Data Processing for Interferometry

A Reinforcement Learning approach

Brian Kirk, Urvashi Rau, Ramyaa
New Mexico Institute of Mining and Technology
NRAO

1. **The problem and the vision**
2. ***How* to solve it**
3. **What we have done**
4. **What we are going to do**

What is wrong with the current approach?

Slow as *developers* find heuristics

Inflexible pipelines

Expensive approach

Requires lots of human overhead

Vision: dataset-specific processing, at scale

which requires solving automation bottlenecks

Slow as *developers* find heuristics

Inflexible pipelines

Expensive approach

Requires lots of human overhead

Offload heuristic finding to computers

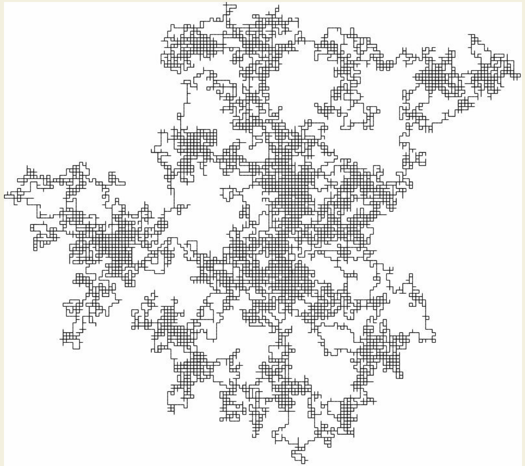
Make decisions, don't follow recipes

Data-driven, a path to SRDP

How?

Reframe as a path-finding and cost-minimization problem

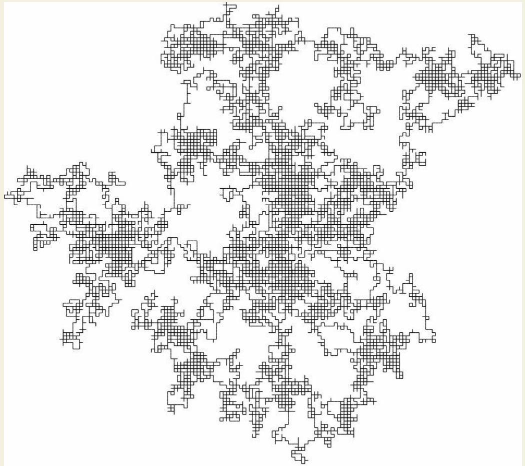
Use RL for Path-finding and Cost Minimization



