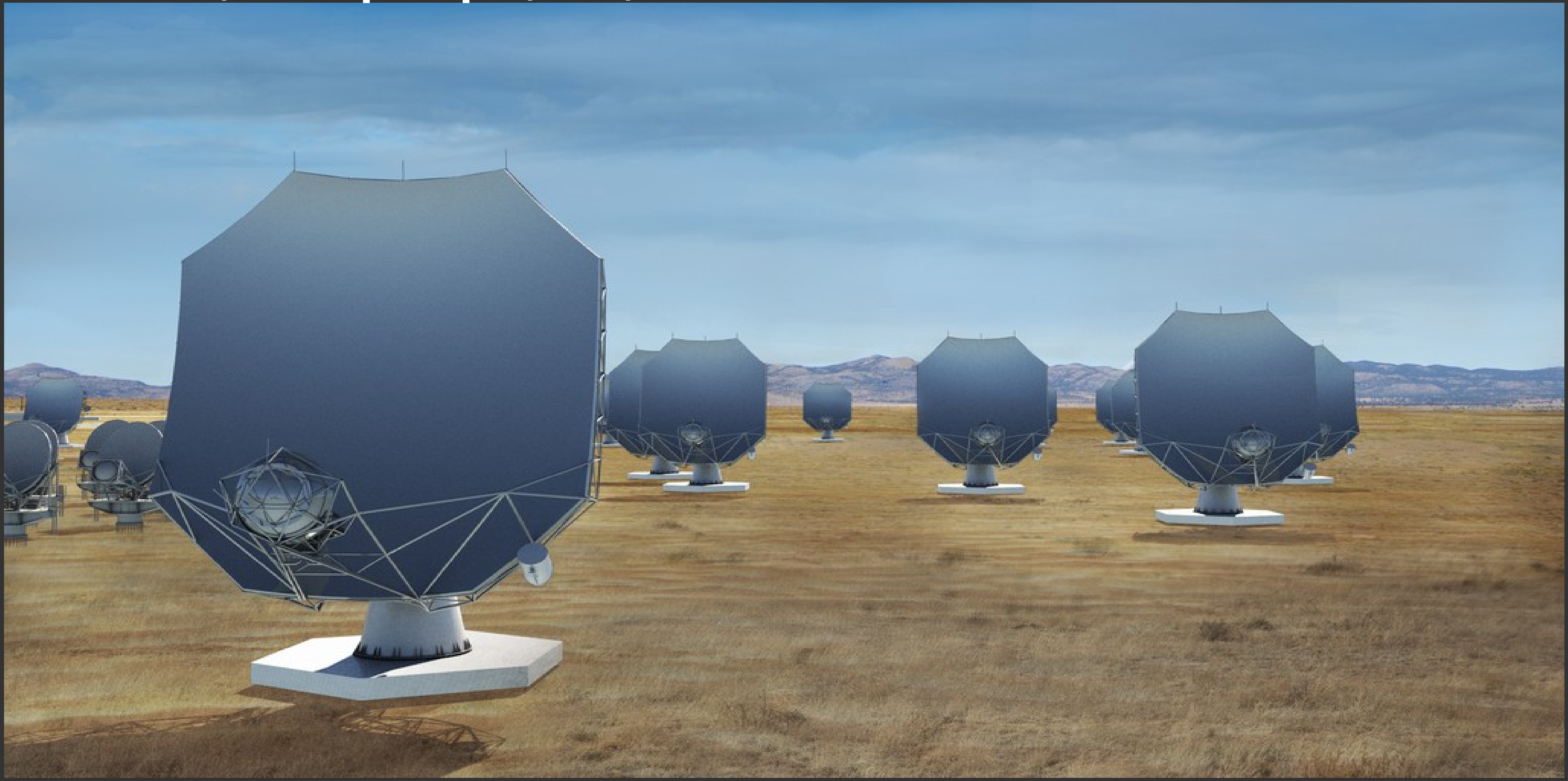
Use of Kokkos for imaging with radio interferometric telescopes

KUG 2023, Albuquerque, NM, Dec 13th 2023



S. Bhatnagar

M. Hsieh, F. Madsen, P. Jagannathan

Algorithms R&D Group,

National Radio Astronomy Observatory, Socorro, NM, USA

Introduction

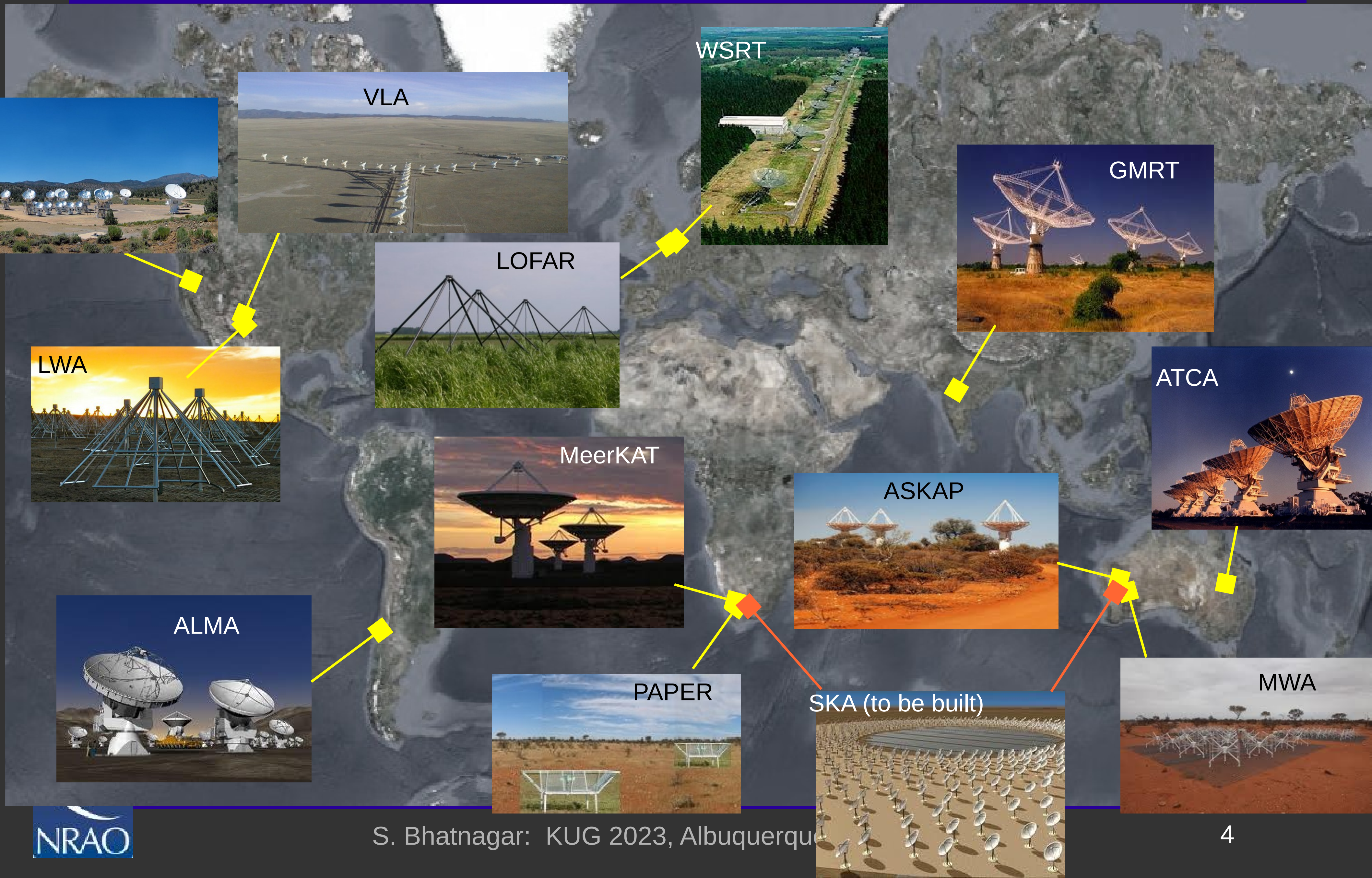
- Sanjay Bhatnagar
  - Algorithms R&D Group at the National Radio Astronomy Observatory

