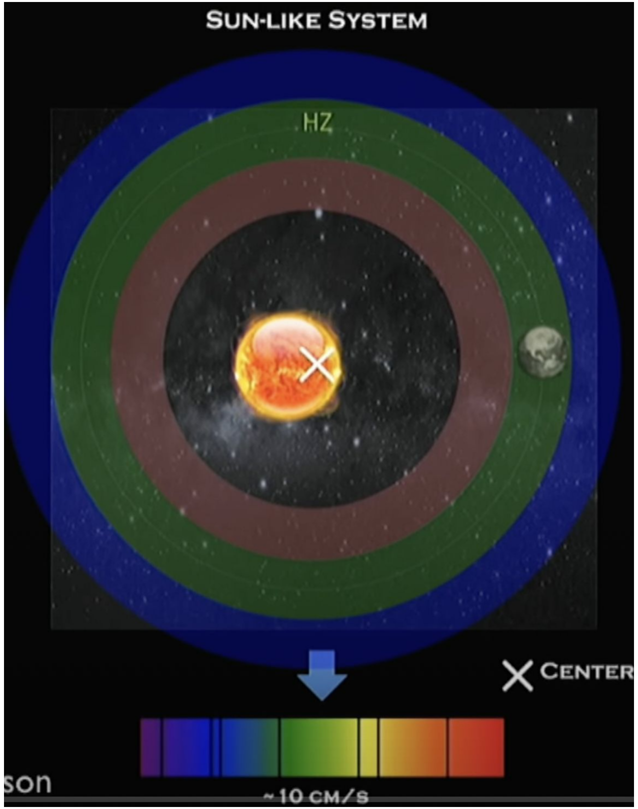
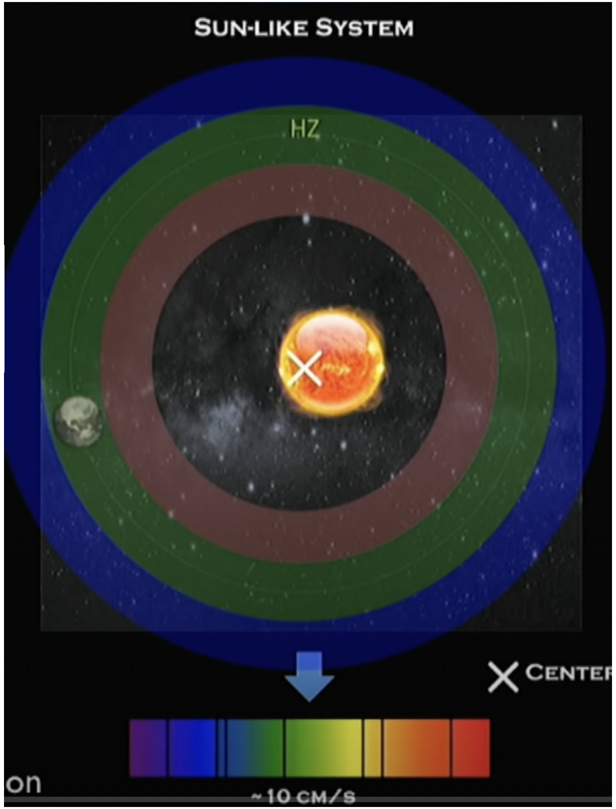
# Velocimetry for exoplanets detection

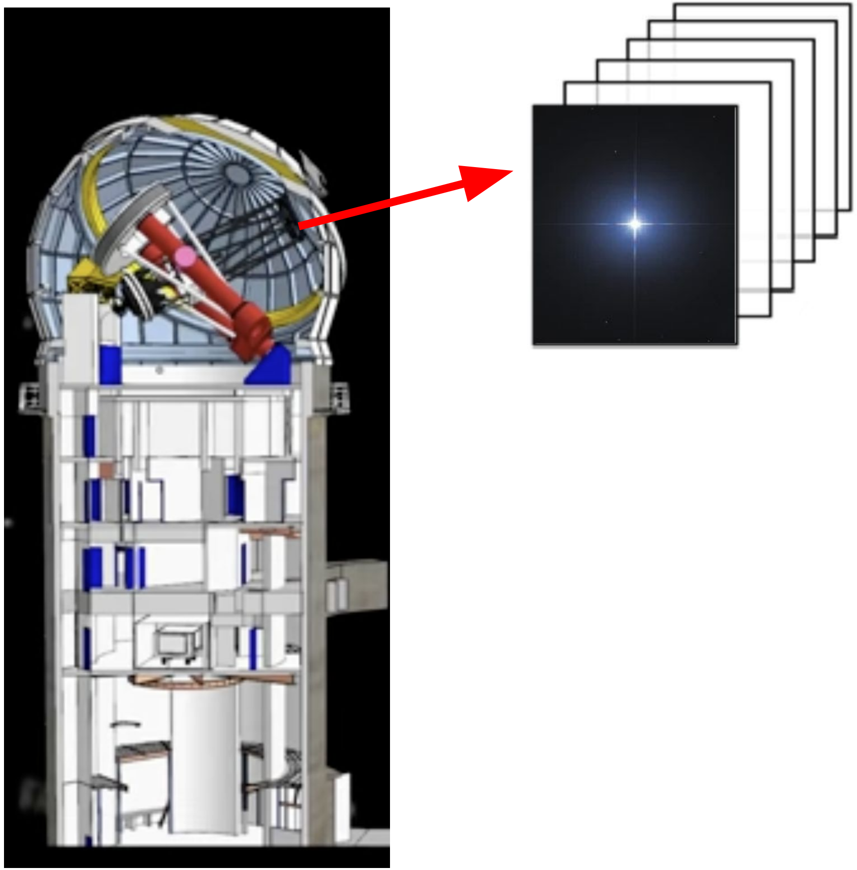
# Velocimetry



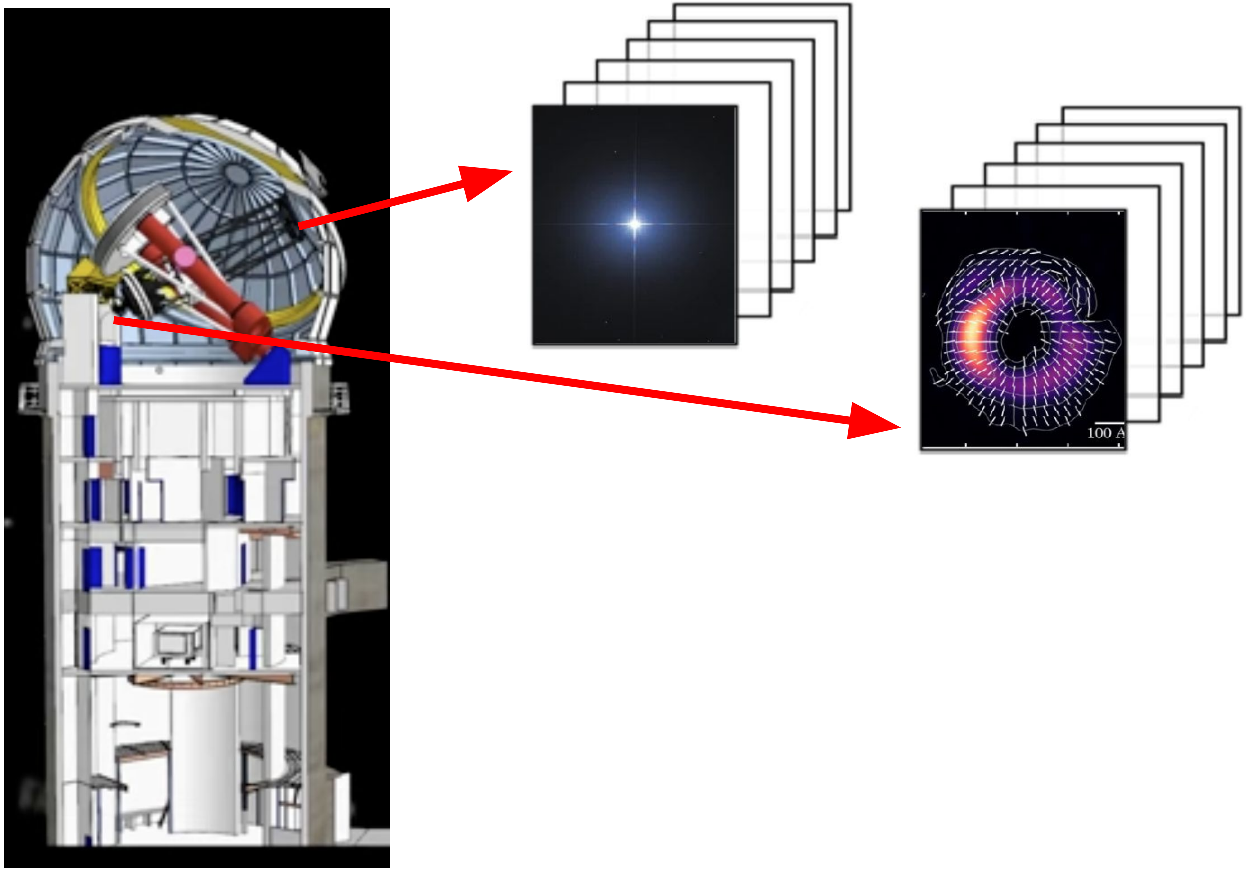

$$K \approx \left(\frac{2\pi G}{PM_*^2}\right)^{\frac{1}{3}} \frac{M_{\text{planet}} \sin i}{\sqrt{1-e^2}}$$



