALL STATIONS
Pre-factor processing

Effects to transfer:
- Polarisation alignment
- Clock
- Bandpass

TARGET - DUTCH STATIONS
Pre-factor processing

Apply effects from calibrator

Phase-only calibration using skymodel

Effects to transfer:
- Phase solutions
- Rotation measure (all stations)

TARGET - DUTCH STATIONS
Optional: ddf-pipeline

Apply effects from prefactor (calibrator and target)

Direction dependent calibration

Effects to transfer:
- direction dependent full jones

image / catalogue if LoTSS does not exist for sky direction

TARGET - ALL STATIONS
LOFAR-VLBI Pipeline

Apply effects:
- Clock (calibrator)
- Amplitudes (calibrator)
- Common Rotation Angle (RMextract)
- Phases for CS/RS (target)

Create field-specific catalogue
- Download from LBCS
- LoTSS or manual catalogue

Prepare Target Direction (best in-field calibrator)

Solve for dispersive delays

Apply effects:
- Dispersive delay (in-field calibrator)

Prepare Target Direction (target of interest)

Self-calibration / Imaging (pipeline or user defined)

Final imaging step: Apply ST001 station solutions to core stations and image with *u*-*v* cut to remove core--core baselines

Prepare Target Direction

Phase shift to direction

Average time / freq

Combine core stations into ST001 station

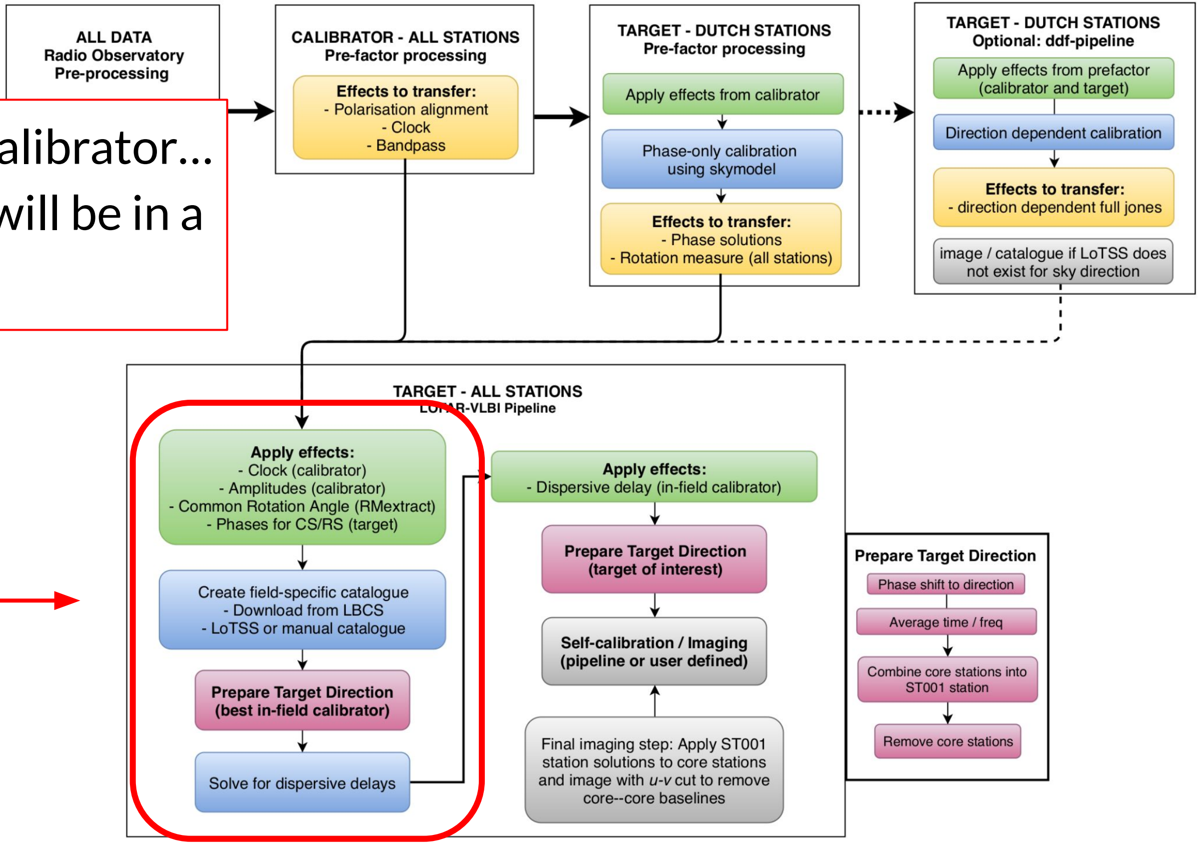
Remove core stations

LOFAR-VLBI Pipeline →

Available on GitHub:
github.com/lofar-vlbi

The LOFAR-VLBI pipeline

Once you have the in-field calibrator… calibrated, most of the FoV will be in a good shape



