RFI mitigation for the ngVLA

- **Background**
  - Types of RFI (derived from A.Erickson's talk last week)
  - Options at different stages in the signal chain
    - Pre-correlation voltage streams
    - Post-correlation high time resolution visibilities
    - Post-processing
  - RFI mitigation approaches (auto-flagging, subtraction, use of meta-data)

- **Questions to be addressed for the ngVLA**

  (1) Techniques to (potentially) use at each stage.
  - Based on types of RFI, algorithms and computing resources

  (2) Hardware/software design features and resources to plan for.

  (3) Demonstrate technical readiness of each approach.

- **Test/development/analysis plan**

  Ways to derive answers for (1), (2) by Q3 2018 and for (3) by 2020.

Discussion topic

- Is this plan OK ?

- What to add or subtract ?

Time/Freq characteristics of RFI (approximate / work-in-progress)

1 GHz	5GHz	10 GHz	20 GHz	50 GHz	100 GHz

Pulse :

Duration
< usec

DME

Short :

Duration
and
spacing :

Few usec
to msec

Bluetooth
- wearables
- car ent.

Wifi
- wearables
- car ent

Wireless usb
- wearables

WiGig / WHD
- wearables

Cow
rfid

Military
(F-35 comms)
Ku band

Traffic
radar (24)

Civilian aircraft

Military : WSMR, GPS
jamming, radar tracking,
F-35 crosslinks (L/C band)

SAR ground/airborne
(X, Ku)

SAR
L-band

Aircraft
with 5G

cell

cell

cell

Cell (60-76)

Car radar
(76-80)

Long :

Few sec
or more

Satellites : L, S, C, X, Ku bands

Space-Net
(Ku band)
OneWeb, SpaceX

Space-Net
(Ka band)
Boeing

Color code : Local RFI (~ few antennas) RFI on large fraction of array(airborne) RFI on entire array (satellite)

RFI types and mitigation options at different stages

Voltage Streams (antenna based) → **Correlation/ Detection (visibilties in the CBE)** → **Averaging for storage (post-processing)**

Voltage Streams (antenna based)

- Detect strongest RFI in real time
- Best for local RFI
- Catch impulsive broad-band RFI before the FFT

Correlation/ Detection (visibilties in the CBE)

- Intermittent RFI (usec-msec)
– Detection may benefit from some averaging
- RFI phase structure is preserved in visibilities (airborne, satellite, etc...)

Averaging for storage (post-processing)

- Time-persistent RFI that requires analysis along some other axes.
- Weak RFI that becomes visible only after averaging

RFI types and mitigation options at different stages

Voltage Streams (antenna based)

- Detect strongest RFI in real time

- Best for local RFI

- Catch impulsive broad-band RFI before the FFT

Detection : Median / MAD / MoM / Kurtosis / Threshold... w/wo FFT.

Action :
 - Add to RFI monitoring database but don't edit data

 - Remove/replace samples containing RFI (and adjust weight)

Correlation/ Detection (visibilties in the CBE)

- Intermittent RFI (usec-msec)

– Detection may benefit from some averaging

- RFI phase structure is preserved in visibilities (airborne, satellite, etc...)

Detection :
 - Autoflag on visibilities
 - Use prior information
 - Matrix decomposition (PCA)
 - Imaging (RealFast) / Peeling (– RFI source location)

Action :
 - Remove samples that go into the next integration step and adjust weight

 - Subtraction (subspace projection / PCA methods)

Averaging for storage (post-processing)

- Time-persistent RFI that requires analysis along some other axes.

- Weak RFI that becomes visible only after averaging

Detection :
 - Look at plots of data (no more)

 - Autoflag on visibilities (rflag, tfcrop, uvbinflag)

 - Use prior information for deterministic flags and tuning of autoflag methods (online flagcmds, database queries, tsys tables, cal solutions, etc)

Action :
 - Remove from further processing (flagging).