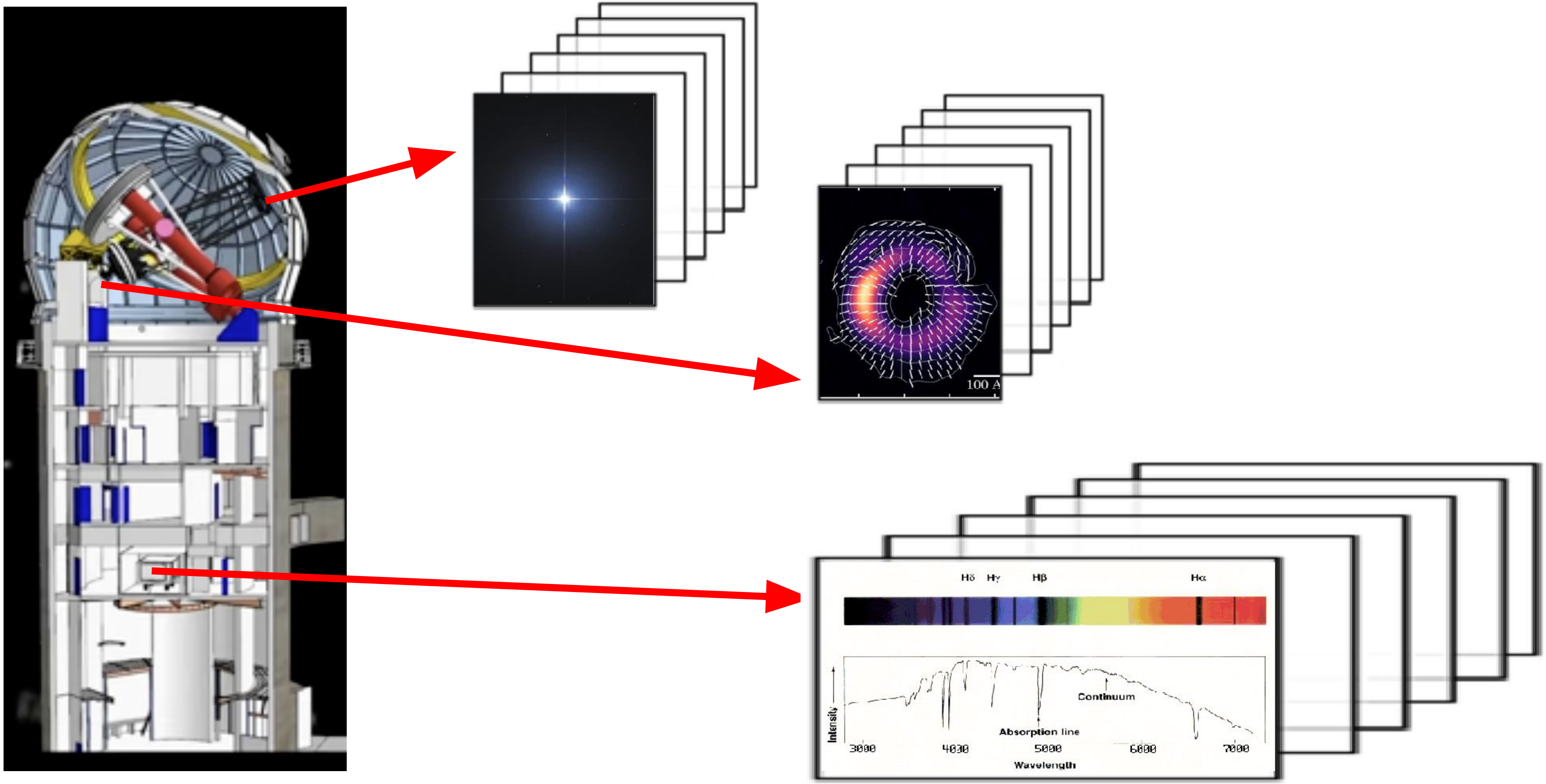
# Observation



# Observation



# Observation



# SERVAL

Flux error

model(obs)

$$\chi^2 = \sum_{n,i} w_{n,i} [f_{n,i} - p(\lambda_{n,i}, a) \cdot F(\lambda'(\lambda_{n,i}, v_n), b)]^2$$

Obs flux

polynomial/vertical

template/shift

Time, orders, pixels

# SERVAL

Flux error

model(obs)

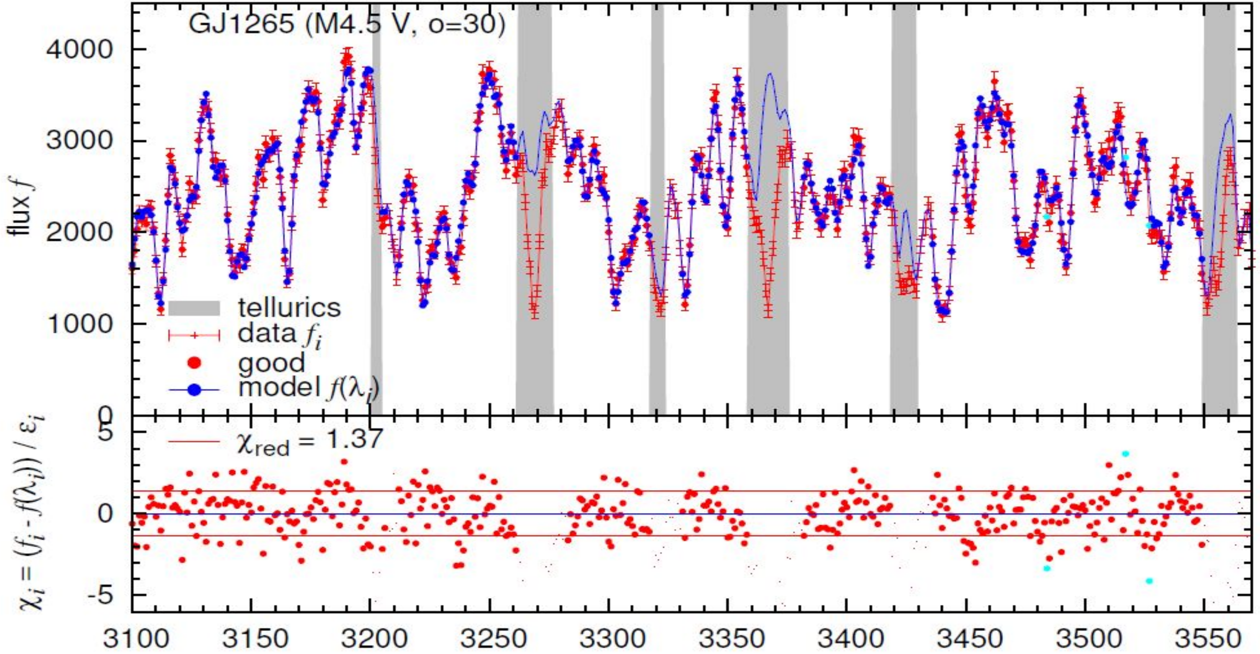
$$\chi^2 = \sum_{n,i} w_{n,i}[f_{n,i} - p(\lambda_{n,i}, a) \cdot F(\lambda'(\lambda_{n,i}, v_n), b)]^2$$