- NRAO: A NSF funded national observatory
  - Build and operate large radio astronomy facilities: VLA/ALMA/VLBA
  - Next-gen: ngVLA with 300 antennas spread across the US South-west
- Open source software for calibration and image reconstruction
  - Widely used in the RA community internationally
  - Runs on laptops, cluster, GPU/CPU,…,heterogeneous h/w

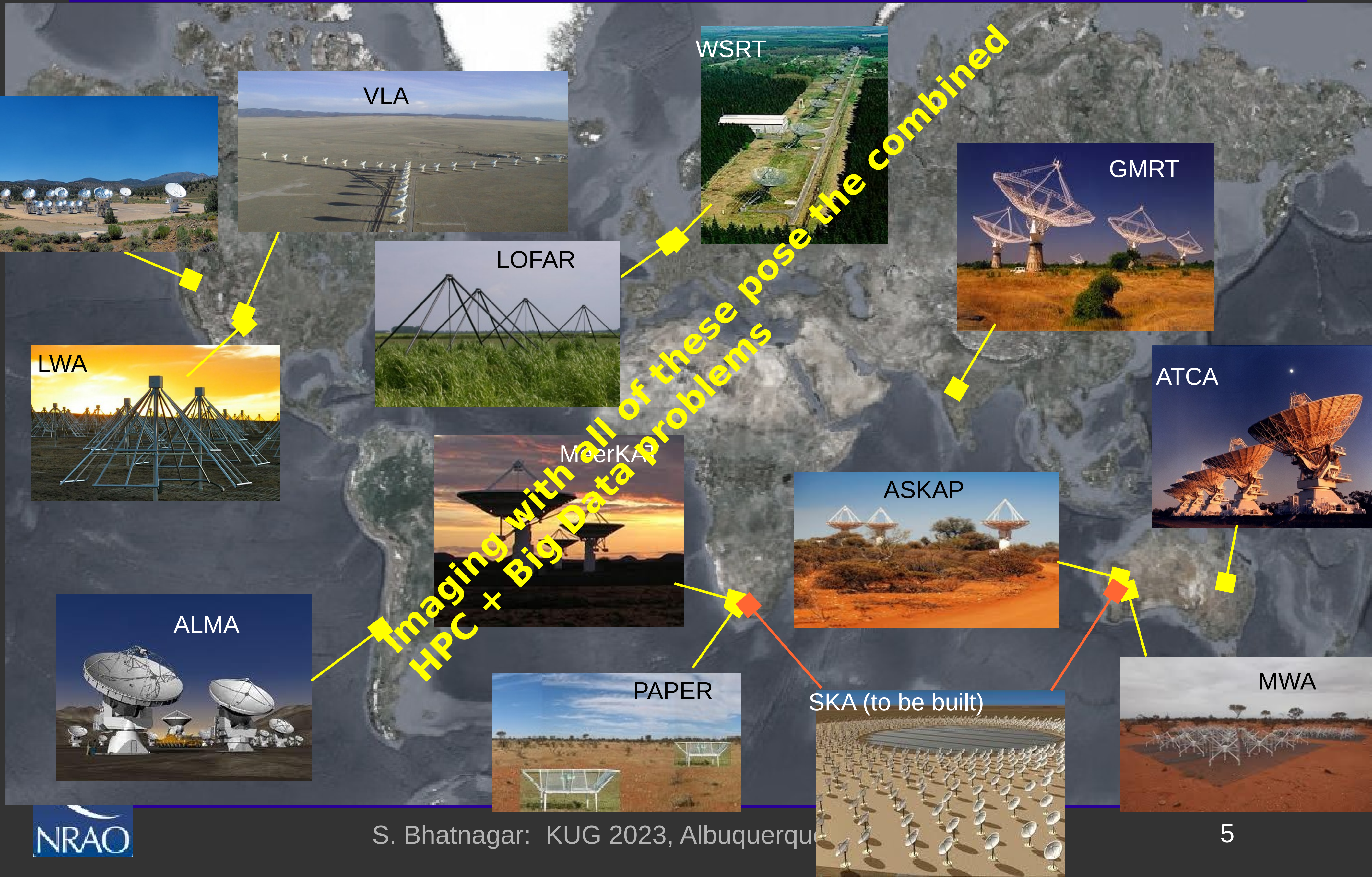
The Very Large Array (NM, USA)

- Very Large Array
- 27 antennas
- Antennas movable on rails
  1 – 27 Km radius
- Spread over
  27 Km radius
- Size of the “lens”
  30 Km
- Frequency range
  300 MHz – 50 GHz

Other RA Observatories in the world

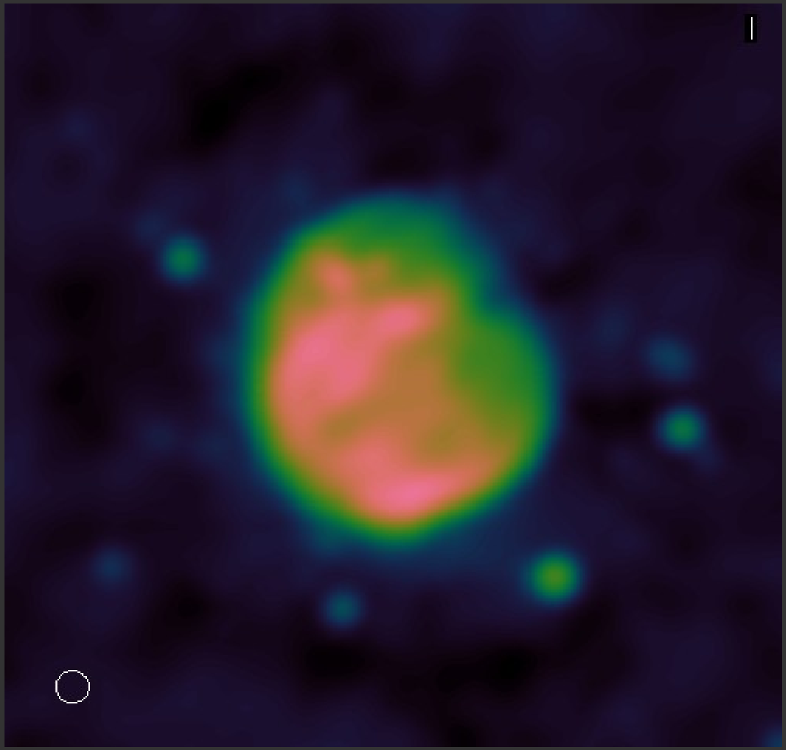


Other RA Observatories in the world



Aperture Synthesis Imaging: Why?

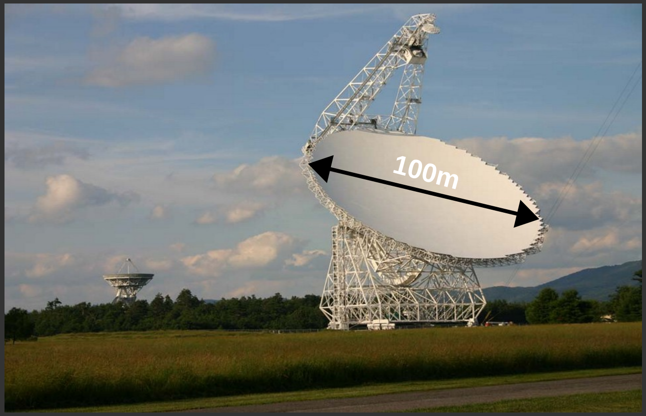
- Single dish Resolution too low for many scientific investigations
  - Limited collecting area + resolution limits sensitivity at low frequencies



Single dish resolving power

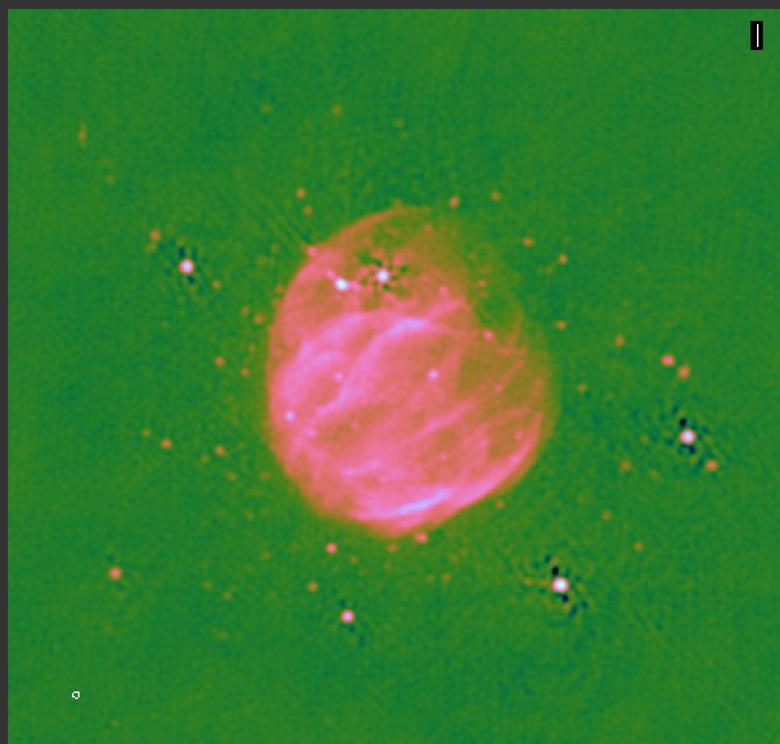
$$\frac{\textit{Wavelength}}{\text{Dish Diameter}}$$

Biggest steerable single dish = 100 m



Aperture Synthesis Imaging: Why?