Random walks to explore the parameter space

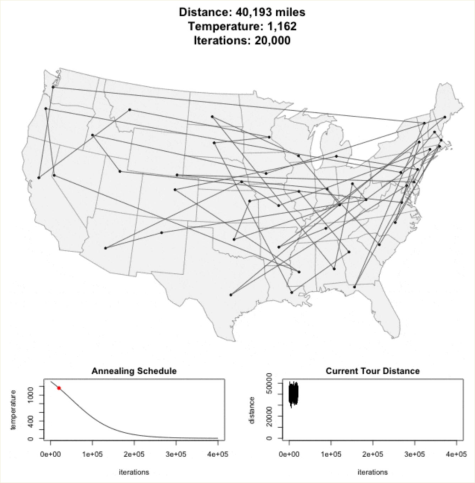
- computer doing the heuristic search

Use RL for Path-finding and Cost Minimization



Random walks to explore the parameter space

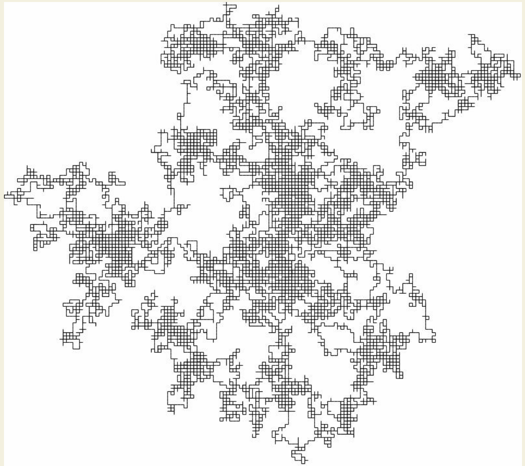
- computer doing the heuristic search



Simulated annealing constricts randomness for cost minimization

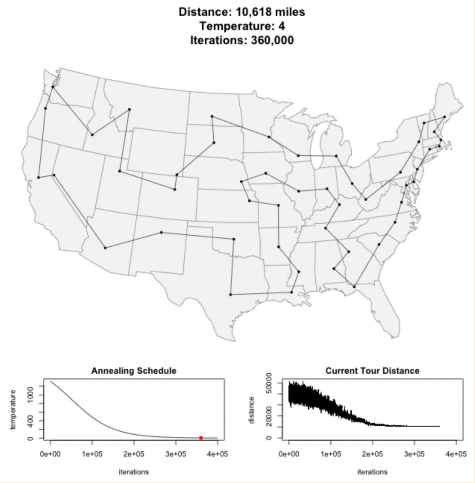
- prevents greedy behavior
- resistant to local minima

Use RL for Path-finding and Cost Minimization



Random walks to explore the parameter space

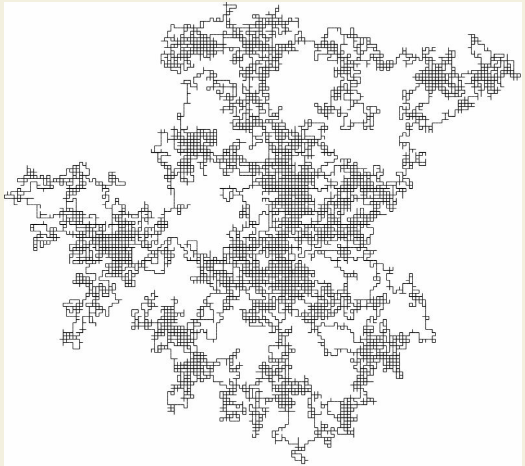
- computer doing the heuristic search

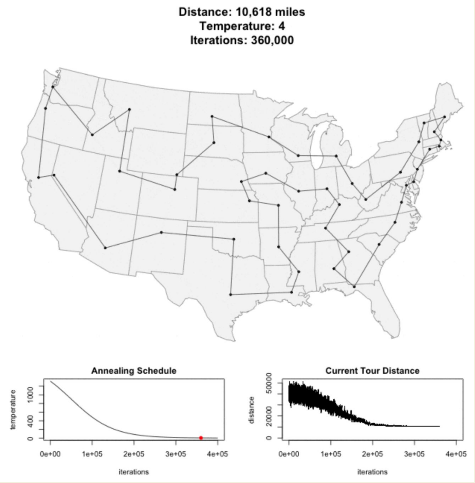


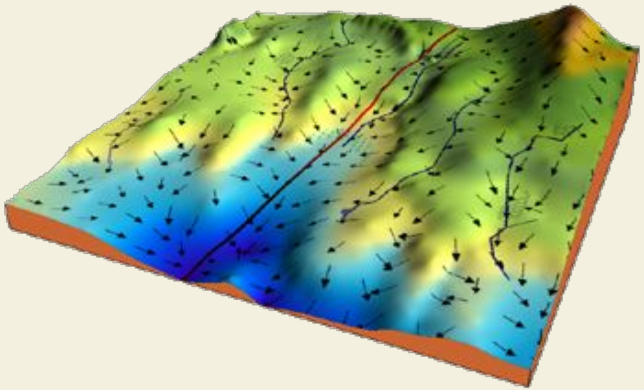
Simulated annealing constricts randomness for cost minimization

- prevents greedy behavior
- resistant to local minima

Use RL for Path-finding and Cost Minimization







Random walks to explore the parameter space

- computer doing the heuristic search

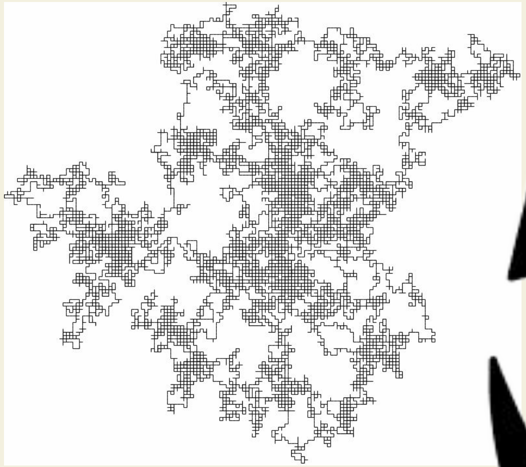
Simulated annealing constricts randomness for cost minimization