Plan for the next several months – by Q3 2018

(1) Test real-time RFI detection and database creation/use :

– A test observation with known short-duration RFI (DMEs), save visibilities at ~10ms.
– Run a realtime RFI detector and populate a database (prototype exists)
(– If possible, include O.Ojeda's kurtosis filter within A.Erickson's framework.)
– Use metadata from the database for post-processing flagging (python queries/scripts)

(2) Evaluate real-time RFI removal strategies :

– Simulate voltage streams and correlation (multiply-accumulate) to evaluate blanking vs replacement with zero/noise (python script, simulator from BYU for decorrelation ?)
– Evaluate computing/design requirements + analyse external efforts/experience

(3) Evaluate RFI subtraction methods

– Interface with BYU group to apply (simulation) results from their ngVLA community study to real VLA data taken in summer 2017 (awaiting ~6k funding)
– Evaluate computing requirements and feasibility + analyse other methods

(4) Autoflag methods (all stages)

– Analyse numerical/computational scaleability of current algorithms
– Identify areas where heuristics may need to change + new R&D focus

(5) Connect with RFI survey predictions to identify what methods need the most focus.

Some ideas for the next 1.5 yrs – by 2020

(1) Demonstrate intelligent use of meta-data :

– Use RFI monitoring information to implement heuristics for better deterministic flagging
– Demonstrate the use of RFI meta-data to tune autoflag algorithms
– Deploy the above for the VLA pipeline ?
– Explore : Feed back of RFI information into Observation Scheduling (RFI avoidance)

(2) Test real-time RFI flagging/blanking strategies :

– Implement in antenna station boards (and/or CBE)
  – Flagging on time series, 2D time/freq planes, uv-grid, etc)

– Demonstrate on VLA data containing known RFI (DMEs, comm signals, etc)

(3) Proof-of-concept of RFI modeling and subtraction methods

– Demonstrate something that works (start with existing high time res test data)
– Explore : Use high time res visibilities to calculate RFI location (eliminate RFI sources)

(4) Improve auto-flag methods at all stages

– Continue R&D on ways to increase robustness and minimize the need for tuning
  – robustness analyses (e.g. sensitivity of kurtosis filters to target spectral lines),
  – algorithmic improvements and autotuning

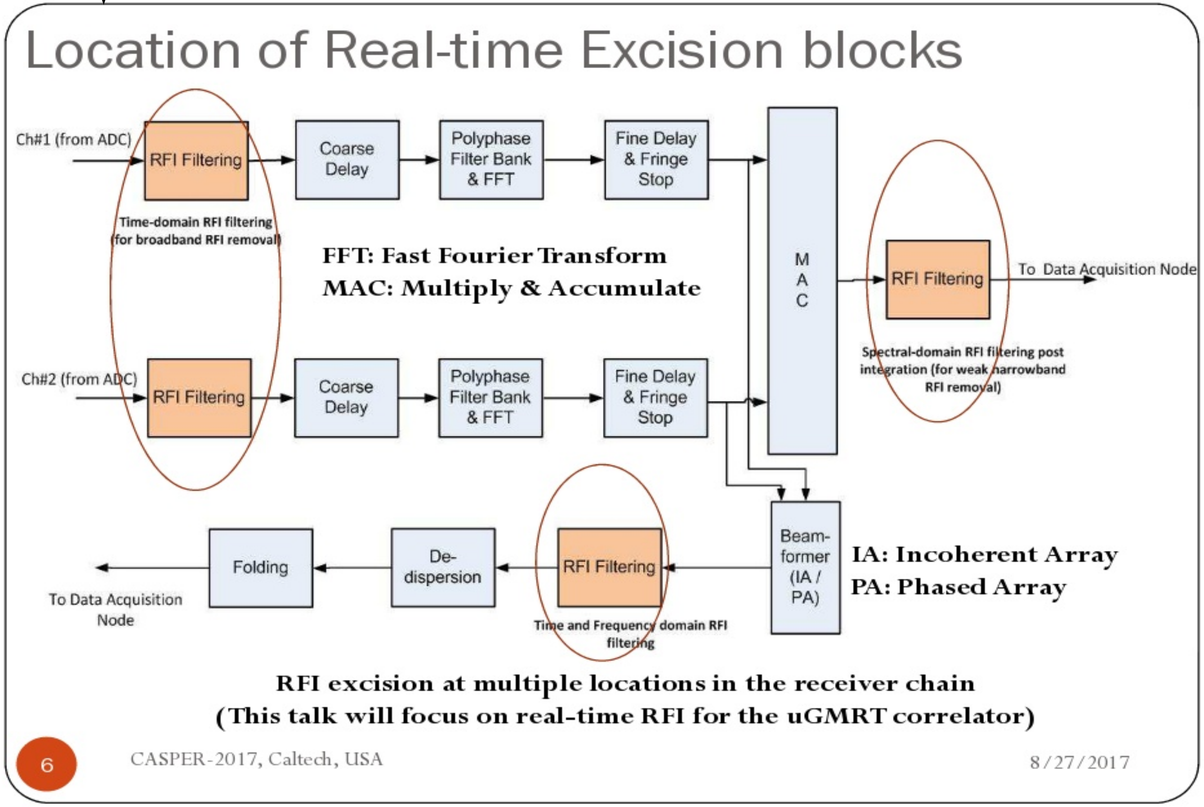
Extra Slides – Details...

