Reinforcement Learning for Radio Astronomy

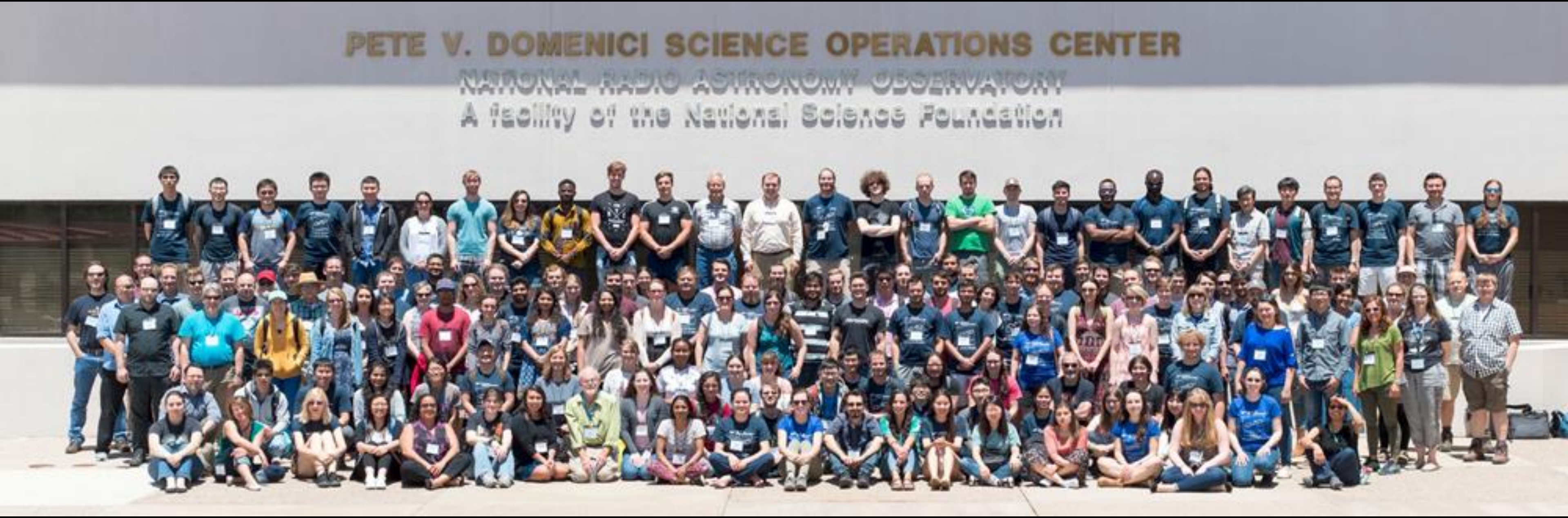
Brian Kirk, Urvashi Rau, Ramyaa

PhD Candidate at New Mexico Tech & NRAO Reber Fellow

CosmicAI NSF SIMONS FOUNDATION

Top image credit: S. Mantovanini & the GLEAM-X team Bottom image credit: Axel Mellinger, milkywaysky.com

NRAO Synthesis Imaging Workshop



CosmicAI

Credit: Jonah Rose / DREAMS Collaboration

THE CORE CHALLENGE

Can we teach the computer to make these decisions?

CosmicAI

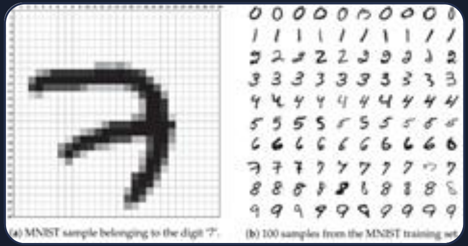
Credit: Jonah Rose / DREAMS Collaboration

Major Flavors of Machine Learning Algorithms

Supervised

Requires labeled data

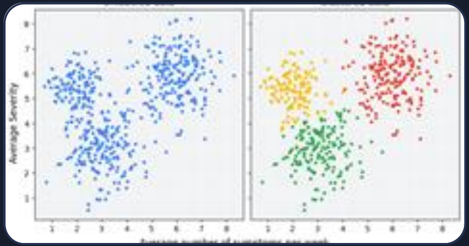
Learns a mapping function from input X → known label ‘Y’



Unsupervised

No labeled data required

Learns the underlying, hidden structure or patterns



Reinforcement

Generates its own training data

Learns by rewards, where the value of the reward *is* the label

Does require training environment

Note: These are learning *algorithms*, not architectures. Architectures include NNs, CNNs, RNNs, U-nets, Transformers, etc.