Obs flux

polynomial/vertical

template/shift

Time, orders, pixels



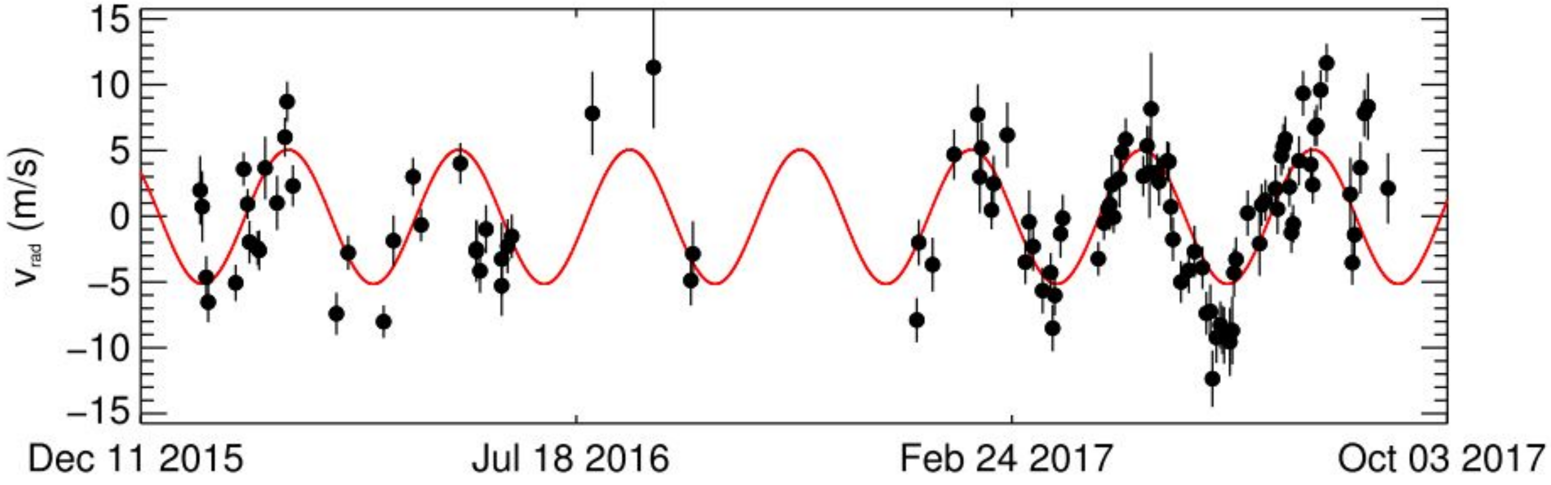
# SERVAL

Flux error                                        model(obs)

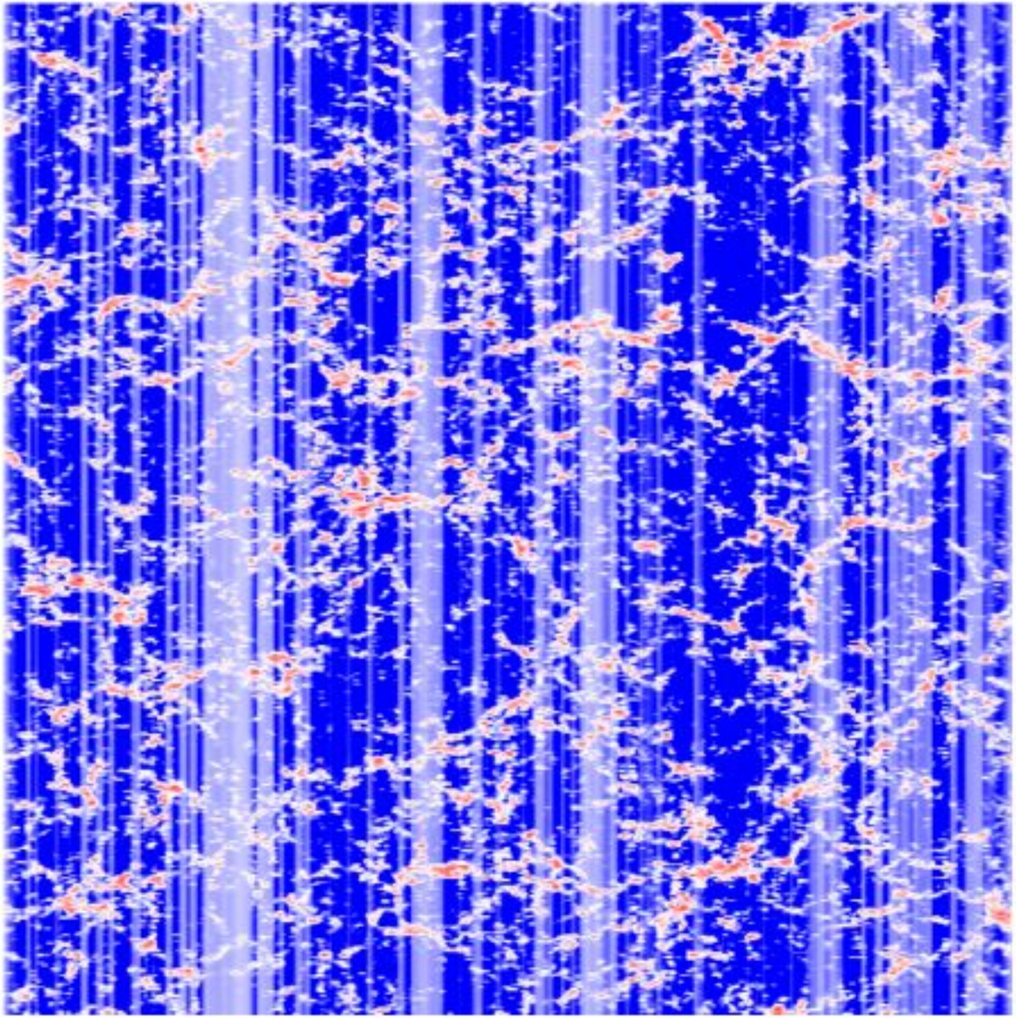
$$\chi^2 = \sum_{n,i} w_{n,i}[f_{n,i} - p(\lambda_{n,i}, a) \cdot F(\lambda'(\lambda_{n,i}, v_n), b)]^2$$