RFI mitigation approaches : Real time RFI removal

Experiments at other observatories : replace RFI filled samples with noise, zeros, etc...

GBO Pulsar Backend : R.Prestage et al (Ref: URSI-NRSM 2018 talk)
- Real-time peak detection along time/freq
- Replace RFI filled samples with digital noise

uGMRT : K.Buch et al (Ref : https://arxiv.org/pdf/1710.08576.pdf)
- Real time RFI detection/excision at multiple stages
- Comparison of replacement by digital noise, zeros, and threshold value
  - They chose replacement by zeros.



Questions :

→ Other groups/results to evaluate ? SKA? LOFAR?

Todo (at NRAO)

– Flagging in “real-time”

=> Antenna boards : Leave samples out of the MAC step

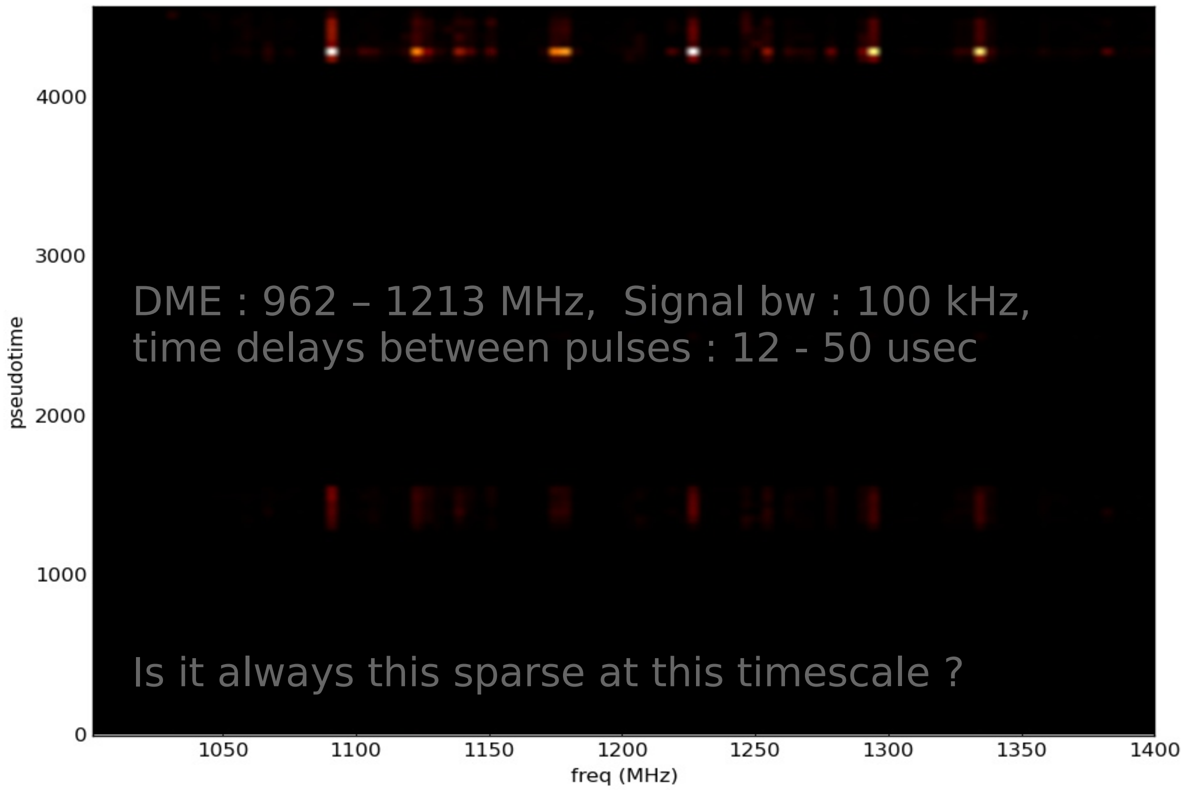
=> CBE : Apply RFI detection algorithms to high time-res visibilities