Ingredients for a Training Environment

Actions

What moves can be taken?

State Features

State of the board forms the input features.

Reward Function

To evaluate how good or bad a move is

Our Training Environment

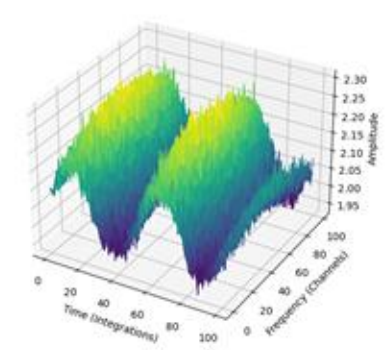
CASA Tasks as Actions

- Time-based corrections
- Freq-based corrections
- Flagging (rflag)
- Self-cal modeling
- Stack calibration tables

blcal	Calculate a baseline-based calibration solution (gain or bandpass)
clearcal	Re-initializes the calibration for a visibility data set
defintent	Manually set scan intents
fluxscale	Bootstrap the flux density scale from standard calibrators
fringefit	Fringe fit delay and rates
gaincal	Determine temporal gains from calibrator observations
gencal	Specify Calibration Values of Various Types
getantposalma	Retrieve antenna positions by querying ALMA web service.
getcalmodvla	Retrieve calibrator brightness distributions from a VLA web service.
initweights	Initializes weight information in the MS
polcal	Determine instrumental polarization calibrations
polfromgain	Derive linear polarization from gain ratio

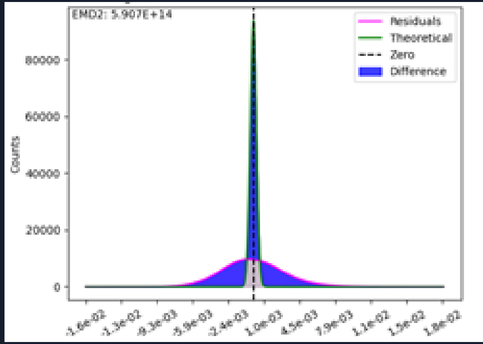
Dataset State Features

- **Vis: Mean gain**
- **Vis: Std-dev in freq & time**
- **Vis: Max value**
- **From both DATA and MODEL**
- **Distance to theoretical limit**



Reward Function

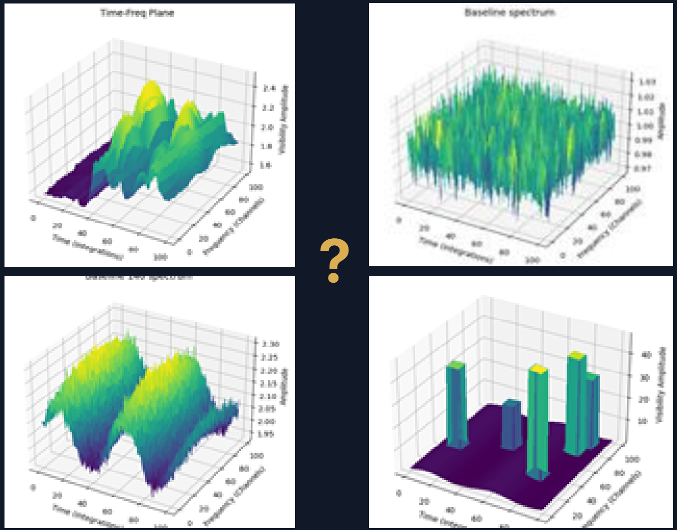
Objective: The closer the sky model's residuals were to theoretical noise, the higher the reward value