- Single dish Resolution too low for many scientific investigations
  - Limited sensitivity/limits sensitivity at low frequencies



Synthesis Array resolving power

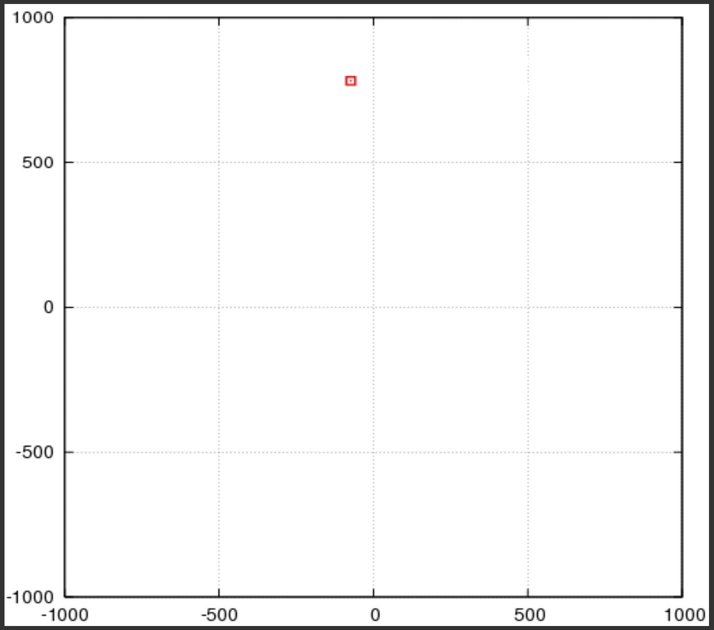
$$\frac{\textit{Wavelength}}{\text{Max. separation between antennas}}$$

Max. separation in VLA
= 35 km

Resolution: ~ 350x better

Aperture Synthesis Imaging: How?

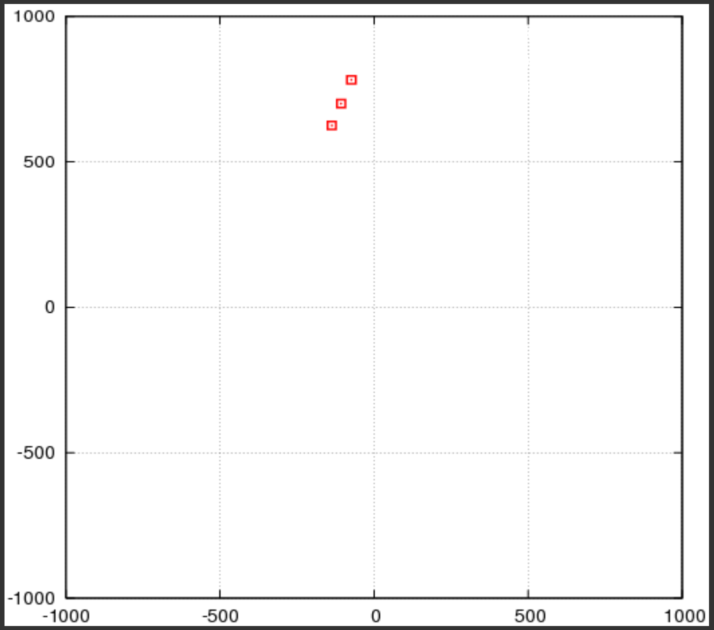
- An indirect imaging technique that collects data in the Fourier domain
  - Many antennas separated by 10s – 100s Km
  - Each pair of antennas measure **one** Fourier Component



  - Synthesized aperture equal to the largest separation between antennas

Aperture Synthesis Imaging: How?

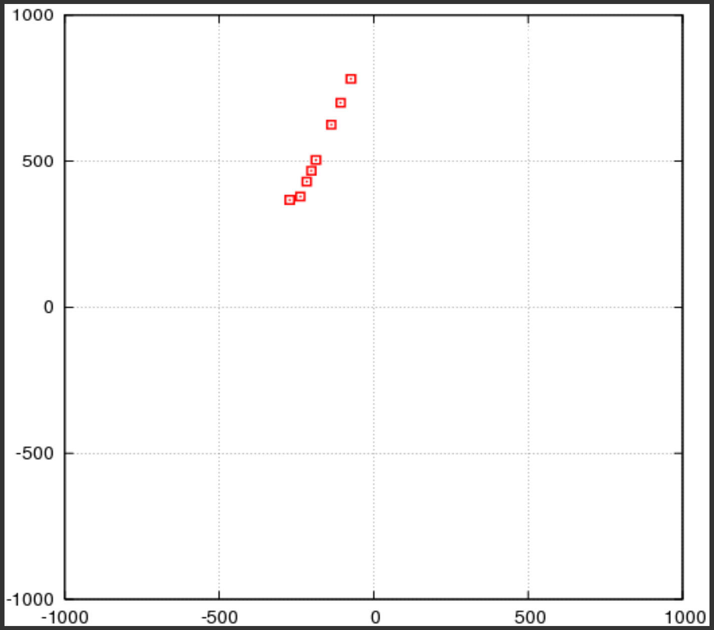
- An indirect imaging technique that collects data in the Fourier domain
  - Many antennas separated by 10s – 100s Km
  - Each pair of antennas measure **another** Fourier Component



  - Synthesized aperture equal to the largest separation between antennas

Aperture Synthesis Imaging: How?

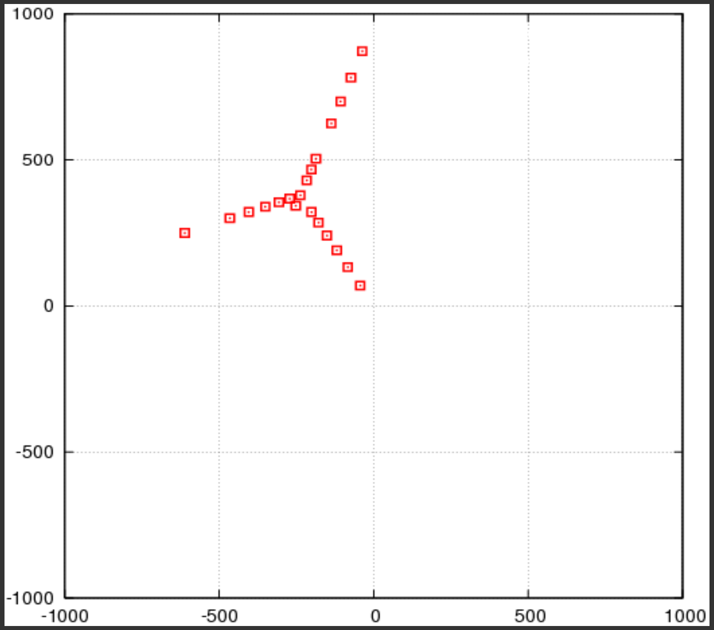
- An indirect imaging technique that collects data in the Fourier domain
  - Many antennas separated by 10s – 100s Km
  - Each pair of antennas measure **another (one)** Fourier Component