RFI mitigation approaches : Real time RFI removal

Experiments at other observatories : replace RFI filled samples with noise, zeros, etc...

GBO Pulsar Backend : R.Prestage et al (Ref: URSI-NRSM 2018 talk)
- Real-time peak detection along time/freq
- Replace RFI filled samples with digital noise

uGMRT : K.Buch et al (Ref : https://arxiv.org/pdf/1710.08576.pdf)
- Real time RFI detection/excision at multiple stages
- Comparison of replacement by digital noise, zeros, and threshold value
  - They chose replacement by zeros.



DME waterfall plot (from A.E.)

Questions :

→ Other groups/results to evaluate ? SKA? LOFAR?

Todo (at NRAO)

– Flagging in “real-time”

=> Antenna boards : Leave samples out of the MAC step

=> CBE : Apply RFI detection algorithms to high time-res visibilities

RFI mitigation approaches : Real time RFI removal

Very simple simulation

 – Flagging at high resolution can be very useful

 – Flagging on voltage streams can help reduce noise even for uncorrelated RFI

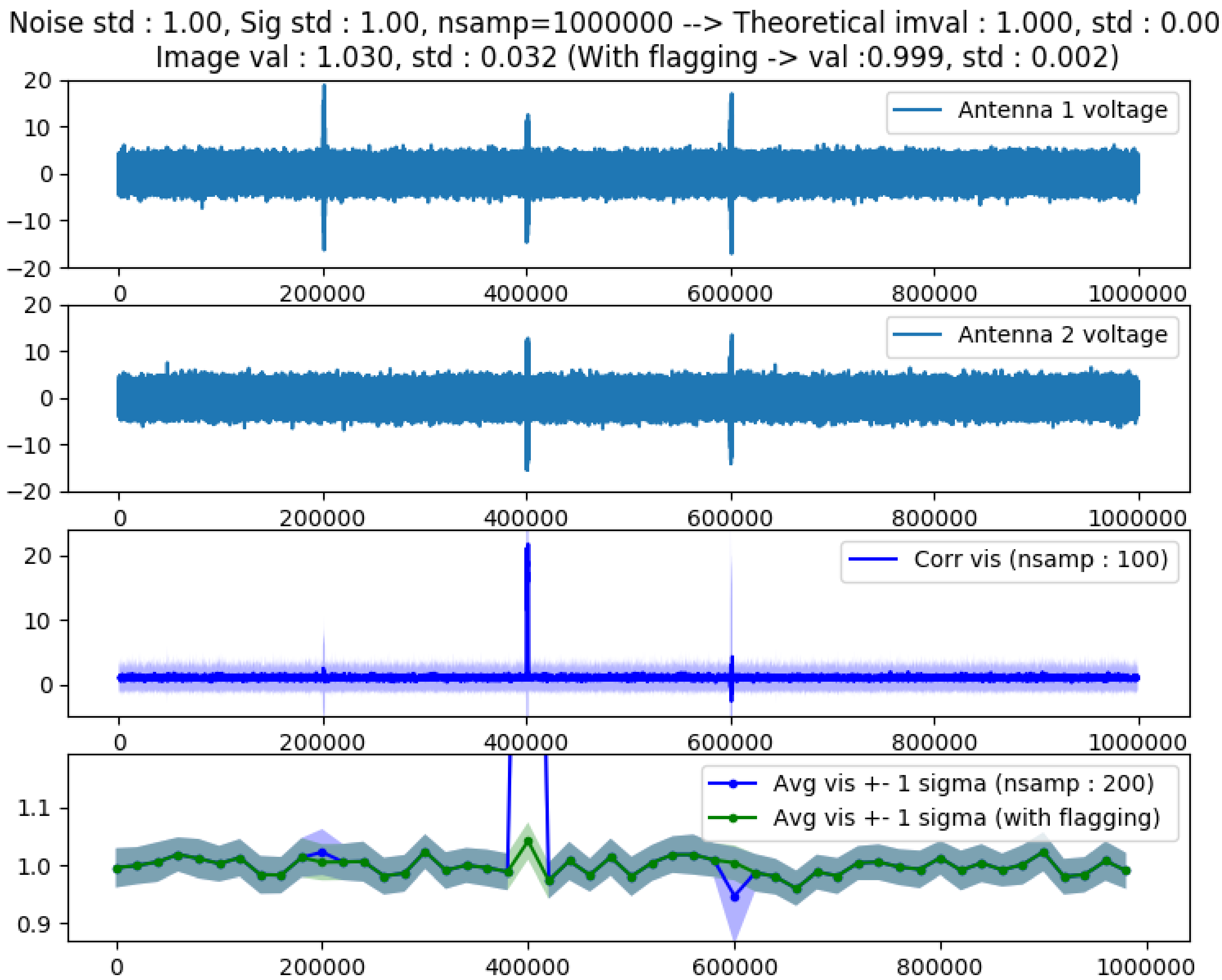
Pre-corr or Post-corr ?