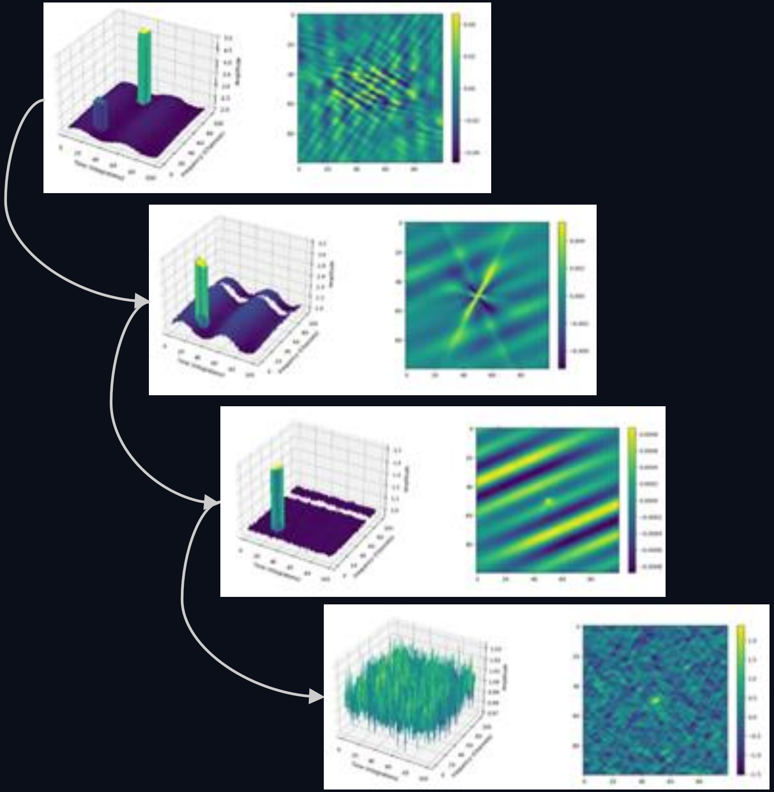


DQN Policy On Simulated Datasets

What should be done in each scenario?

1000's of sims with gains, RFI, multiple sources, and differing signal-to-noise ratios





Training on Calibrator Survey Datasets

>23,000

Total Datasets, split down to individual scans and spws

8 Calibrator Targets

3C48, 3C123, 3C138, 3C147, 3C196, 3C286, 3C295, 3C380

2 Array Configurations

Observed with A-config and C-config configurations

9 Frequency Bands

A, C, K, L, P, Q, S, U, and X bands

Calibrator Surveys Repeated ~Monthly

*The application to real data was chosen for seed funding by **CosmicAI**, a National Science Foundation (NSF) AI Institute.*