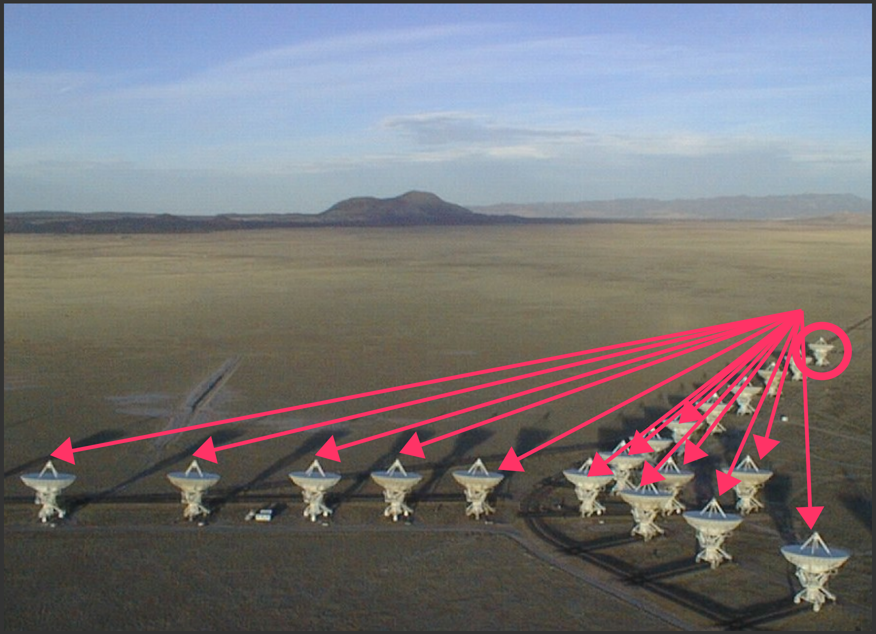


  - Synthesized aperture equal to the largest separation between antennas

Aperture Synthesis Imaging: How?

- An indirect imaging technique that collects data in the Fourier domain
  - Many antennas separated by 10s – 100s Km
  - **All** pairs with **one** antenna measure N-1 Fourier Component = **26**

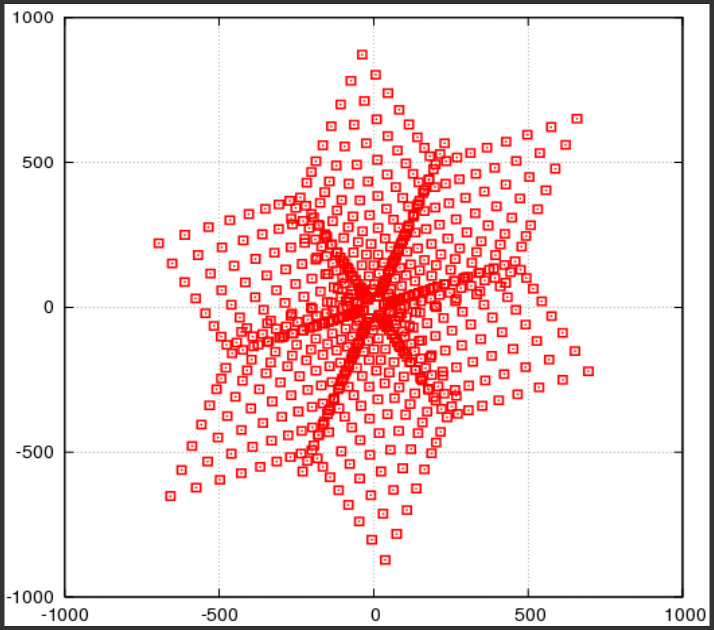


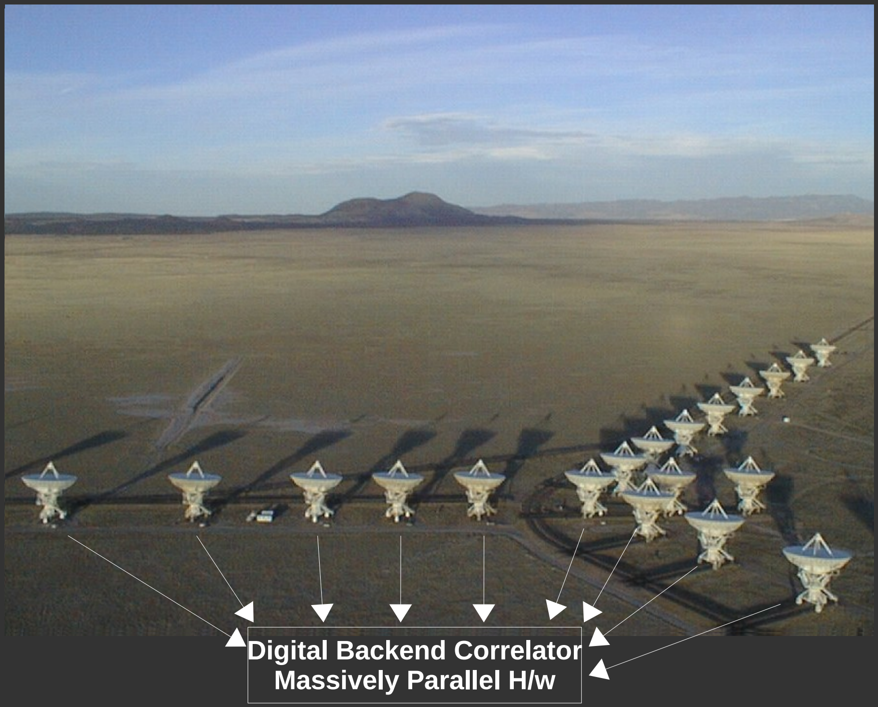


  - Synthesized aperture equal to the largest separation between antennas

Aperture Synthesis Imaging: How?

- An indirect imaging technique that collects data in the Fourier domain
  - Many antennas separated by 10s – 100s Km
  - **All** pairs with **all** antenna measure N(N-1)/2 Fourier Component = **351**

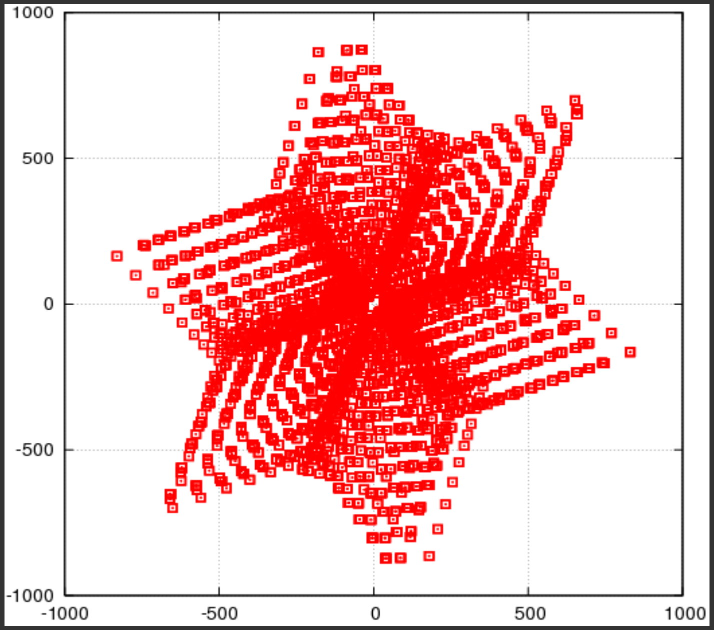


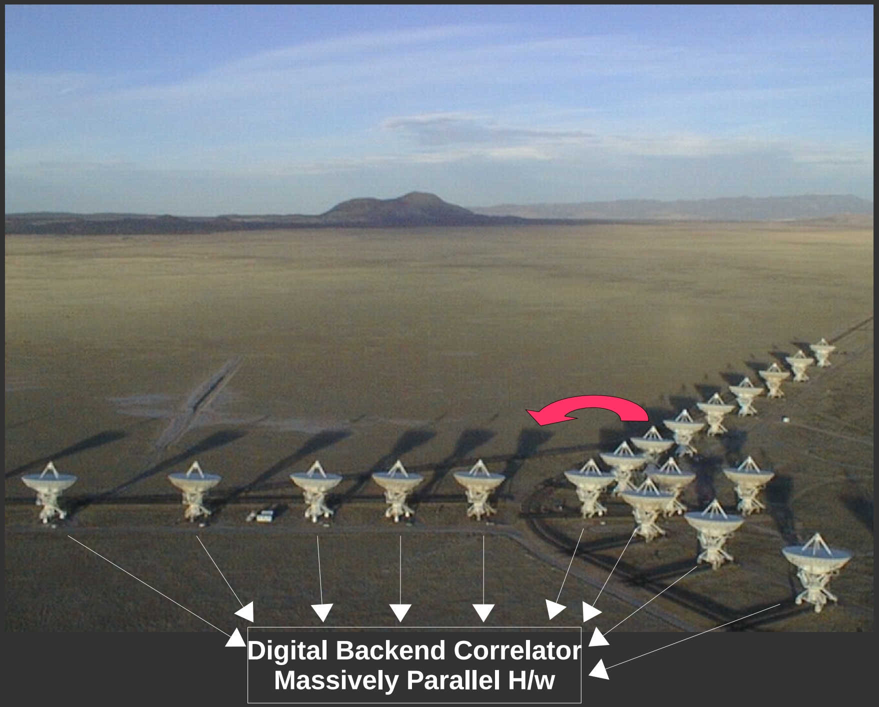


  - Synthesized aperture equal to the largest separation between antennas

Aperture Synthesis Imaging: How?