– Short timescale and/or high power RFI

  => Pre-corr
  (Just a threshold)

– Longer timescale

  => Post-corr
  (can use stats accumulated during MAC step)



RFI mitigation approaches – Auto-Flag

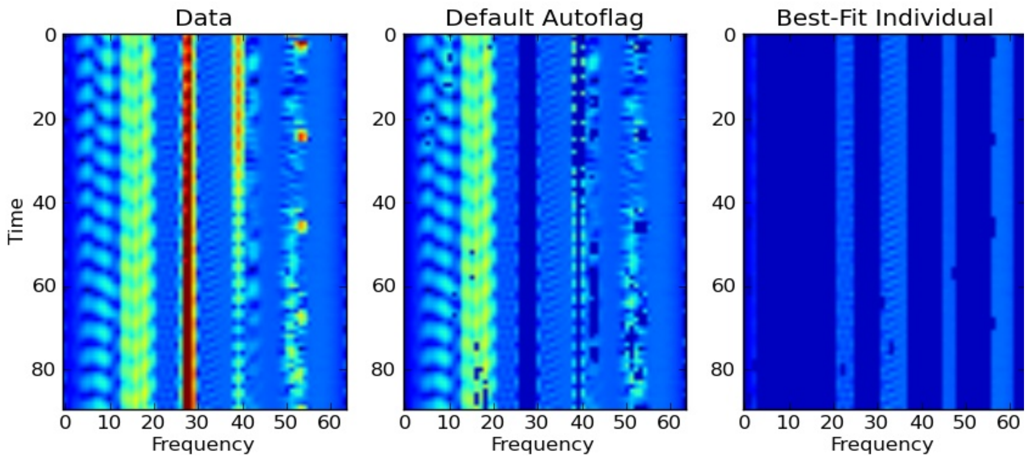
Autoflag :
- Various statistical filters for data streams (Median, MAD/MoM, Kurtosis, FT/threshold)
- RFLAG : Hierarchical statistical flagger across time/freq (E.Greisen)
- TFCROP : Outlier detection on the time-freq plane per baseline (U.Rau)
- UVBINFLAG : Outlier detection on the gridded uv-plane (K.Golap)
- AOFlagger : Outlier detection on the time-freq plane per baseline (A.O. (Astron))

=> All potentially work well but all need manual tuning for best results

To do :

(1) Use a-priori information about RFI to better tune these algorithms

(2) Algorithmic improvements : For example, automatic tuning (experimental) :
Auto-tune : A genetic algorithm that optimizes a flagging quality metric to derive 'best fit' sets of parameters for rflag/tfcrop. (U.Rau / B.Martins)



Work in progress :

- Run autonomously on a dataset
    - Select training subsets
    - Derive best-fit parameters
    - Run autoflag on all data
- Evaluate for use in pipelines

RFI mitigation approaches – Intelligent use of meta-data

RFI monitoring :