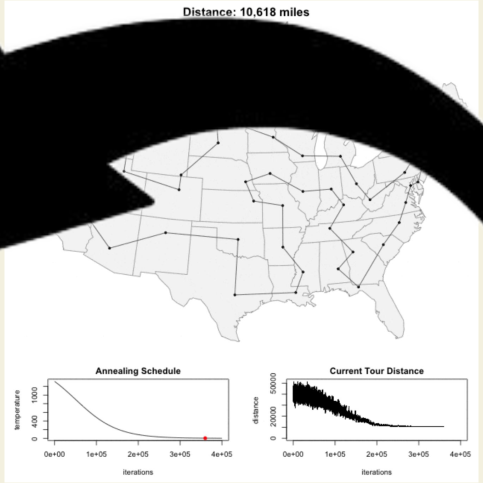
- prevents greedy behavior
- resistant to local minima

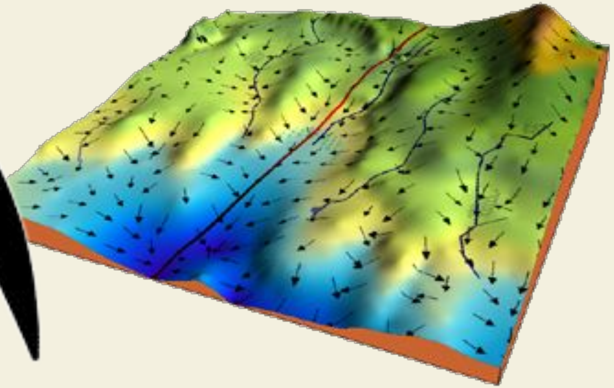
NNs build a model of understanding from the sampling

- avoids exhaustive search
- iteratively improves

Use RL for Path-finding and Cost Minimization







Random walks to explore the parameter space

- computer doing the heuristic search

Simulated annealing constrict
andomness for cost

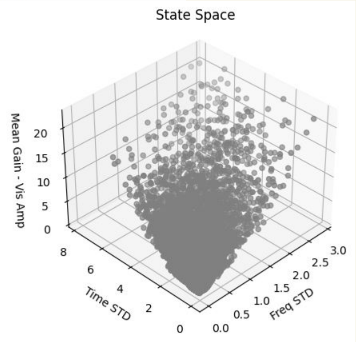
- re

NNs build a model of understanding from the sampling

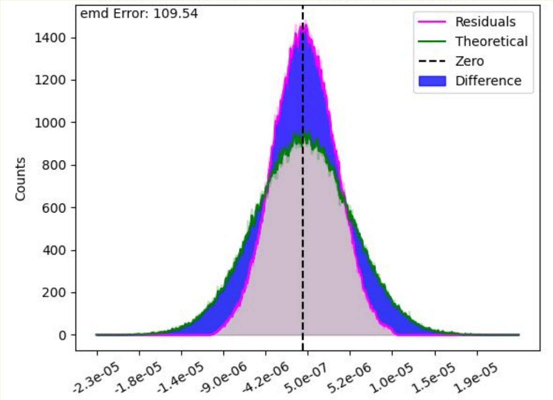
- avoids exhaustive search
- iteratively improves

How we use it

KEYS	DESCRIPTION
blc	absolute PIXEL coordinate of the bottom left corner of the bounding box surrounding the selected region
blcf	Same as blc, but uses WORLD coordinates instead of pixels
trc	the absolute PIXEL coordinate of the top right corner of the bounding box surrounding the selected region
trcf	Same as trc, but uses WORLD coordinates instead of pixels
flux	the flux or flux density. See below for details.
npts	the number of unmasked points used
max	the maximum pixel value
min	minimum pixel value
maxpos	absolute PIXEL coordinate of maximum pixel value
maxposf	Same as maxpos, but uses WORLD coo
minpos	absolute pixel coordinate of minimum p
minposf	Same as minpos, but uses WORLD coor
sum	the sum of the pixel values: $\sum I_i$
sumsq	the sum of the squares of the pixel valu
mean	the mean of pixel values : $\bar{I} = \sum I_i/n$
sigma	the standard deviation about the mean:
rms	the root mean square : $\sqrt{\sum I_i^2/n}$
median	the median pixel value
medabsdevmed	the median of the absolute deviations fr
quartile	the inner-quartile range. Find the points
q1	the first quartile
q3	the third quartile