National Scale Computing

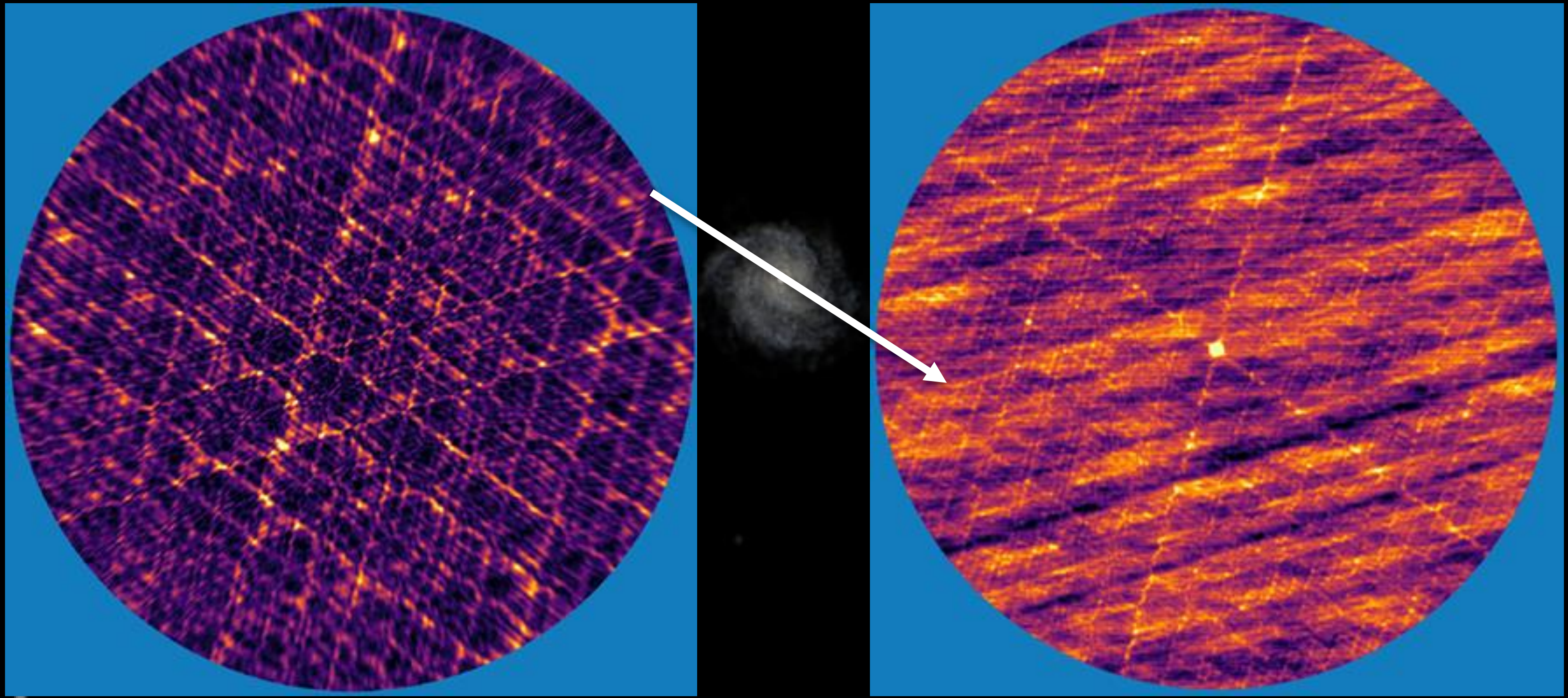
Using dHTC for Experience Collection

~80 clusters
contributing to OSG

CosmicAI Credit: Jonah Rose / DREAMS Collaboration