– Regular monitoring and documentation (D.Mertely et al)

– RFI database : Framework to tap voltage stream data, run peak detection algorithms, populate a database with RFI metadata (A. Erickson et al)

– Can plug in different detection algorithms (e.g. kurtosis filter : Ref : O.Ojeda)
– A scriptable query interface (plots, post-processing use, etc)

```
select o.id,at,fc,power,antenna,az,el
from observations o
join evla_rf_observations eo
join features_peak fp
where round(fc) = 1408
```

id	at	fc	power	antenna	az	el
1129	2016-10-13 17:49:05.546863-06	1408.3717	54.6825838621869	ea21	467.199027	31.77079
1628	2016-10-13 19:24:23.018826-06	1408.3337	66.1043710604892	ea24	244.763031	18.659935
1663	2016-10-13 19:32:48.962686-06	1408.49827	66.7330828433405	ea28	245.988693	21.091347
1685	2016-10-13 19:34:39.944485-06	1407.99509	66.0529598598333	ea19	245.815323	23.25943
2045	2016-10-13 20:45:25.186452-06	1407.6797	55.1536793906279	ea18	252.924164	35.071106
2509	2016-10-13 22:14:19.627664-06	1407.7601	55.7998185629984	ea21	262.202789	54.940567
2642	2016-10-13 22:29:37.128164-06	1408.0483	54.8150993445764	ea13	263.822937	55.304832
2755	2016-10-13 22:32:36.318104-06	1408.49579	56.5655637769771	ea21	263.988885	55.569516
2760	2016-10-13 22:33:18.908546-06	1408.3803	67.8603100327471	ea07	263.915649	56.25515
2839	2016-10-13 22:35:35.886739-06	1407.6797	55.0361618504793	ea18	264.8515	56.635384

Ready for preliminary tests for post-processing use.

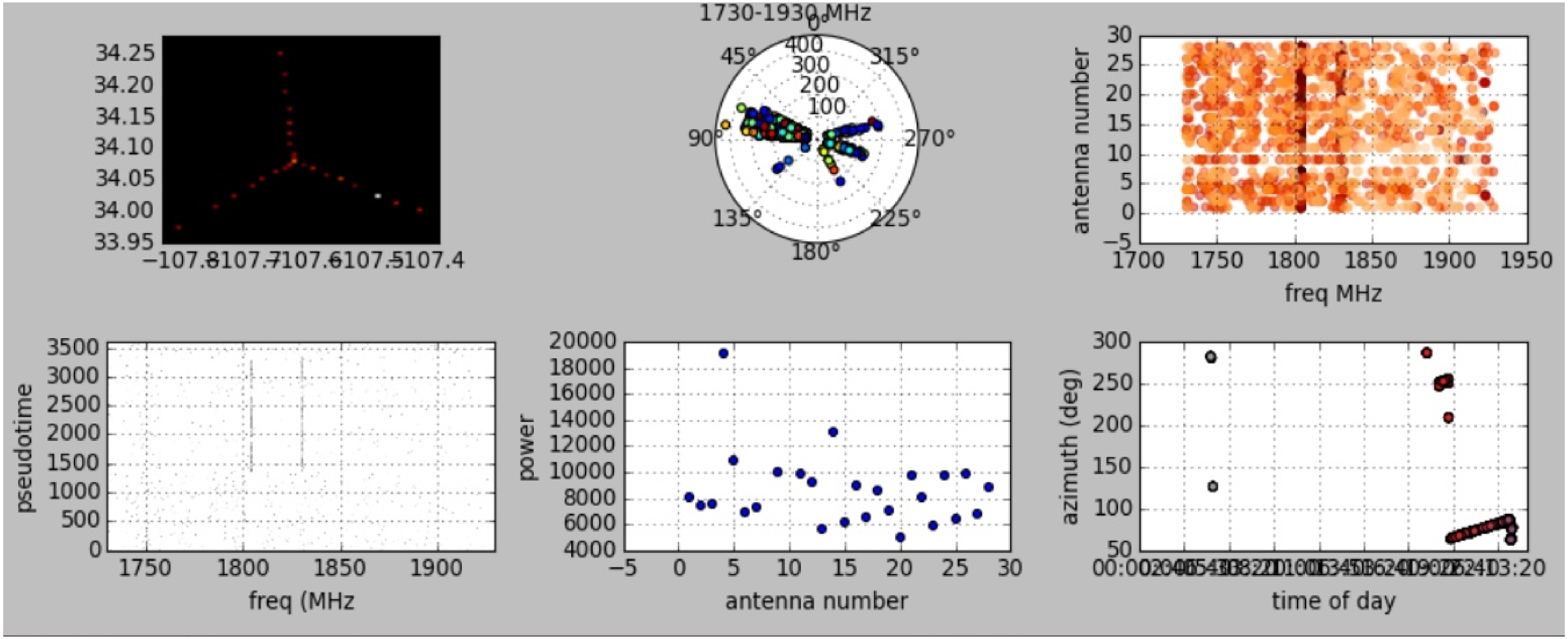
RFI mitigation approaches – Intelligent use of meta-data