`accor`	Normalize visibilities based on auto-correlations
`applycal`	Apply calibrations solutions(s) to data
`bandpass`	Calculates a bandpass calibration solution
`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
`rerefant`	Re-apply refant to a caltable
`smoothcal`	Smooth calibration solution(s) derived from one or more sources
`wvrgcal`	Generate a gain table based on Water Vapour Radiometer data



Dataset properties define the "location"

– *data-driven* specifics

CASA tasks are the "actions" you can take to navigate

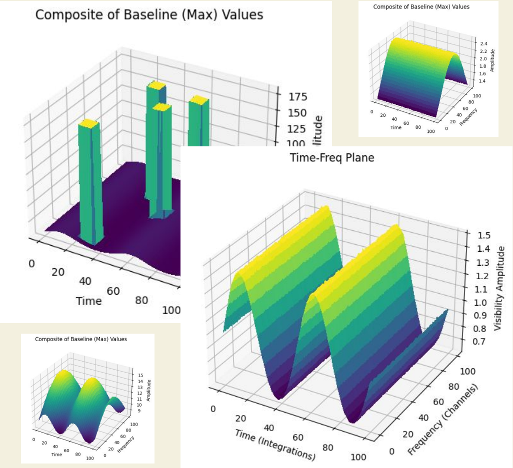
– evaluated on performance towards an objective

The objective is the "destination"

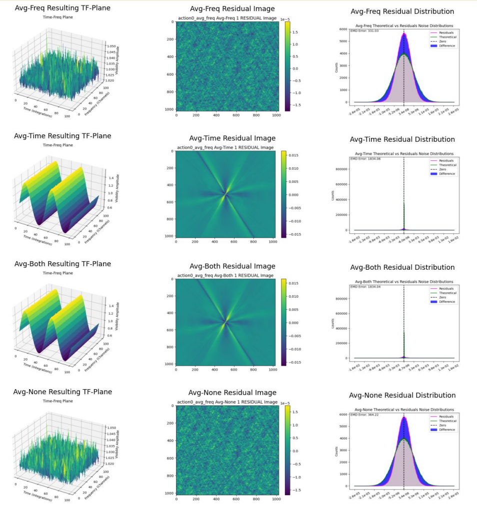
– the feedback signal for pathfinding

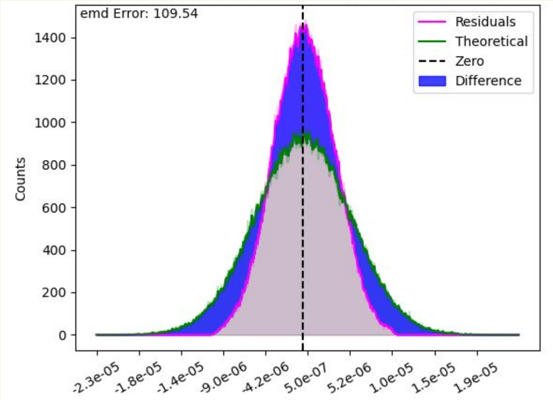
– defined by the user/observatory

A simplified scenario in calibration



Created ~5000 sims of varying gains & RFI





Chosen dataset features:

- Mean amplitude
- Time-std
- Freq-std
- Max amplitude

Actions available:

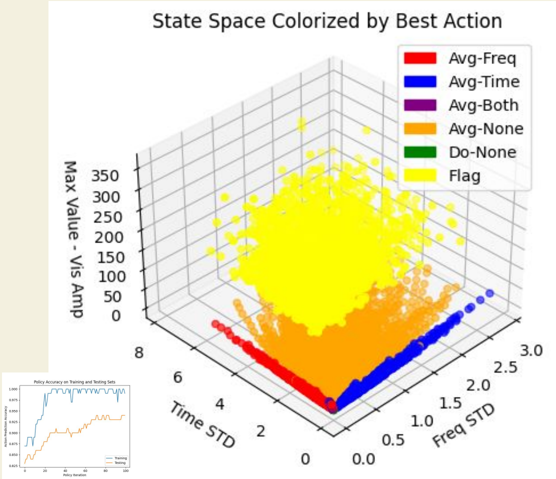
- Average freq & calibrate
- Average time & calibrate
- Avg-Both & calibrate
- Avg-None & calibrate
- Do Nothing (not pictured)
- Flag (not pictured)