Obs flux        polynomial/vertical        template/shift

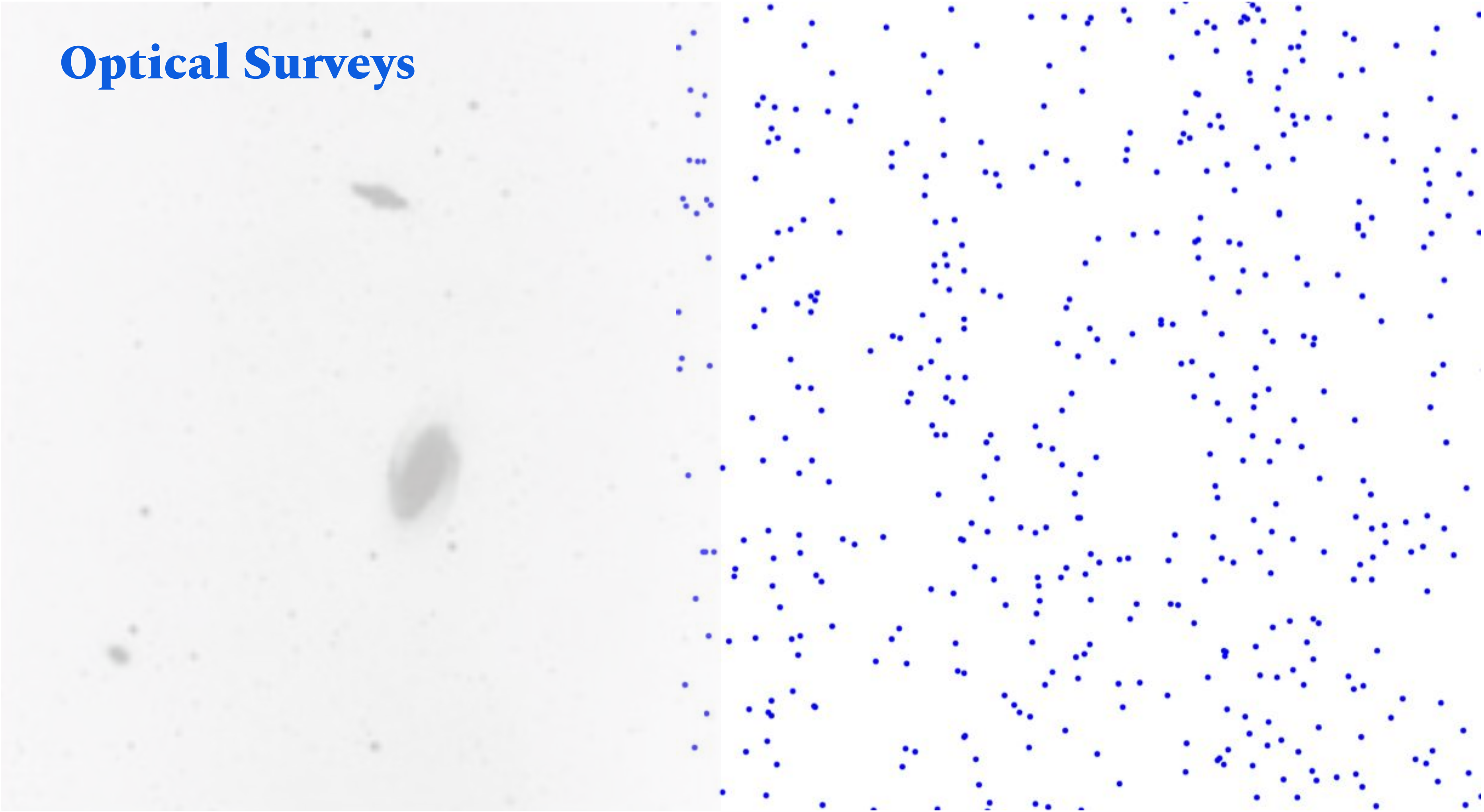
Time, orders, pixels



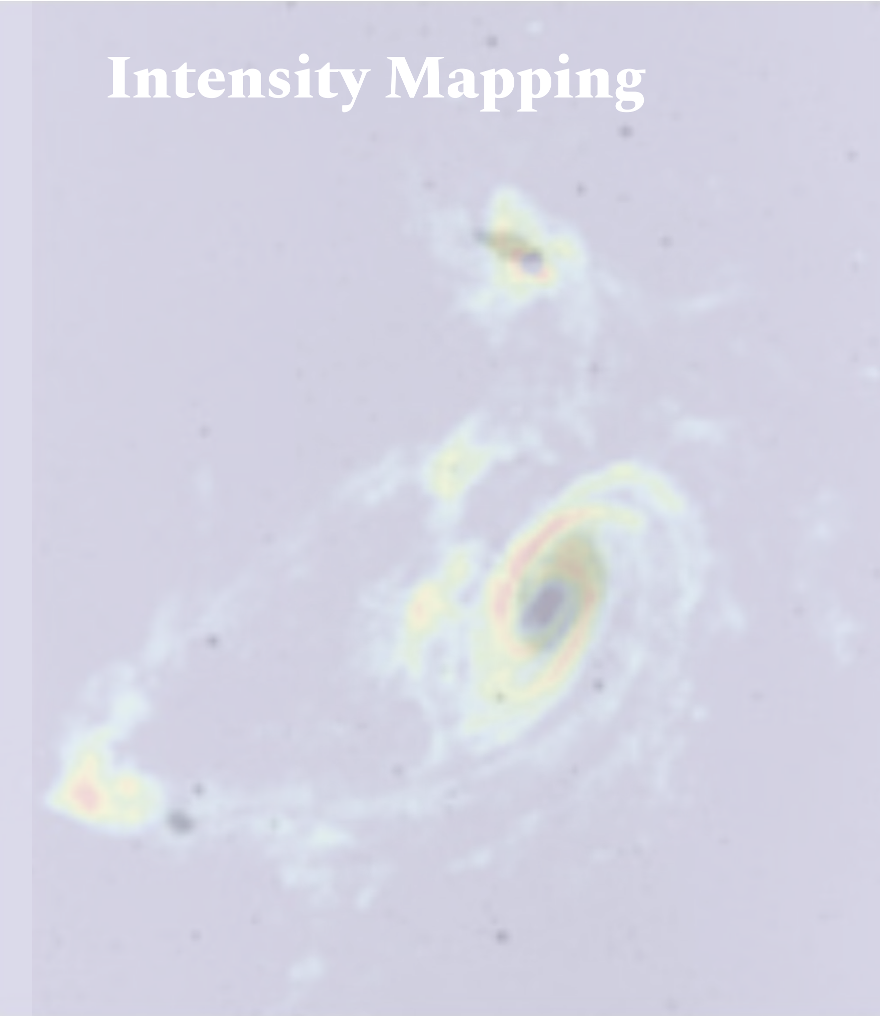
# Intensity Mapping

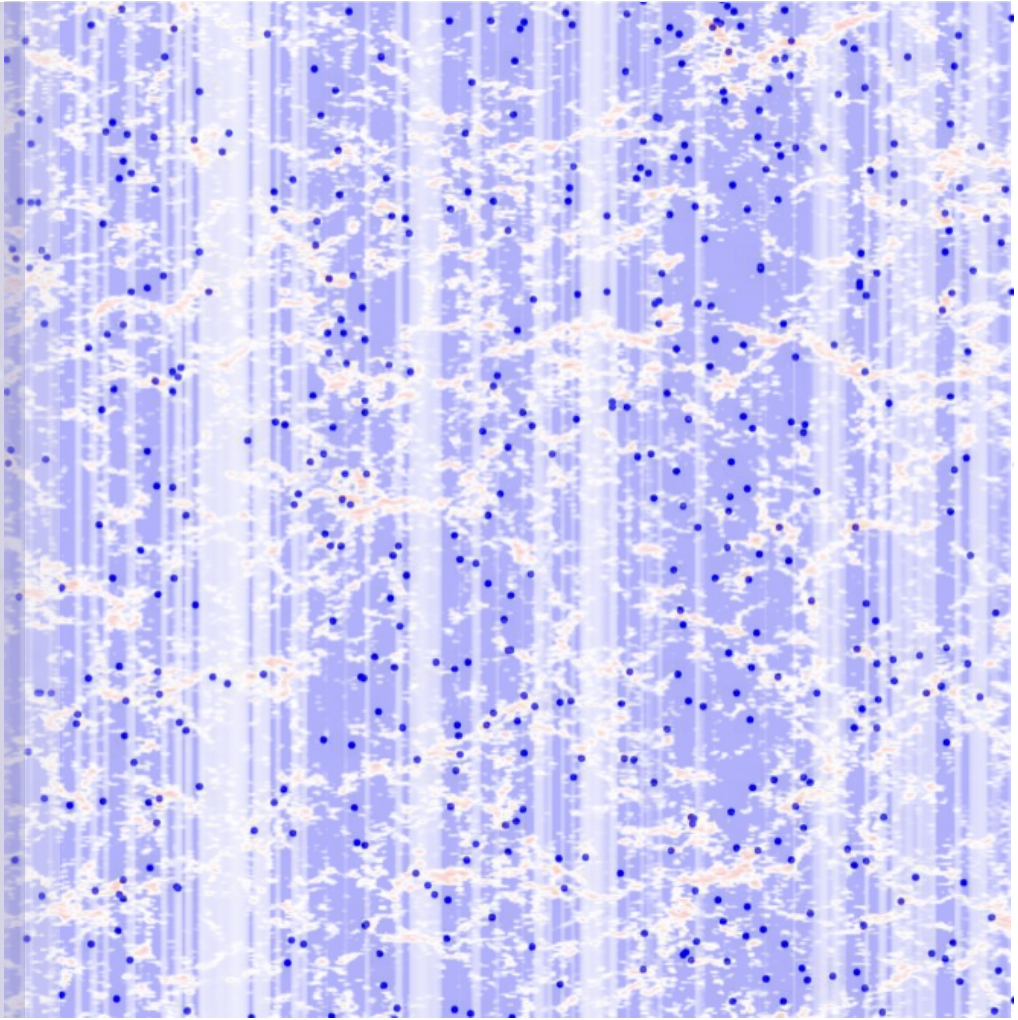


# Optical Surveys

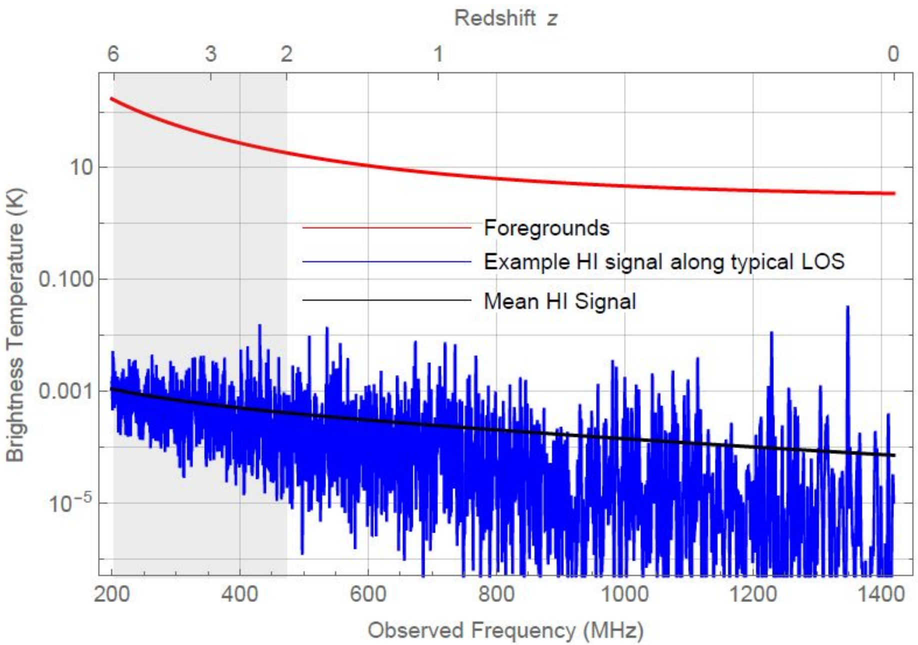


# Intensity Mapping



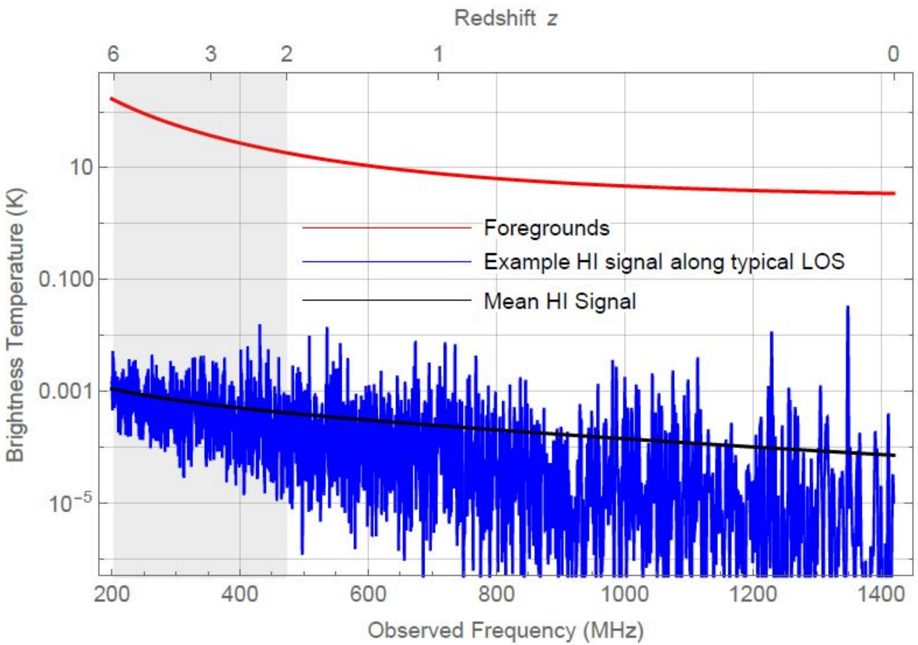


# Aims



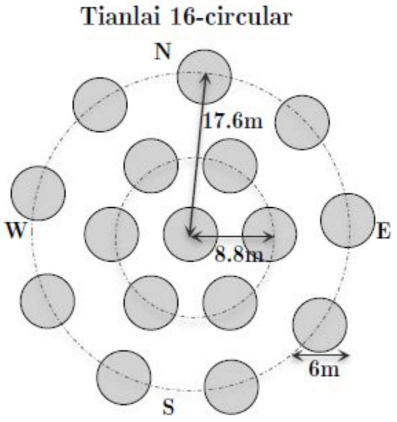

$$\sigma_{noise} = \frac{T_{sys}}{\sqrt{\Delta\nu\Delta T}}$$

# Aims

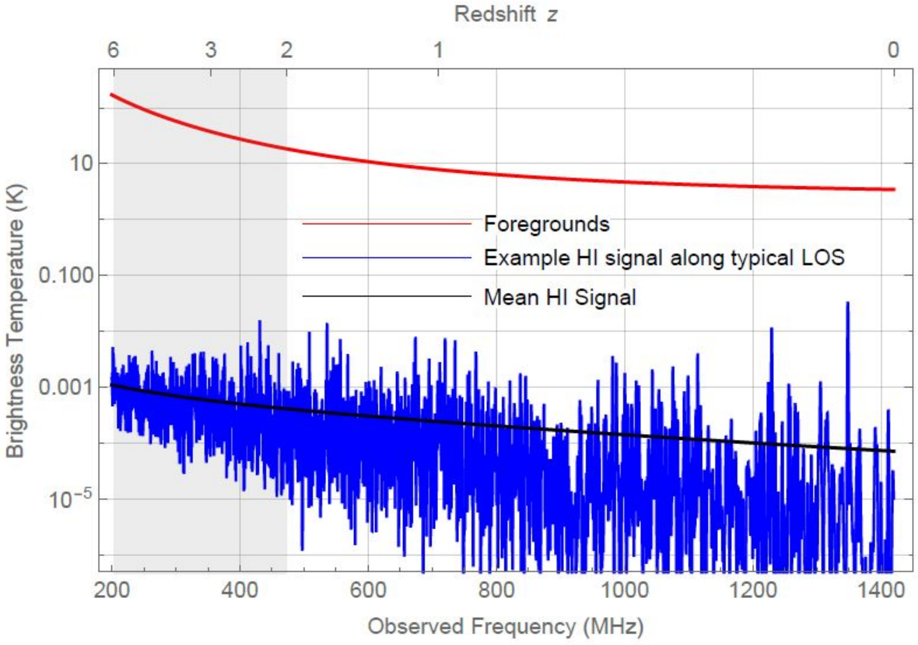



$$\sigma_{noise} = \frac{T_{sys}}{\sqrt{\Delta\nu\Delta T}}$$

NCP surveys

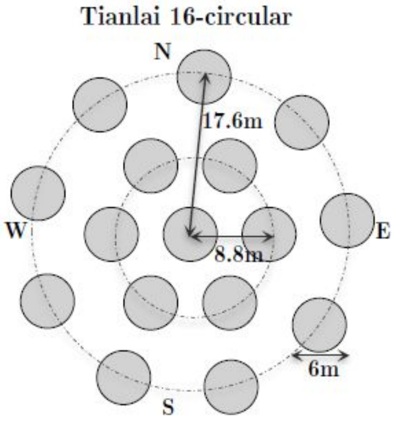


# Aims




$$\sigma_{noise} = \frac{T_{sys}}{\sqrt{\Delta\nu\Delta T}}$$

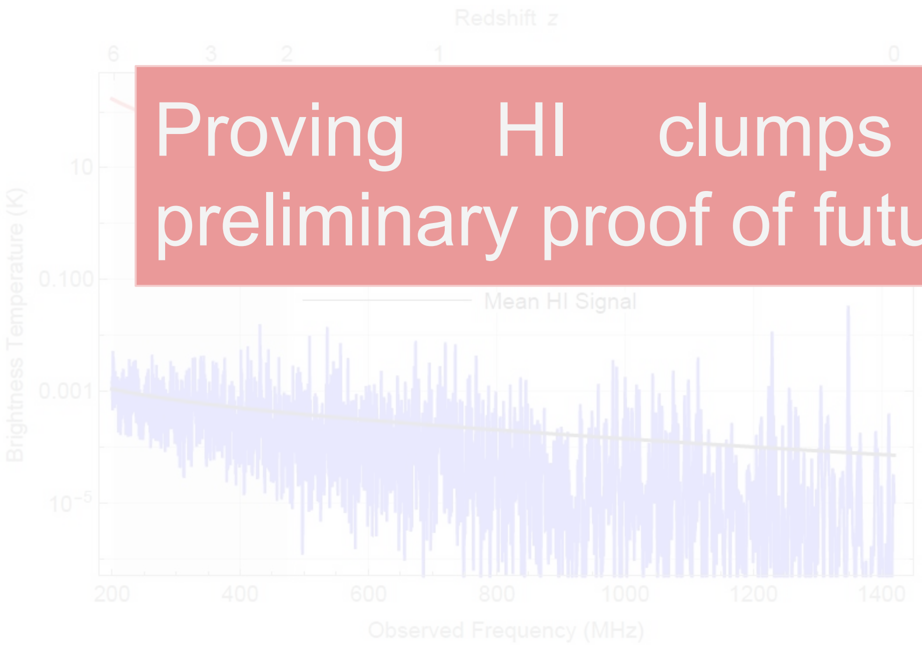