RFI monitoring :

– Regular monitoring and documentation (D.Mertely et al)

– RFI database : Framework to tap voltage stream data, run peak detection algorithms, populate a database with RFI metadata (A. Erickson et al)

– Can plug in different detection algorithms (e.g. kurtosis filter : Ref : O.Ojeda)
– A scriptable query interface (plots, post-processing use, etc)
(Real time plots of RFI stats)



RFI mitigation approaches – Intelligent use of meta-data

Pipeline Heuristics : VLASS and ALMA (Ref L.Davis)

- A deterministic set of rules based on meta-data provided as auxiliary input.
- This is likely to suffice for a lot of RFI and reasons for bad data

Biggest improvements may come from better meta-data about intent and RFI

Examples

- Flag edges of scans and spws, apply QA0 flags (from pre-processing), online flags
- ALMA :
  - Look for outliers in Tsys and bandpass solution tables + apply flags to data
  - Look for outliers in fluxscale tables + implement flux transfer heuristics
  - Derive 2D views of the raw data to pick out deviant baselines, etc..
- VLA :
  - Deterministic heuristics similar to the above
  - Autoflag (rflag) with/without tuning.

To do :

- Use RFI database to inform mitigation choices at later stages (CBE, PP)
  - Deterministic flags as well as tuning of auto-flag methods
  - Decide what to turn on (and when).
    - E.g. RFI autoflag uncertainties shouldn't affect target science

- Think/learn about other machine learning / artificial intelligence approaches ?
- Explore RFI avoidance strategies (i.e. feed information back to the Observation setup)

RFI mitigation approaches – Modeling and Subtraction

(1) Several experiments have been done over the last 15 years (literature search)

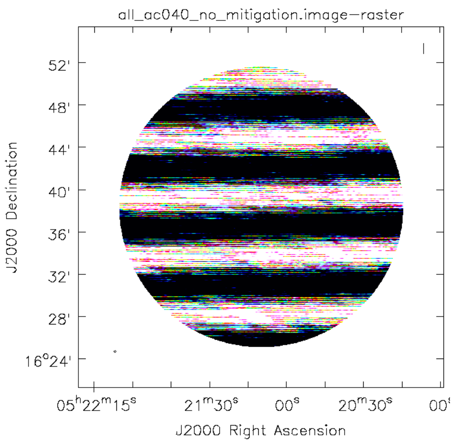
– Various matrix decompositions, RFI modeling by phase fitting, imaging at NCP, etc
– All methods work only sometimes, and none are in routine use.

(2) One example : Ongoing NRAO-BYU collaboration (with Brian Jeffs' group)

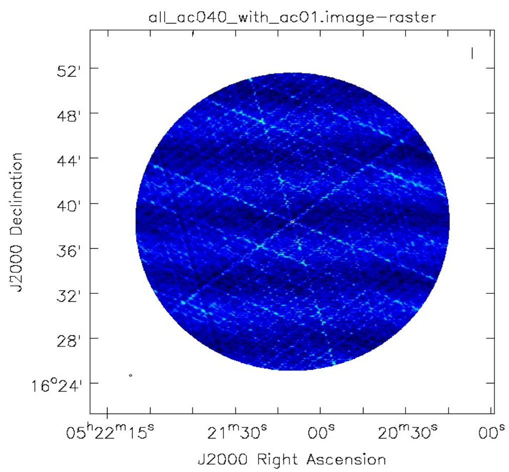
(a) 2016/2017 ngVLA community study on subspace projection methods
– Simulation-based proof of concept on data containing partially decorrelated RFI

(b) Summer 2017 (NRAO student project with U.R.)
– Obtained some C-config S-band data at 20msec/50kHz resolution (w/ K.S., V.D., B.B.)
– Basic demo of eigen-space projection to subtract RFI : Something worked......