- Aperture Synthesis
  - Use **Earth Rotation Synthesis** to fill the Fourier plane
  - **All** pairs with **all** antenna measures N(N-1)/2 Fourier Component
  - Measure N(N-1)/2 x 2 Fourier components over 2 integration time = **702**

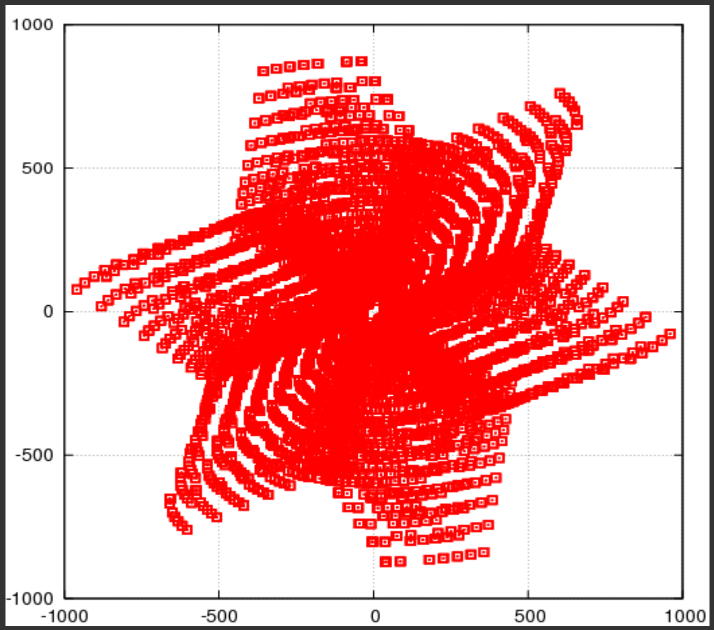


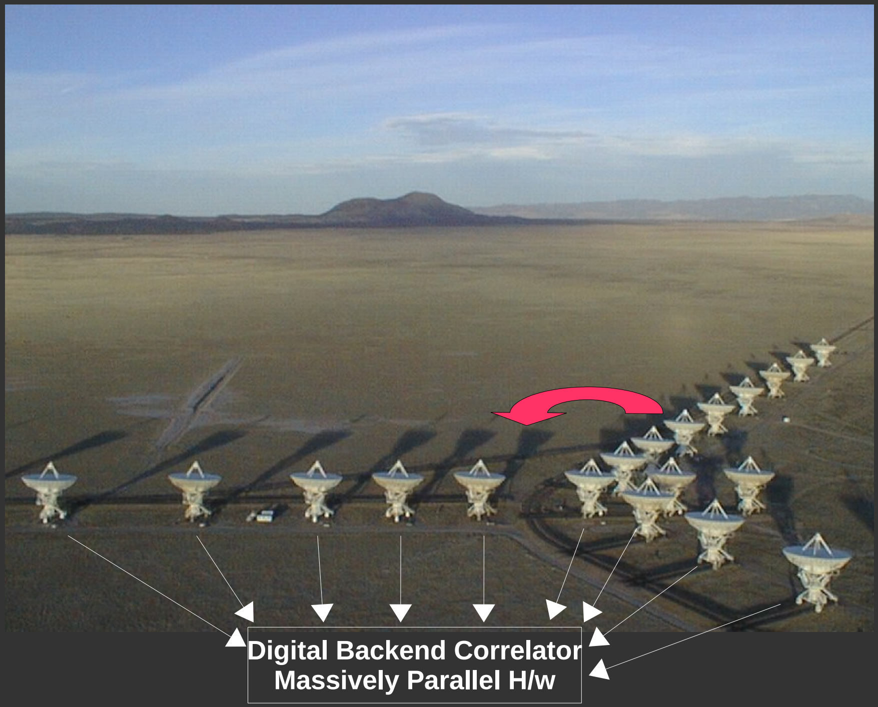


  - Synthesized aperture equal to the largest separation between antennas

Aperture Synthesis Imaging: How?

- Aperture Synthesis
  - Use **Earth Rotation Synthesis** to fill the Fourier plane
  - **All** pairs with **all** antenna measures N(N-1)/2 Fourier Component
  - Measure N(N-1)/2 x 10 Fourier components over 10 integrations = **7020**

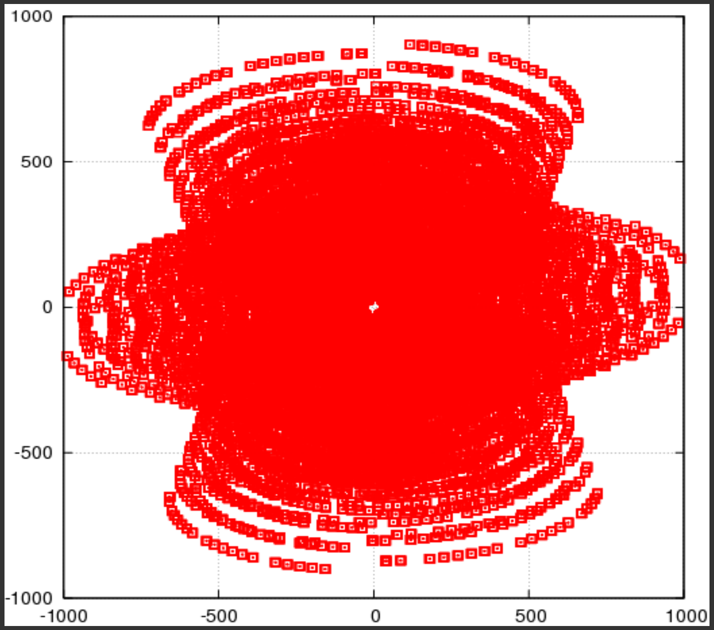


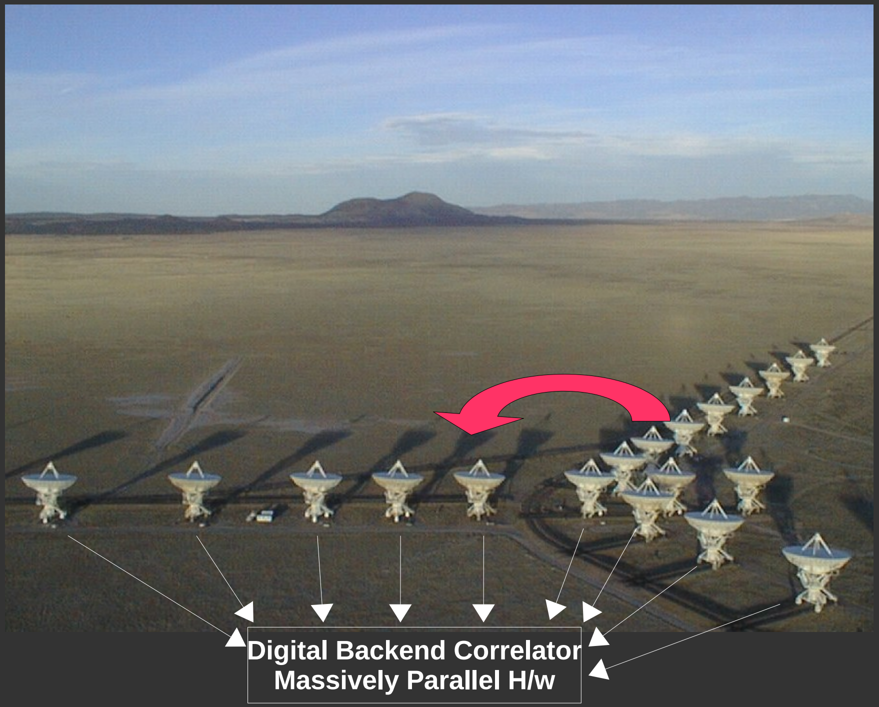


  - Synthesized aperture equal to the largest separation between antennas

Aperture Synthesis Imaging: How?