NCP surveys



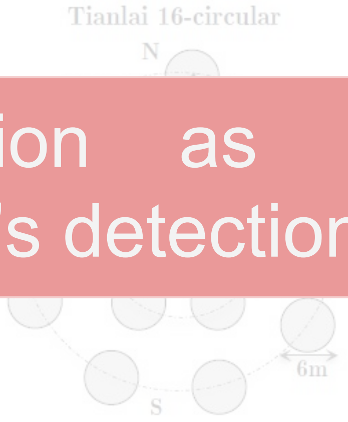

Redundancy

# Aims

**Proving HI clumps detection as a preliminary proof of future BAO's detection**




$$\sigma_{noise} = \frac{T_{sys}}{\sqrt{\Delta\nu\Delta T}}$$




Redundancy

# Aims

Proving HI clumps detection as a preliminary proof of future BAO's detection

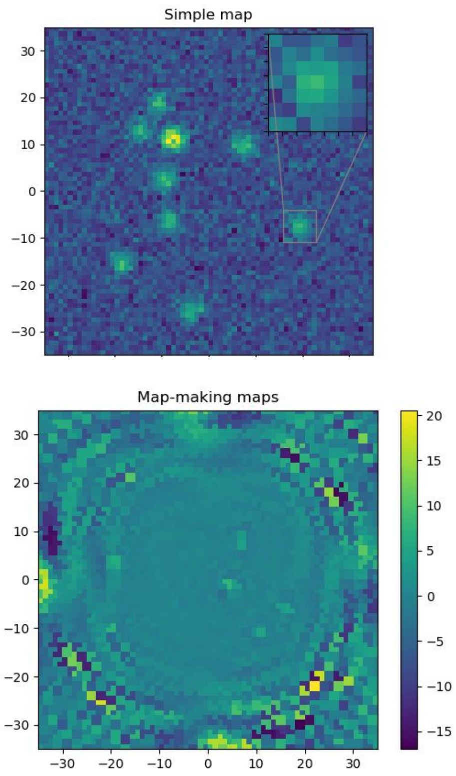
We detected with 70 % efficiency clumps with S/N=1.5 in 42 days per declination: <u>six months of total exposure</u>

$$\sigma_{noise} = \frac{T_{sys}}{\sqrt{\Delta\nu \Delta T}}$$

# Method

Sky-Maps Simulation | Source detection algorithm | Post processing

# Method

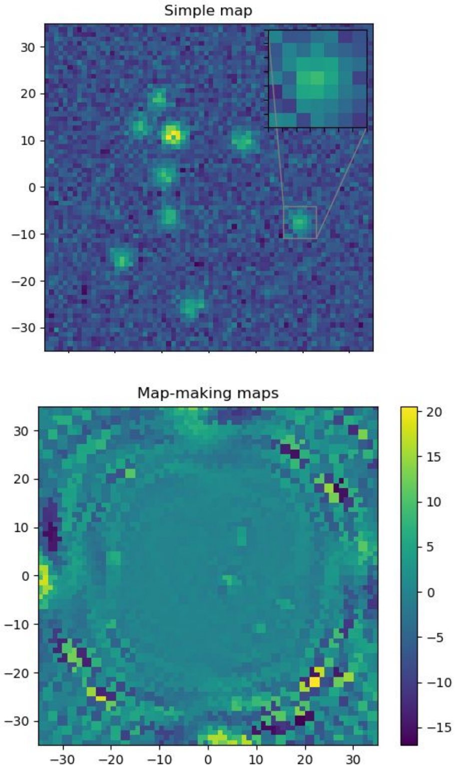


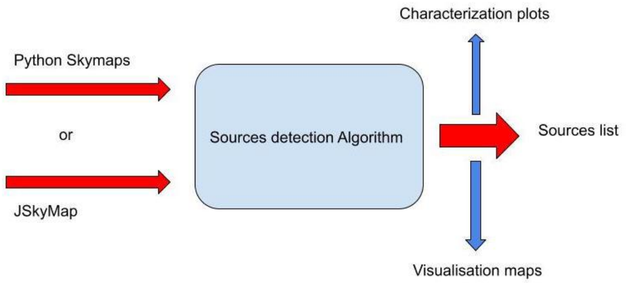