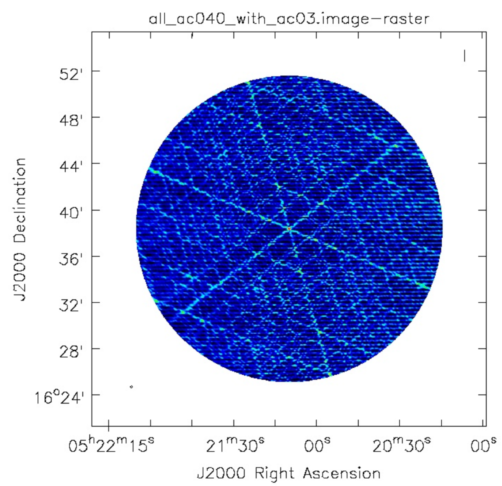
SIRIUS-XM : Projecting out eigen-vectors of the visibility matrix



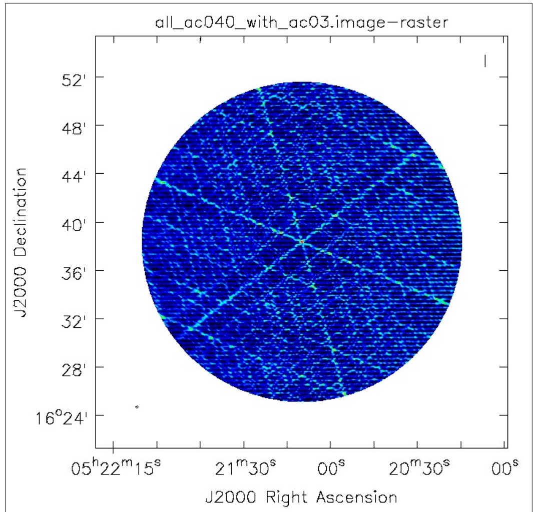
Original data



Project out 1 eigen vector



Project out 3 eigen vectors

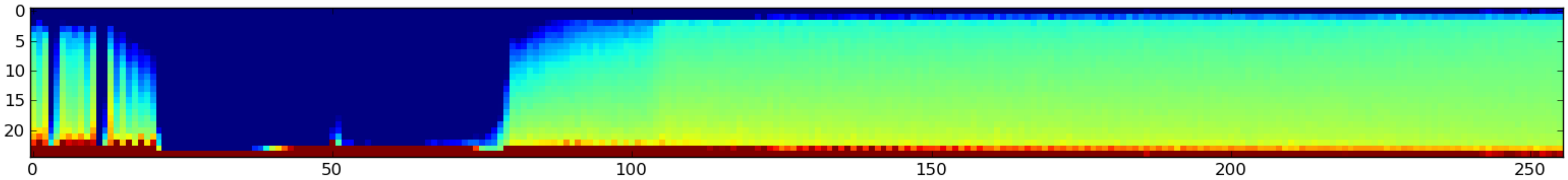


RFI - free channel

RFI mitigation approaches – Modeling and Subtraction

To do : More experiments on the Summer 2017 dataset.

Eigen value spectra for all 256 channels



=> Out of 256 channels, about 50 channels have strong RFI, 40 have moderate to weak RFI, a few channels may have more than one RFI source.

(1) At BYU : Average the data from summer 2017 data (@20msec, 50kHz) and apply algorithms from their ngVLA study to test robustness to decorrelation

- Awaiting ~6K funding for student to do this and write it up.
- They may want a new dataset in a larger config if averaging doesn't suffice.
- Study their algorithm for different RFI levels

[One key lesson : Matrix decomposition methods need autocorrelations]

(2) At NRAO : Try imaging/self-cal/peeling ideas

- Time/Freq resolution constraints for subspace projection methods are identical to those for all-sky imaging.