- Aperture Synthesis
  - Use **Earth Rotation Synthesis** to fill the Fourier plane
  - **All** pairs with **all** antenna measures N(N-1)/2 Fourier Component
  - Fourier Components measured over 10 hr: **O(10^{12-15})**

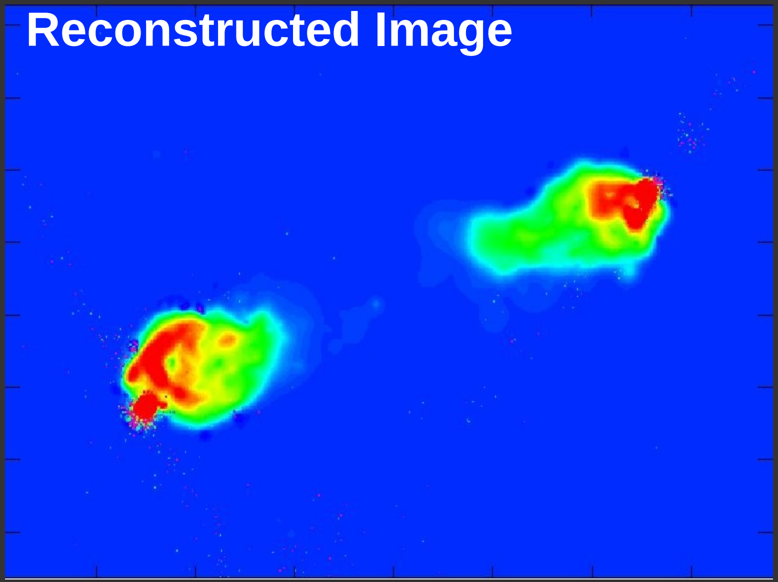




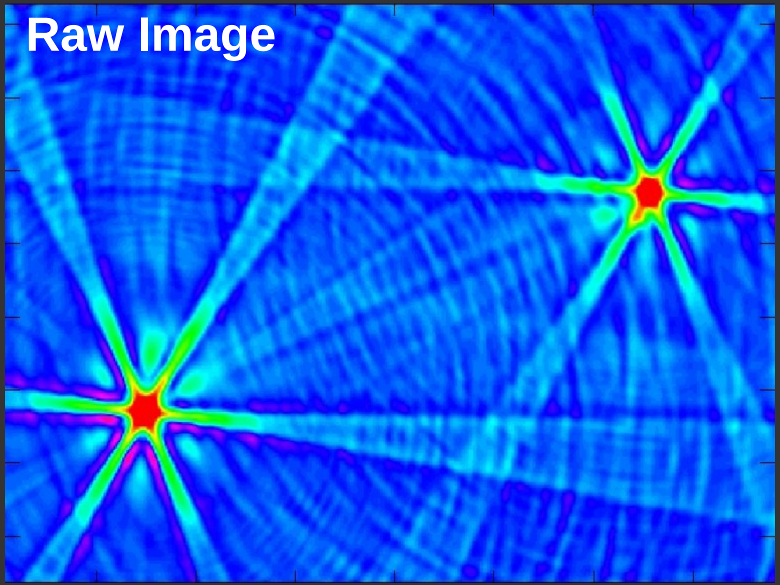
  - Data Size: 10s – 100s TB now Up to Exa Bytes for SKA-class telescopes
  - Data not on a regular grid.

Interferometric Imaging

- Raw image (FT of the raw data) is dynamic range limited



Dynamic range: > 1 : 1000, 000

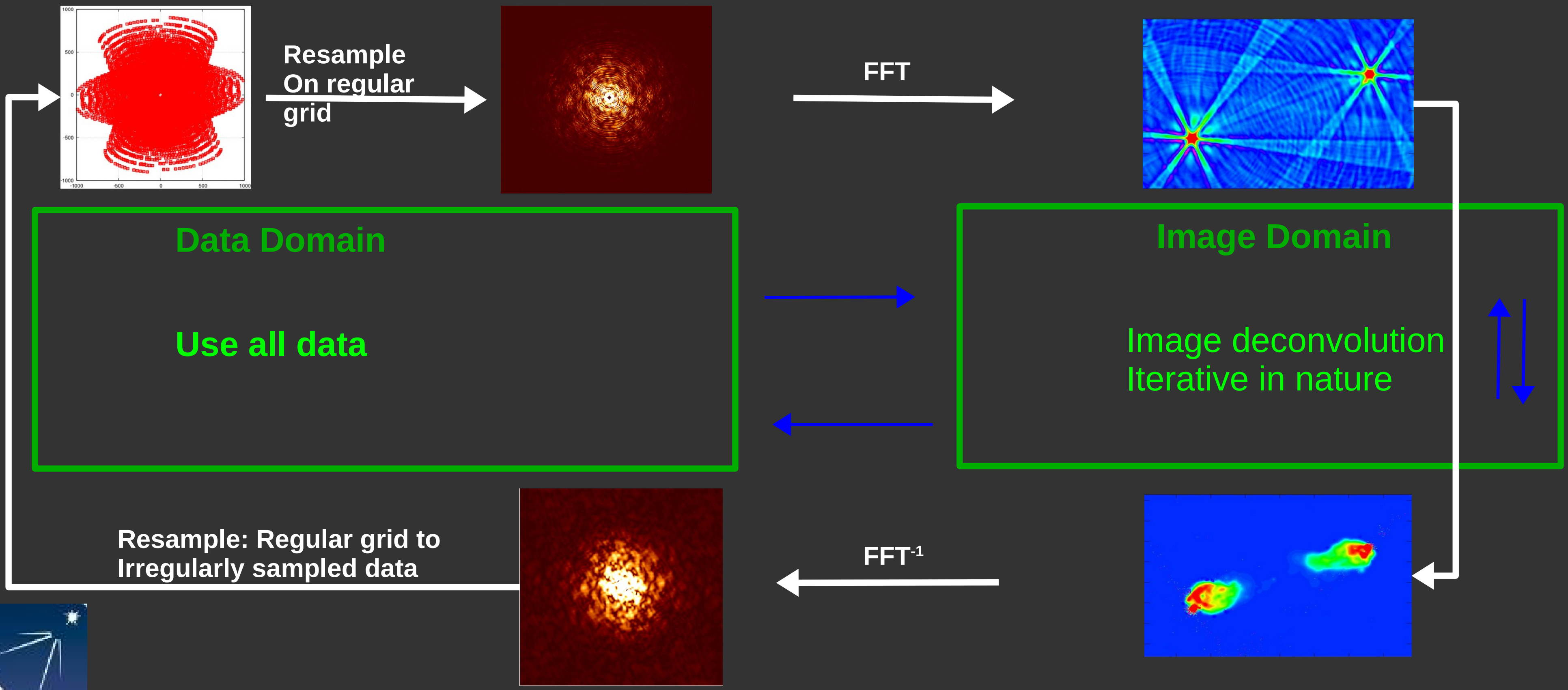


Dynamic range: 1 : 1000

- Processing: Remove telescope artifacts to reconstruct the sky brightness
- Image reconstruction is a High-Performance-Computing-using-Big-Data problem

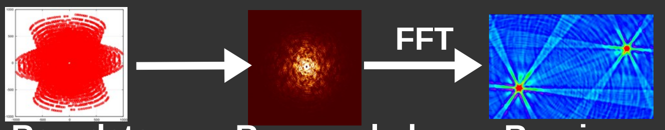
The Computing Problem

- Basic computing steps
  1. Use FFT to transform to the image domain: Gridding + FFT
  2. Image-plane deconvolution of the PSF : Search and subtract on images
  3. Inverse transform to the data domain: De-gridding + Inv. FFT