ALL DATA
Radio Observatory
Pre-processing

CALIBRATOR - ALL STATIONS
Pre-factor processing

Effects to transfer:
- Polarisation alignment
- Clock
- Bandpass

TARGET - DUTCH STATIONS
Pre-factor processing

Apply effects from calibrator

Phase-only calibration using skymodel

Effects to transfer:
- Phase solutions
- Rotation measure (all stations)

TARGET - DUTCH STATIONS
Optional: ddf-pipeline

Apply effects from prefactor (calibrator and target)

Direction dependent calibration

Effects to transfer:
- direction dependent full jones

image / catalogue if LoTSS does not exist for sky direction

TARGET - ALL STATIONS
LOFAR-VLBI Pipeline

Apply effects:
- Clock (calibrator)
- Amplitudes (calibrator)
- Common Rotation Angle (RMextract)
- Phases for CS/RS (target)

Create field-specific catalogue
- Download from LBCS
- LoTSS or manual catalogue

Prepare Target Direction
(best in-field calibrator)

Solve for dispersive delays

Apply effects:
- Dispersive delay (in-field calibrator)

Prepare Target Direction
(target of interest)

Self-calibration / Imaging
(pipeline or user defined)

Final imaging step: Apply ST001 station solutions to core stations and image with *u-v* cut to remove core--core baselines

Prepare Target Direction

Phase shift to direction

Average time / freq

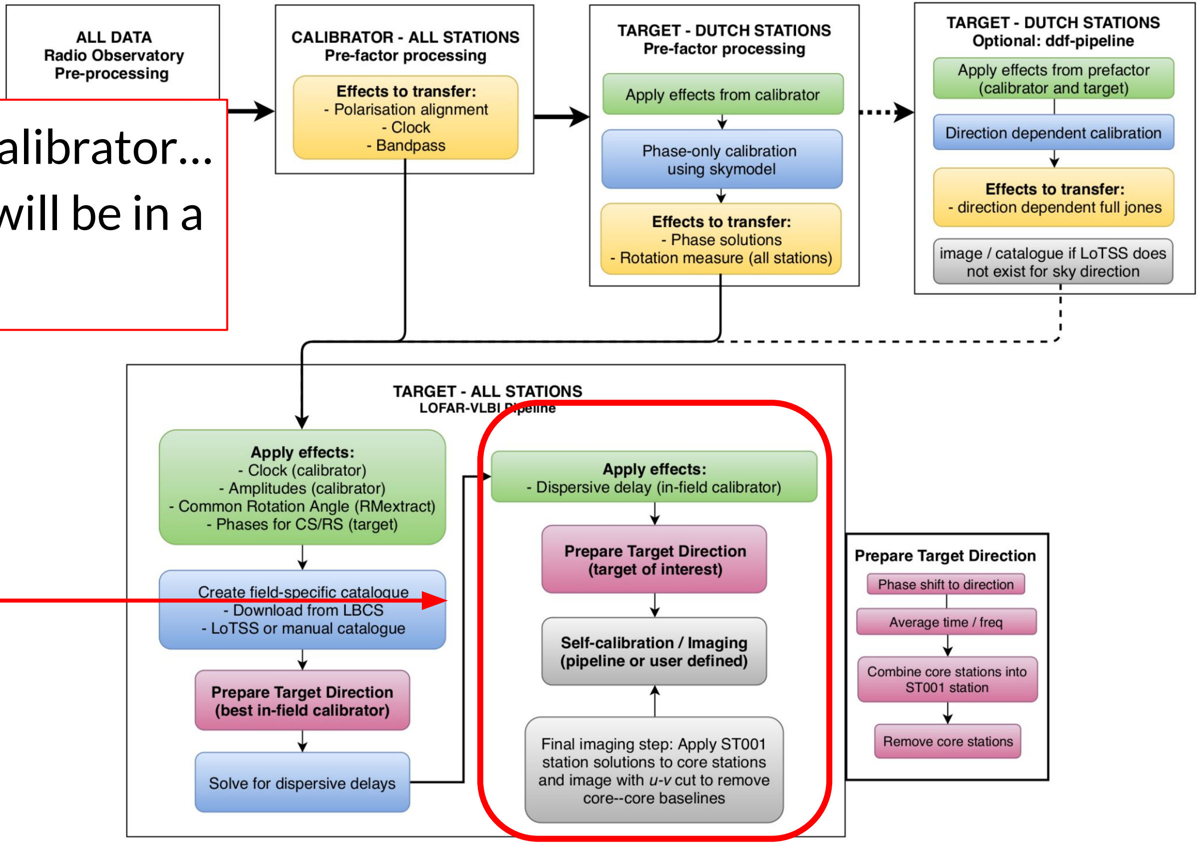
Combine core stations into ST001 station

Remove core stations

Solve for dispersive delays using in-field calibrator source

The LOFAR-VLBI pipeline

Once you have the in-field calibrator… calibrated, most of the FoV will be in a good shape