3C286 Single Scan and SPW - 1st Step - Self-Cal

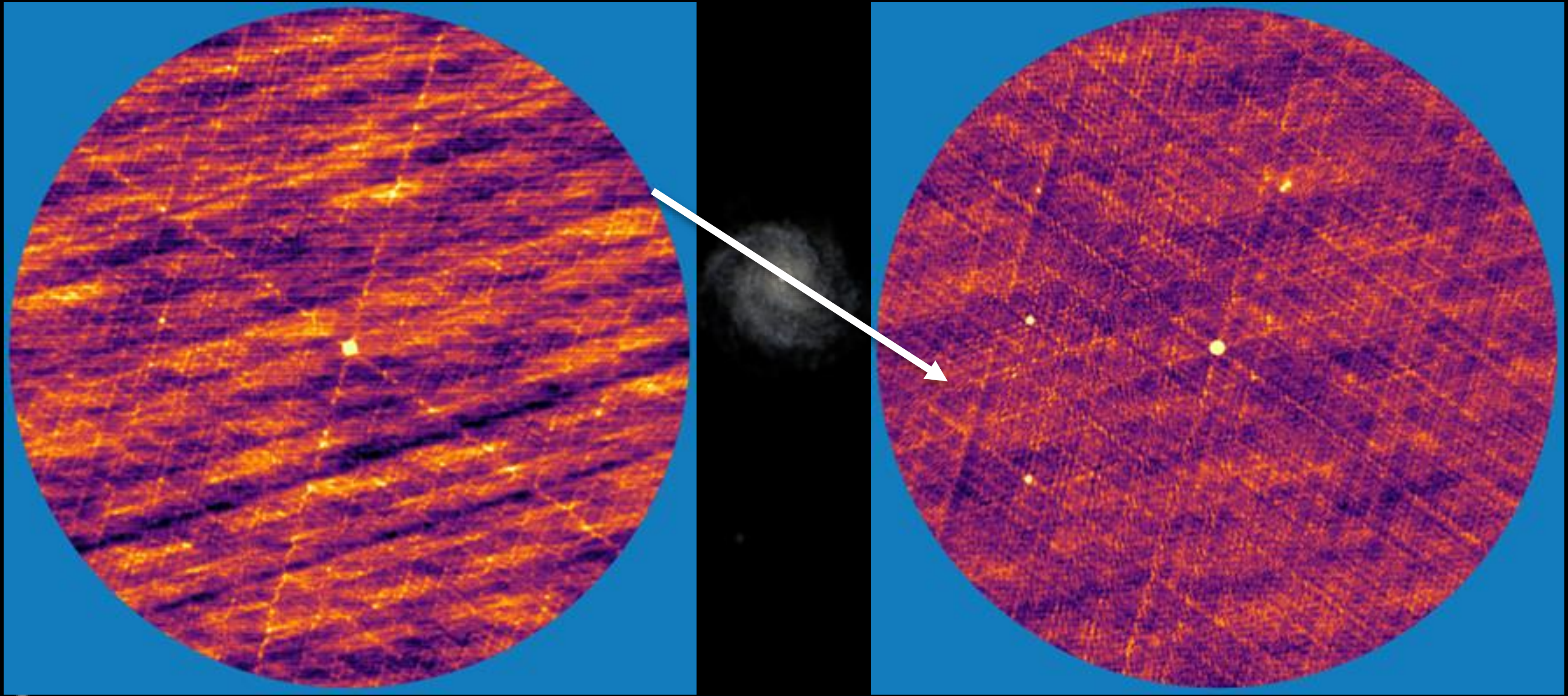
C-Config L-band • LHI Spw6 Scan4 • 64 channels • ~15 seconds