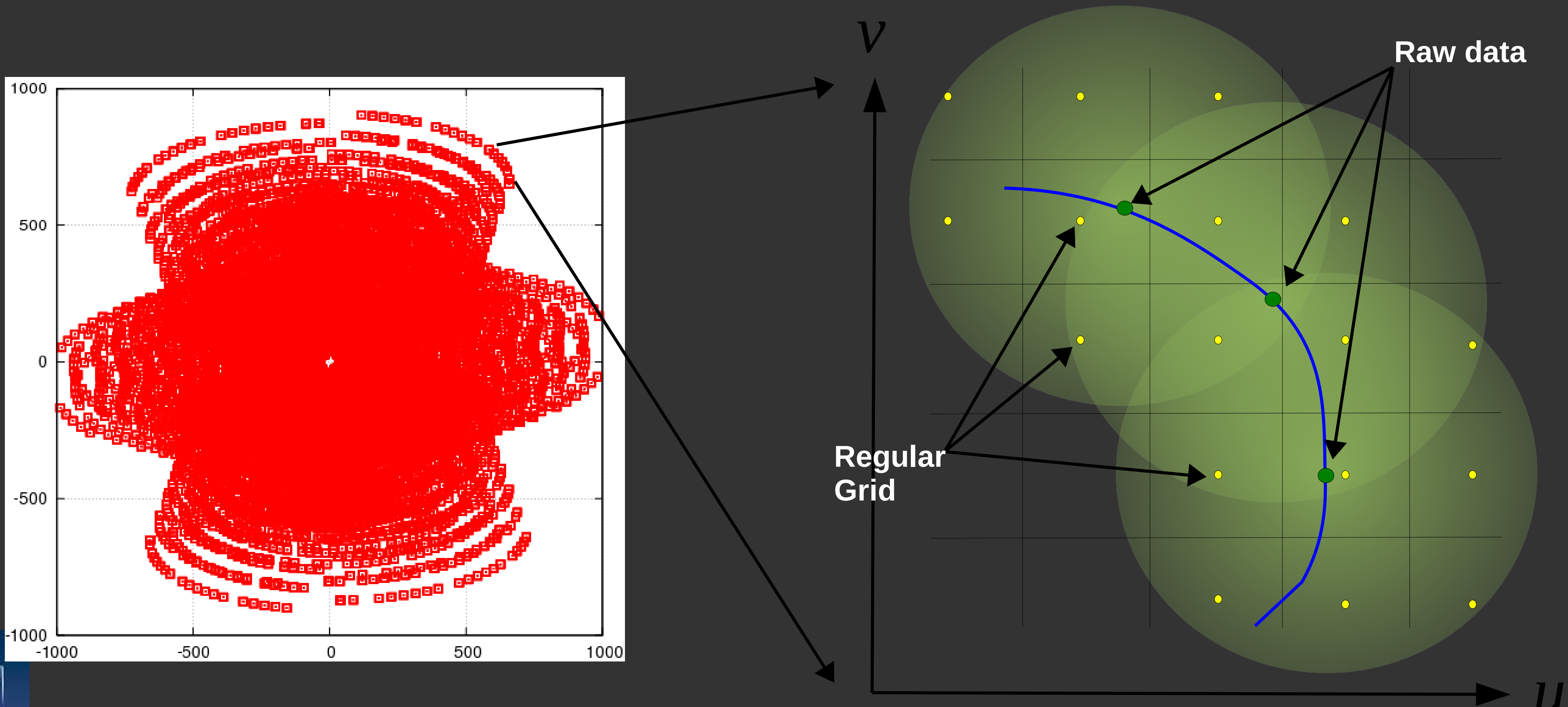


The Computing Problem: Why Gridding?

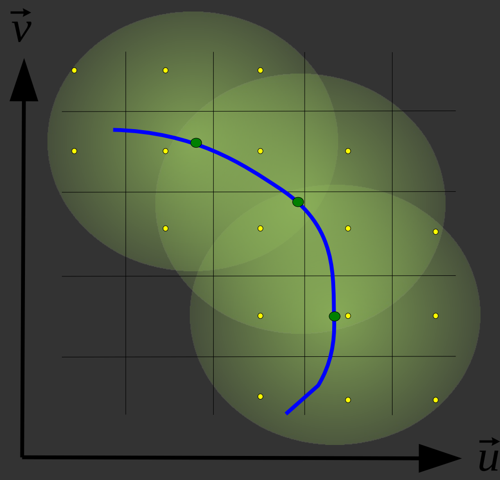
- Raw data is not on a regular grid
  - FFT require re-sampling on a regular grid



NU-FFT: But with specialized kernels



Computing requirements



- N_{data} x N^2_{CF} x Gridding FLOP + overheads
- ngVLA: $O(10^{13-14})$ x (10x10) x ...= ~50 PFLOP/s
- SKA: $O(10^{15})$ x ... = ~ExaFLOP

- HPC + Big-Data
  - Continuous data flow (24x7 observing)
  - PFLOPS / ExaFLOPS to keep-up with the data rates
  - 100s of Tera Bytes for a typical observing session

- Computing needs to be efficient and 24x7
  - Not a one-shot experiment on a homogeneous super-computer

- Requirement: Seamless computing 24x7 on a heterogeneous cluster

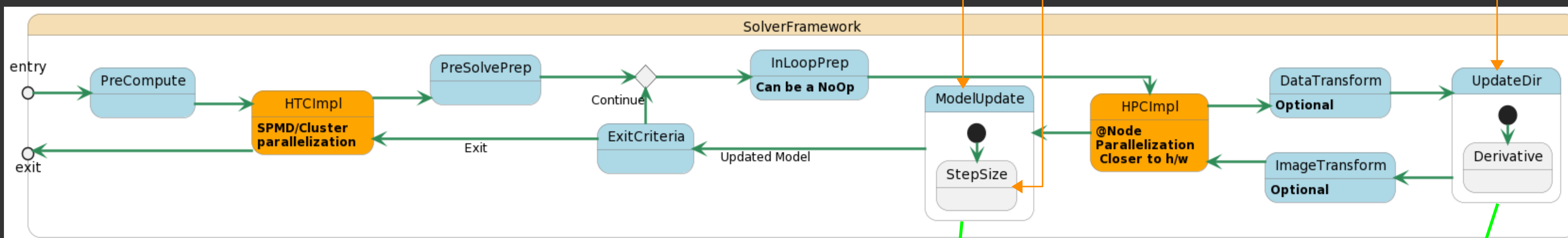
Algorithm Architecture: Components view

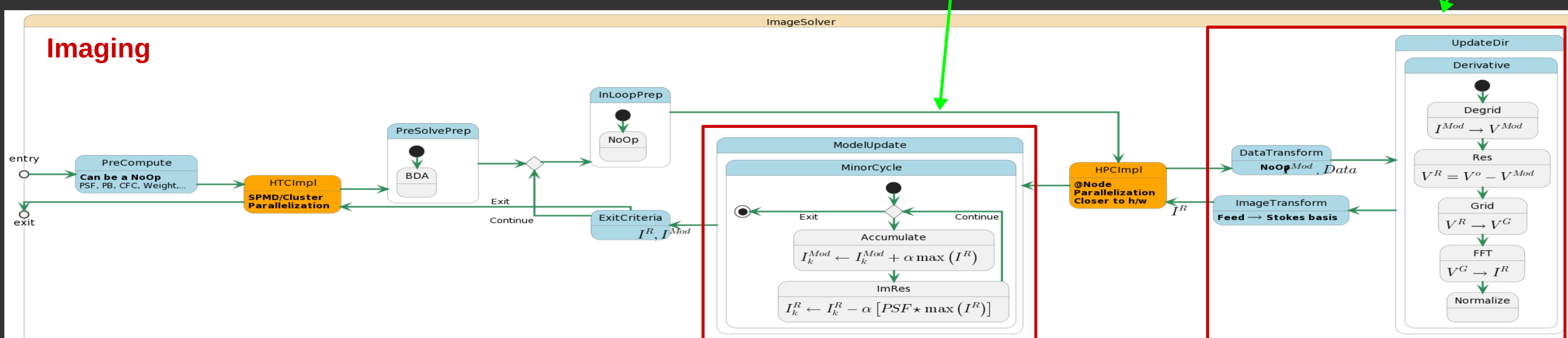
$$P_j^{k+1} = P_j^k + [H]^{-1} f\left(\frac{\partial \chi^2}{\partial P_j^k}\right)$$

a.k.a. the "Major Cycle"
Compute Residual Image

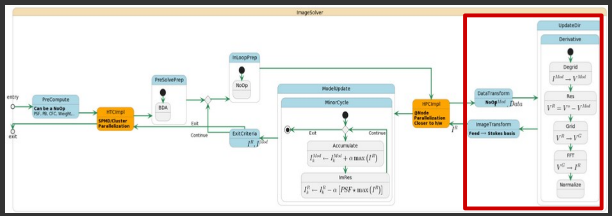
a.k.a. the "Minor cycle"
Update Model Image