Sky-Maps Simulation

Source detection algorithm

Post processing

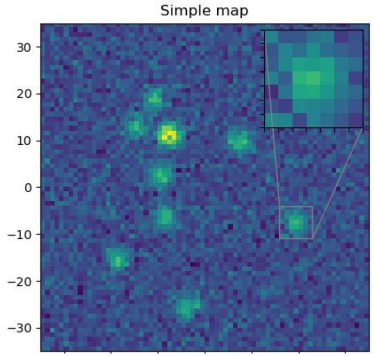
# Method

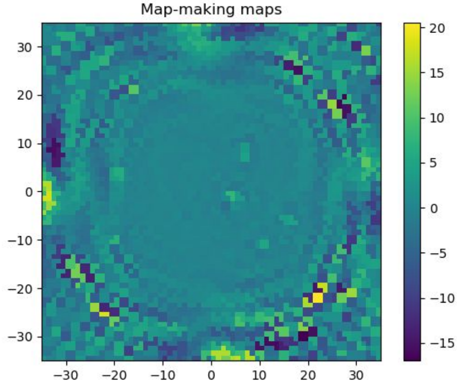

Simple map
Map-making maps

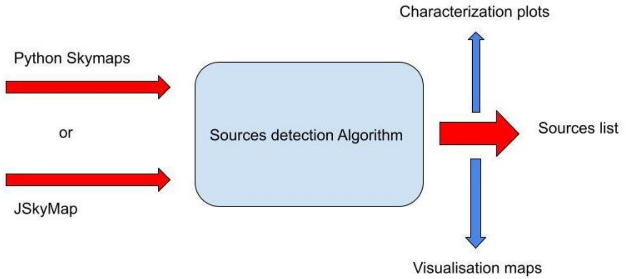

Python Skymaps
or
JSkyMap
Sources detection Algorithm
Characterization plots
Sources list
Visualisation maps

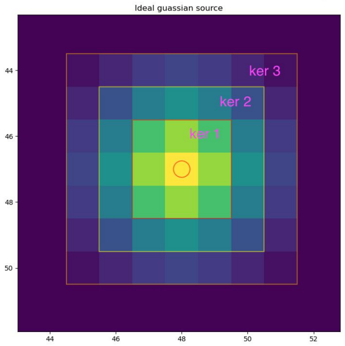

Sky-Maps Simulation
Source detection algorithm
Post processing

# Method



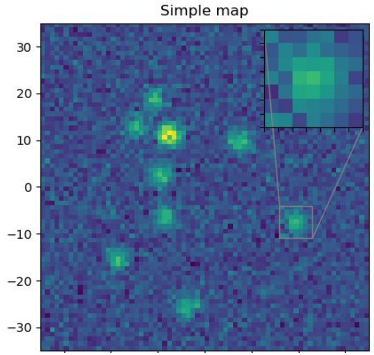



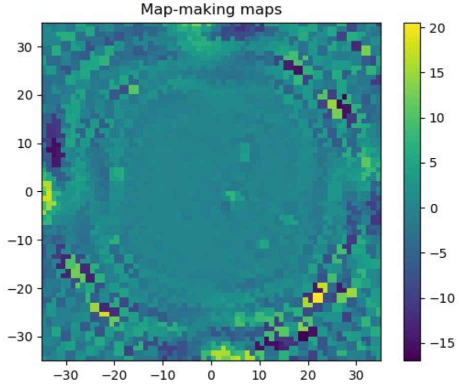



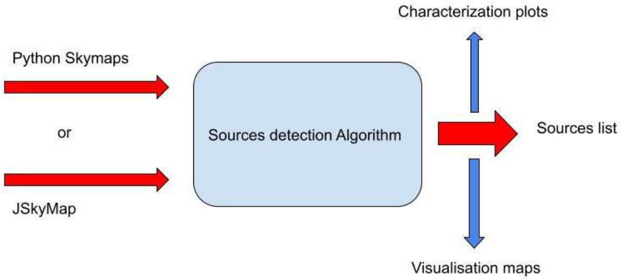

$$mean(ker1) > background + n \times \sigma_{bg}$$

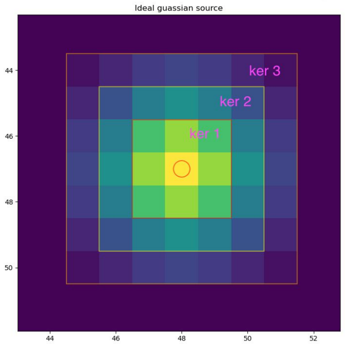
**Sky maps simulation** **Sources finder** **Post processing**