Metric to evaluate actions:

- EMD of actual vs theoretical noise
- Runtime

Runtime penalty prevents using the most expensive algorithms all the time*

Results from the simplified scenario

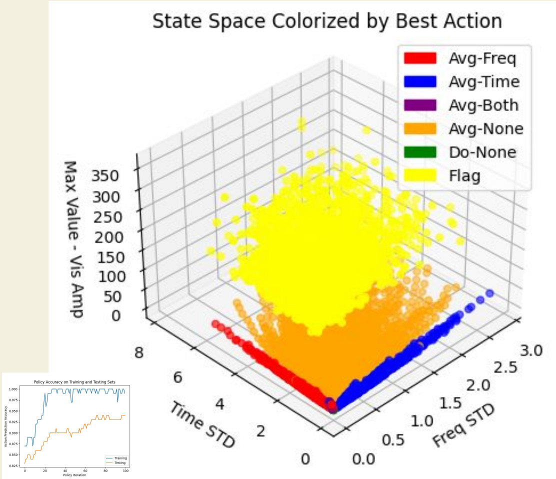


Policy applied to 5000 simulations

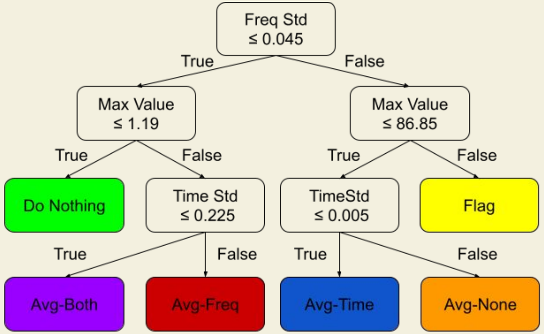
1% of sims used for policy training

- 50 sims for 100 RL-iterations
- >92% accuracy on ~5000 unseen sims
- solved sequencing and actions

Results from the simplified scenario



Policy applied to 5000 simulations



Decision Tree Classifier

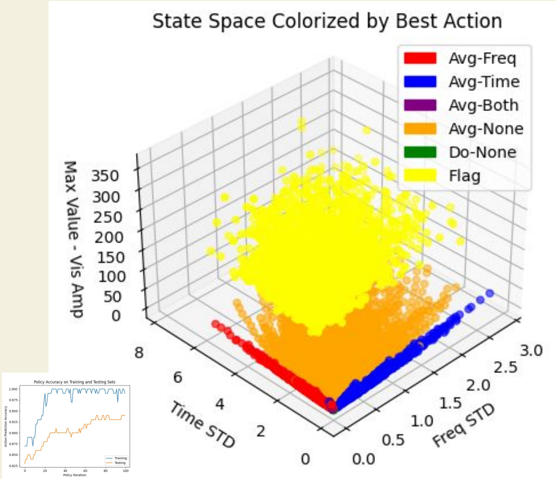
1% of sims used for policy training

- 50 sims for 100 RL-iterations
- >92% accuracy on ~5000 unseen sims
- solved sequencing and actions