ALL DATA
Radio Observatory
Pre-processing

CALIBRATOR - ALL STATIONS
Pre-factor processing

Effects to transfer:
- Polarisation alignment
- Clock
- Bandpass

TARGET - DUTCH STATIONS
Pre-factor processing

Apply effects from calibrator

Phase-only calibration using skymodel

Effects to transfer:
- Phase solutions
- Rotation measure (all stations)

TARGET - DUTCH STATIONS
Optional: ddf-pipeline

Apply effects from prefactor (calibrator and target)

Direction dependent calibration

Effects to transfer:
- direction dependent full jones

image / catalogue if LoTSS does not exist for sky direction

TARGET - ALL STATIONS
LOFAR-VLBI Pipeline

Apply effects:
- Clock (calibrator)
- Amplitudes (calibrator)
- Common Rotation Angle (RMextract)
- Phases for CS/RS (target)

Create field-specific catalogue
- Download from LBCS
- LoTSS or manual catalogue

Prepare Target Direction
(best in-field calibrator)

Solve for dispersive delays

Apply effects:
- Dispersive delay (in-field calibrator)

Prepare Target Direction
(target of interest)

Self-calibration / Imaging
(pipeline or user defined)

Final imaging step: Apply ST001 station solutions to core stations and image with *u-v* cut to remove core--core baselines

Prepare Target Direction

Phase shift to direction

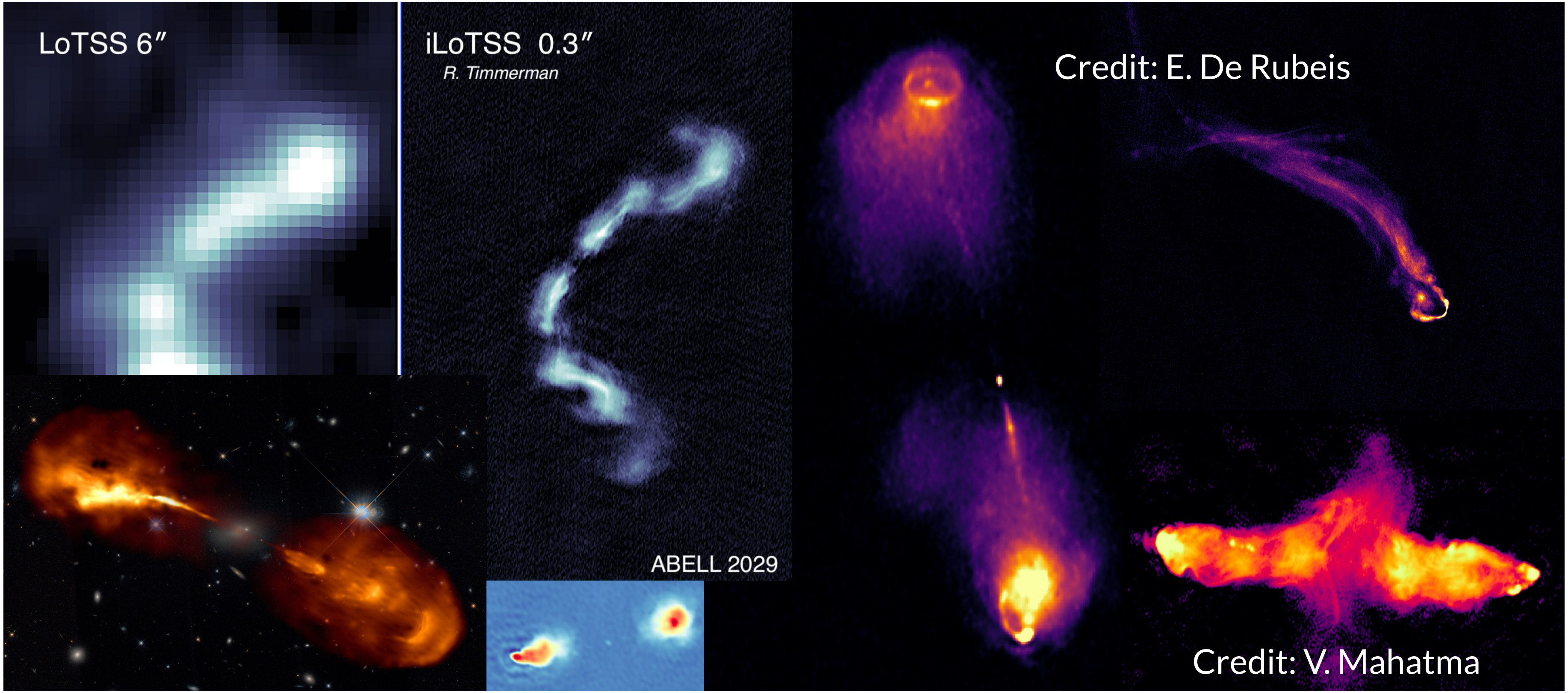
Average time / freq

Combine core stations into ST001 station

Remove core stations

Apply solutions and perform final self-calibration on target

Finally: enjoy your high-resolution images!!



Credit: E. De Rubeis

Credit: V. Mahatma