# Method

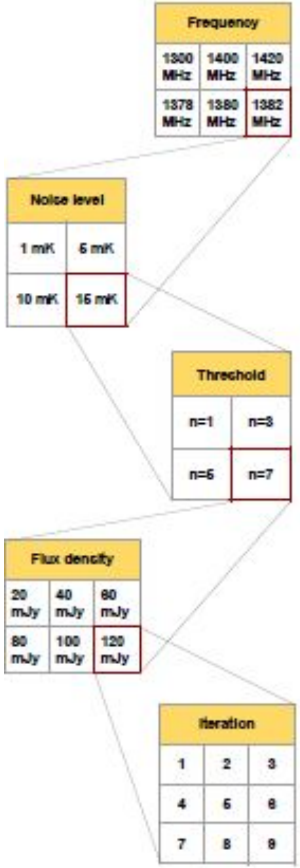


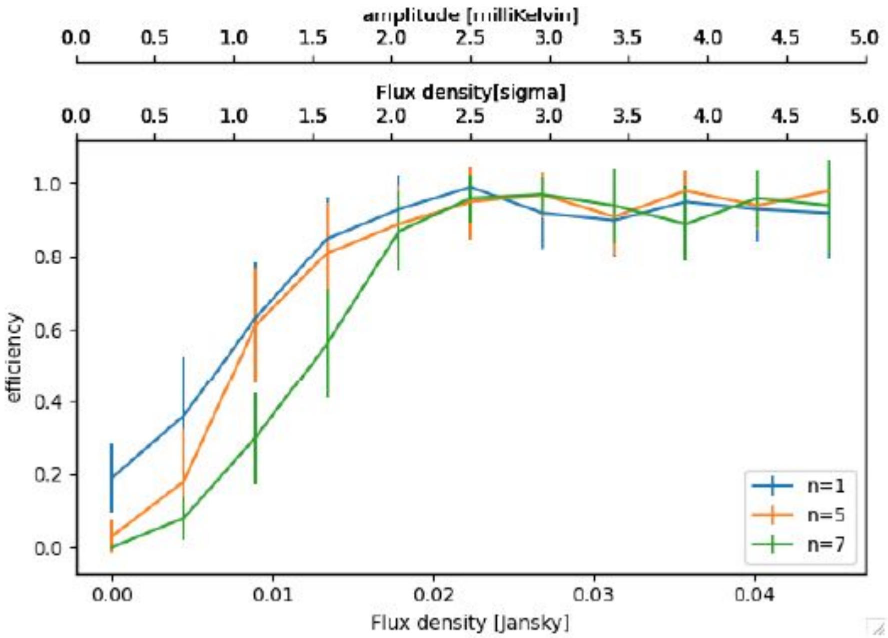



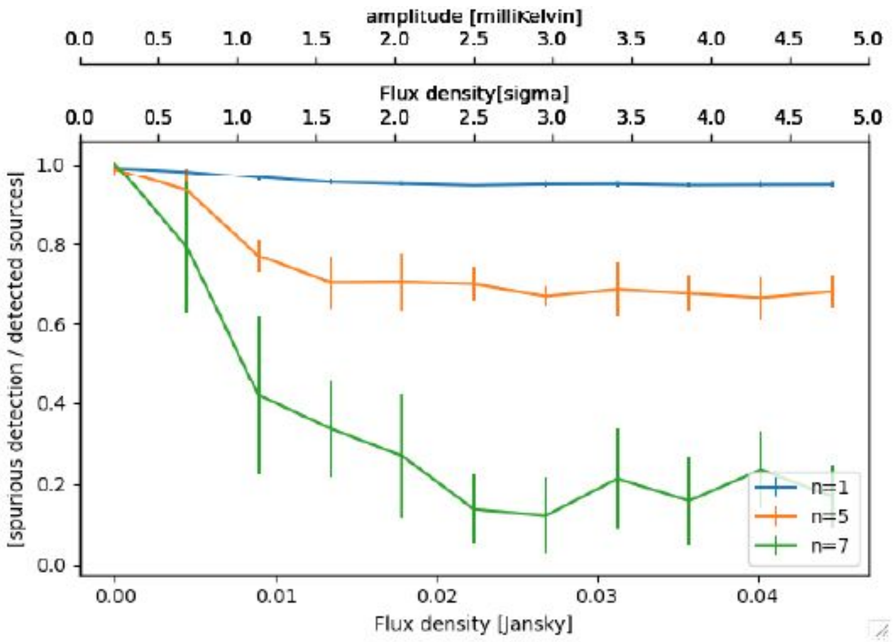

$$mean(ker1) > background + n \times \sigma_{bg}$$




**Sky maps simulation** **Sources finder** **Post processing**

# Results - Python maps



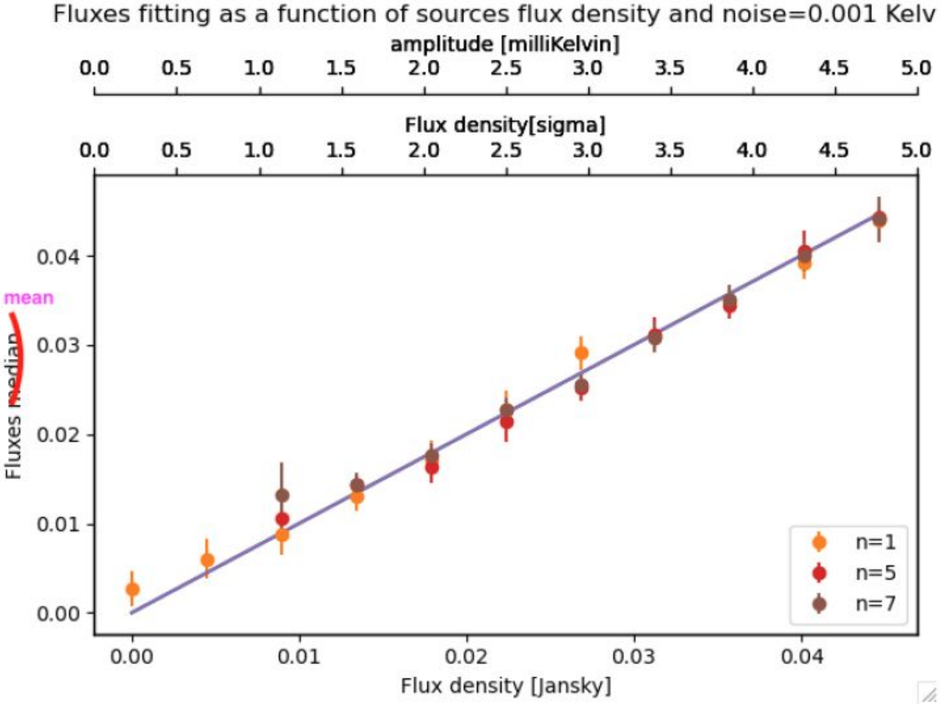



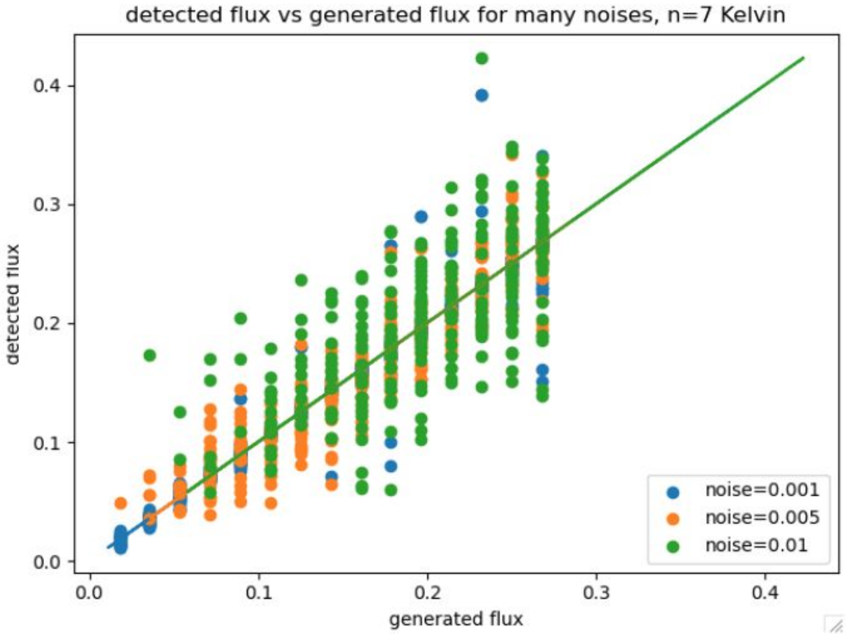

$$mean(ker1) > background + \boxed{n} \times \sigma_{bg}$$

Detection efficiency

False detection rate

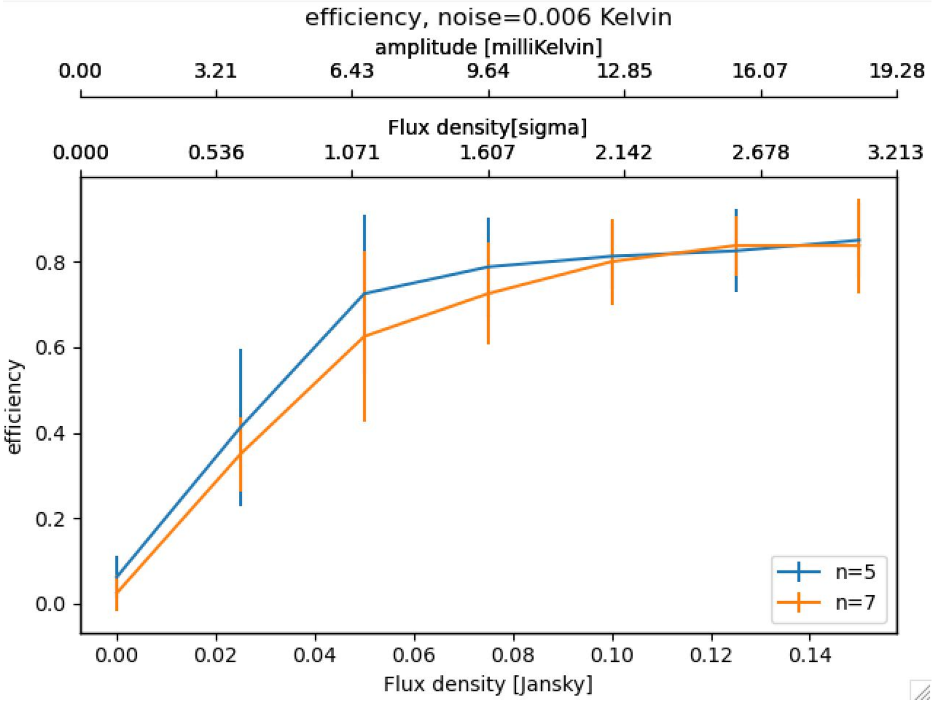