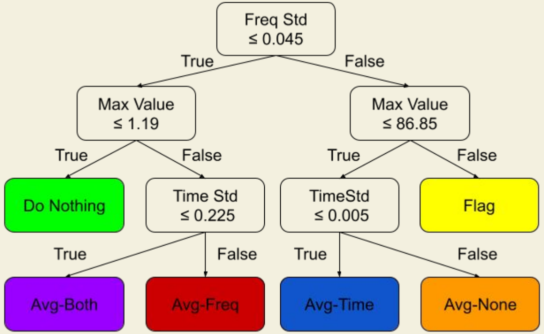
DT for a human readable policy

- RL found the thresholds, not humans
- tree can be validated by experts

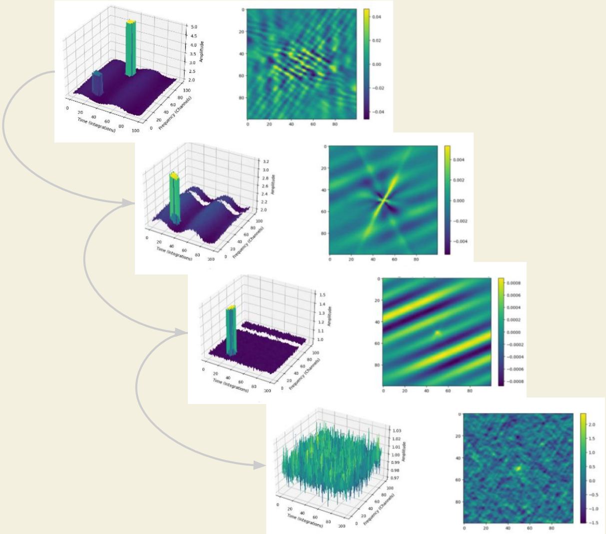
Results from the simplified scenario



Policy applied to 5000 simulations



Decision Tree Classifier



Flag → Avg-None → Flag again → Stop

1% of sims used for policy training

- 50 sims for 100 RL-iterations
- >92% accuracy on ~5000 unseen sims
- solved sequencing and actions

DT for a human readable policy

- RL found the thresholds, not humans
- tree can be validated by experts

Found its way to noise on its own

- no instructions given to do this

What we are going to do

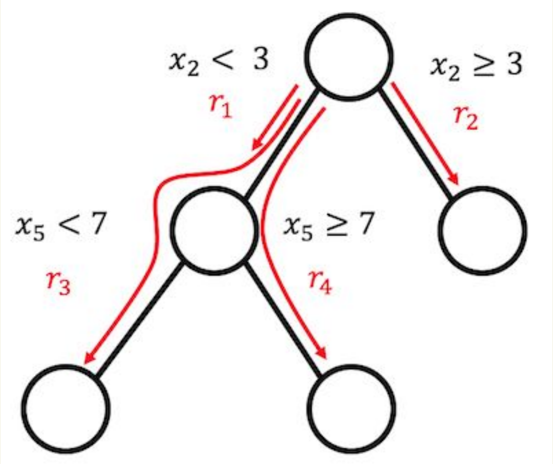
Expanding from here

Expand Scope of Decision Environment

1. Expand the # of actions available
2. Include multiple / diffuse sources
3. Aiming to learn self-cal process

Expanding from here

Expand Scope of Decision Environment



Find scale-invariant rules, not thresholds

1. Expand the # of actions available
2. Include multiple / diffuse sources
3. Aiming to learn self-cal process

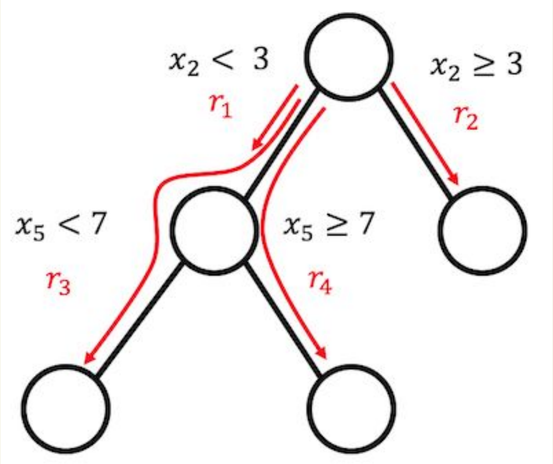
Instead of thresholds, find rules:
- rules are more generic to transfer
- same rules for 100 → 100k ints/chans

Test sim-to-real transfer of rules

Expanding from here

Expand Scope of Decision Environment

1. Expand the # of actions available
2. Include multiple / diffuse sources
3. Aiming to learn self-cal process



Find scale-invariant rules, not thresholds

Instead of thresholds, find rules:
- rules are more generic to transfer
- same rules for 100 → 100k ints/chans

Test sim-to-real transfer of rules

Move to real data

Calibrator source catalog is real data closest to our current sims