Credit: Jonah Rose / DREAMS Collaboration

3C286 Single Scan and SPW - 2nd Step - Flag

C-Config L-band • LHI Spw6 Scan4 • 64 channels • ~15 seconds

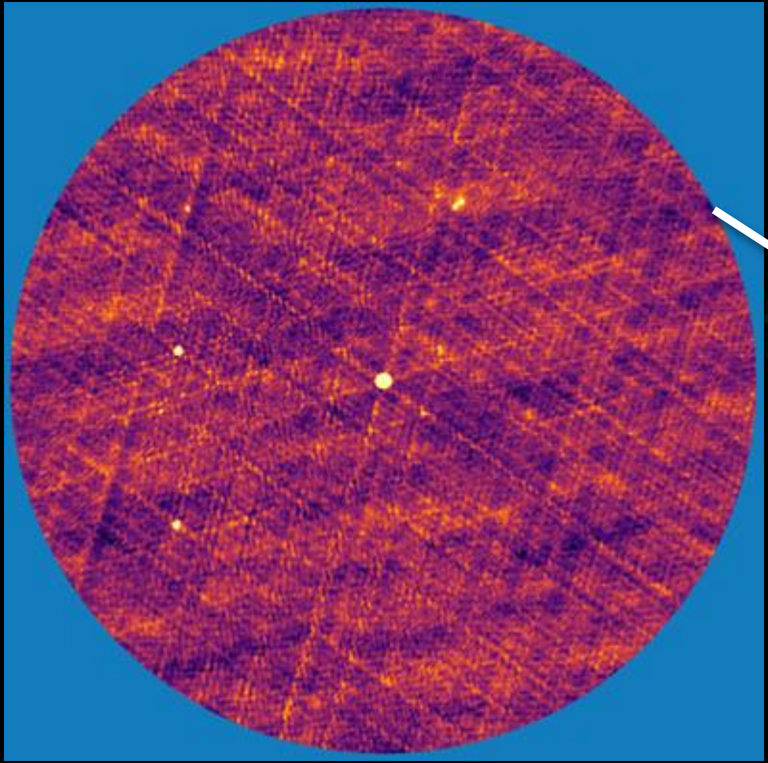


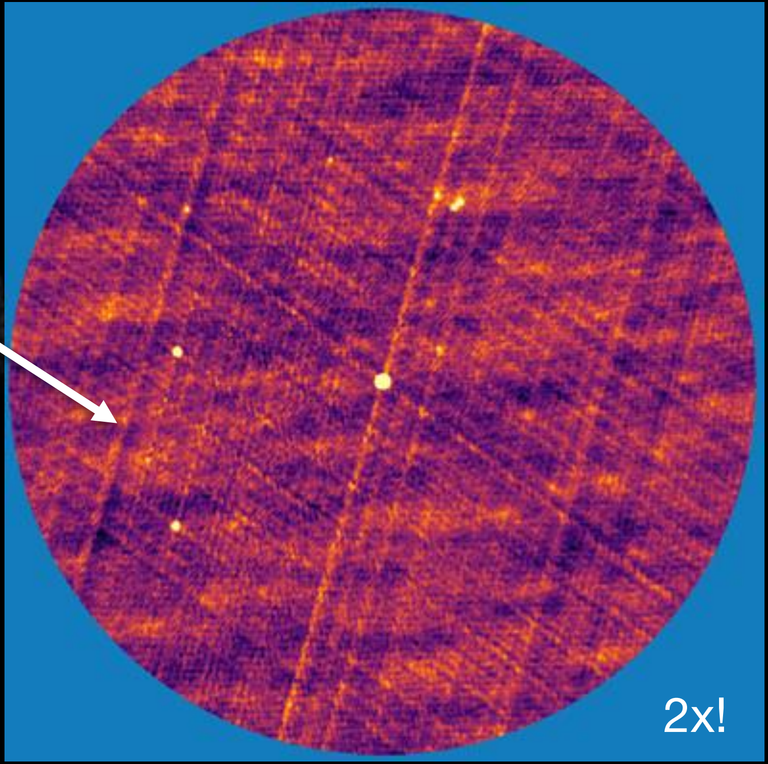
CosmicAI

Credit: Jonah Rose / DREAMS Collaboration

3C286 Single Scan and SPW - 3rd Step - Bandpass

C-Config L-band • LHI Spw6 Scan4 • 64 channels • ~15 seconds



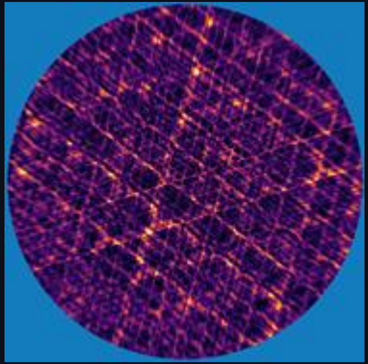


Credit: Jonah Rose / DREAMS Collaboration

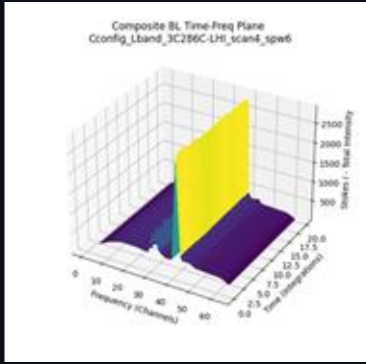
3C286 Single Scan & SPW - 3 step rollout

C-Config L-band LHI Spw6 Scan4 • 64 channels • ~15 seconds

Initial Image & Spectrogram



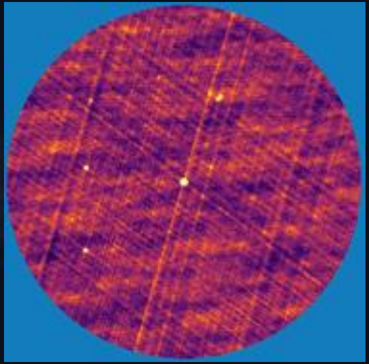
Raw Image



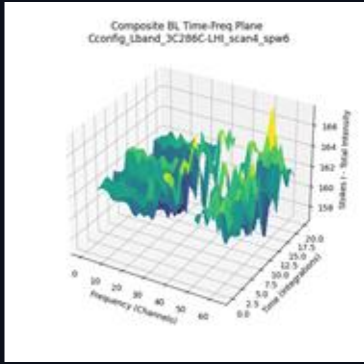
3D Spectrogram

Time-frequency plane intensity

Post-Flagging & Calibration



Resulting Image



3D Spectrogram

Time-freq plane after self-cal, flag, and bandpass

CosmicAI

Credit: Jonah Rose / DREAMS Collaboration

Current Standing

Data

Training on 23k Datasets

Drastically expanded ranges of values to learn from (many more gradients in loss landscape)

The trained policy covers larger parameter space to infer from, reducing OoD chaos

Environment

Better Env → Better Policy

Failures occur when some other feature of the dataset not captured in the state (e.g., uv coverage) distorts an actions performance.

When found, these features can be added to the training environment for improved capability.

The Goal

The Goal

Convenience.

Rather than many policies, we prefer iterating the environment until a single robust policy can handle all spws and all scans of calibrator datasets.