# Results - Python maps



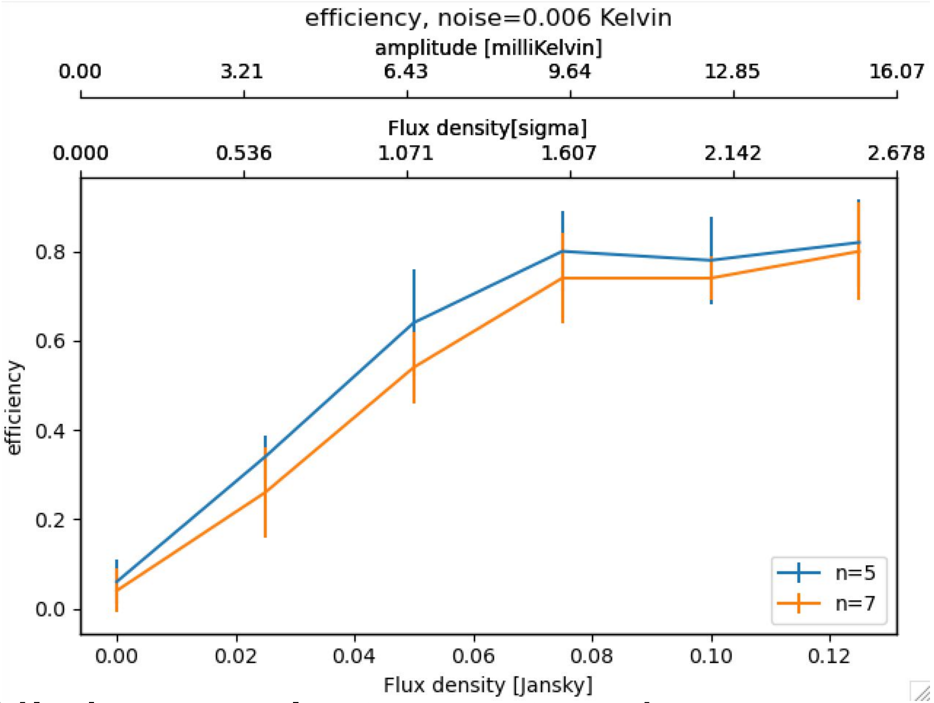




Reconstructed fluxes

# Results - JSkyMap maps




HI clumps + Instruments noise



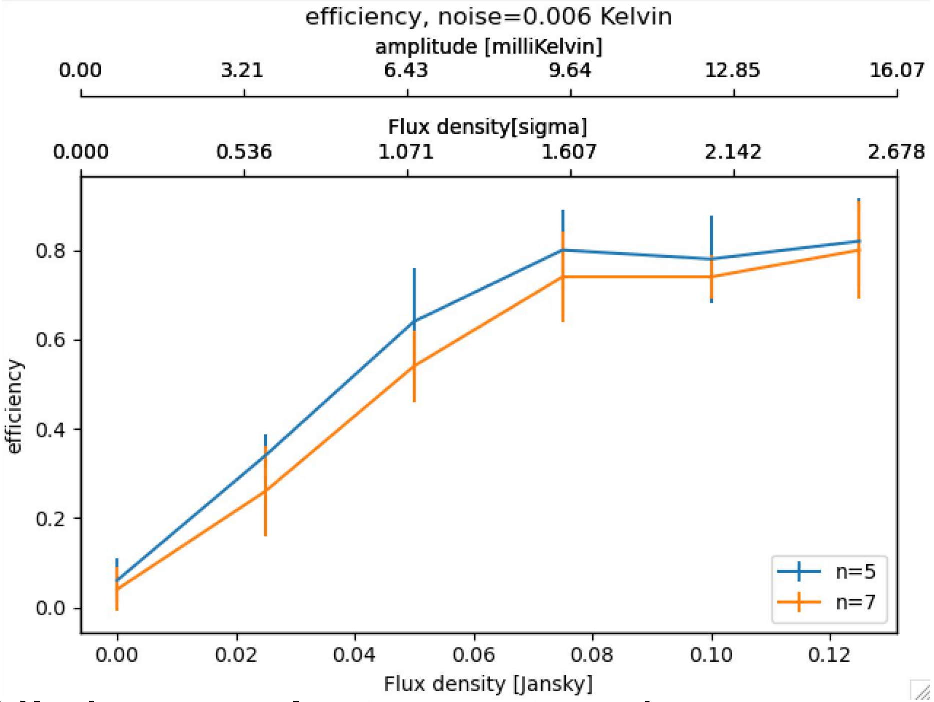

HI clumps + Instruments noise + Foregrounds. Results after subtraction.

# Results - JSkyMap maps

We can detect with an efficiency of 70 % clumps with S/N=1.5 in 42 days per declination.

This corresponds to six months of total exposure

Forecast arXiv:2205.06086




HI clumps + Instruments noise + Foregrounds. Results after subtraction.

Thank you