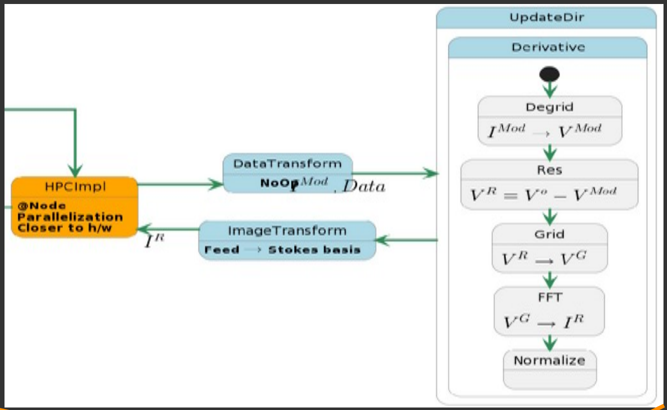
a.k.a. the "Loop gain"

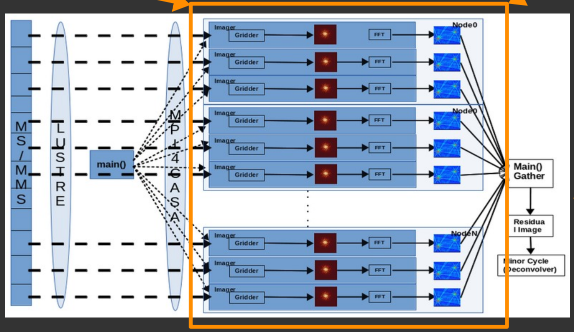


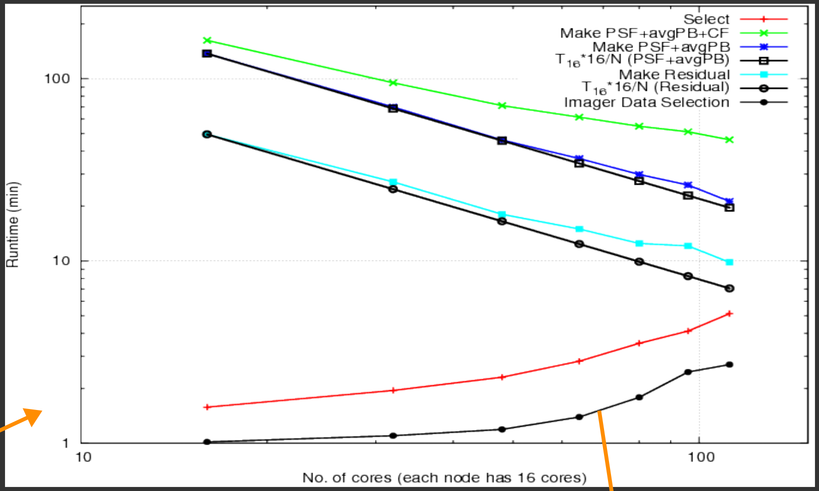


Scaling: On multi-CPU/cores hardware





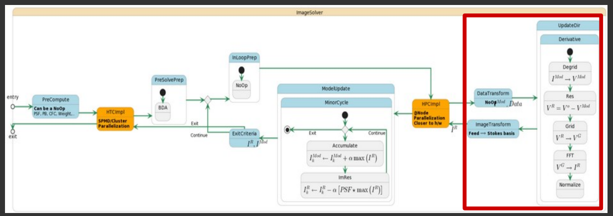


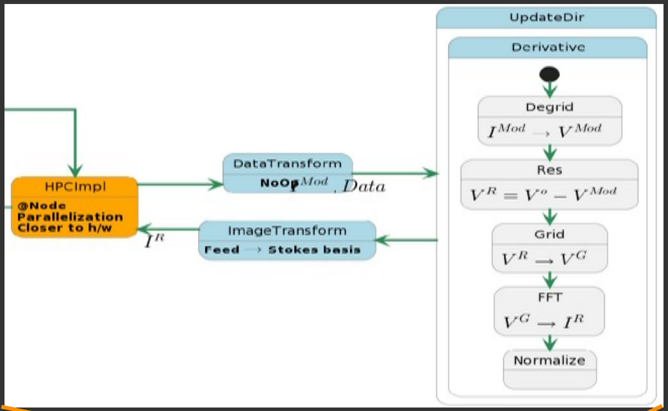


- Data scatter overheads

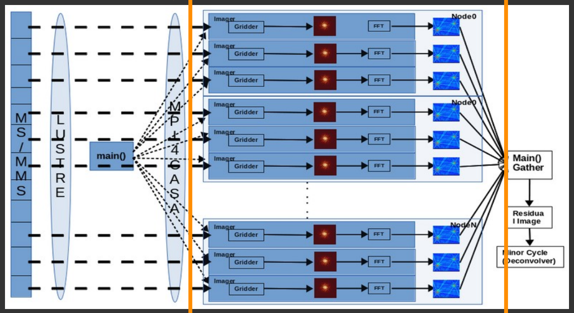
ngVLA would need O(Million)-way parallization!

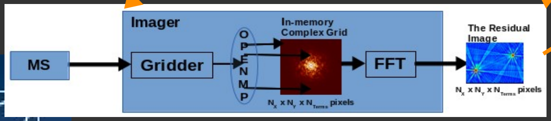
Scaling on GPU: Using Kokkos

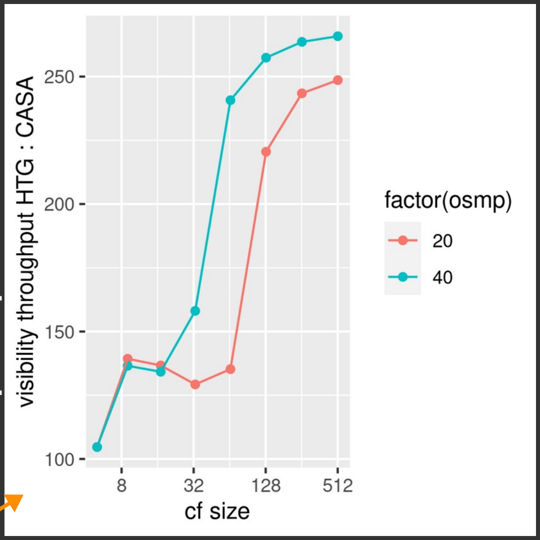




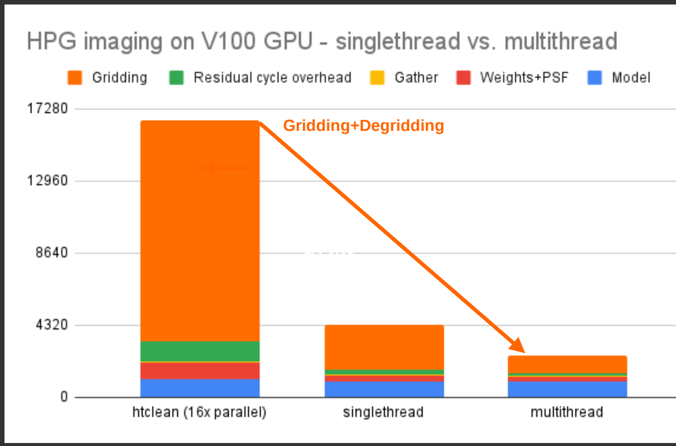
Complexity reduction







ngVLA Computing Memo #5, #7

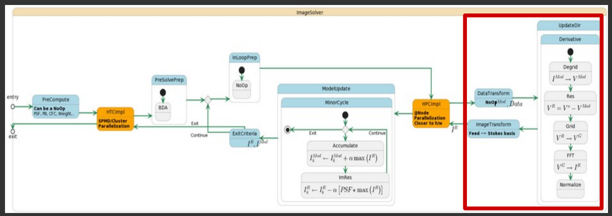


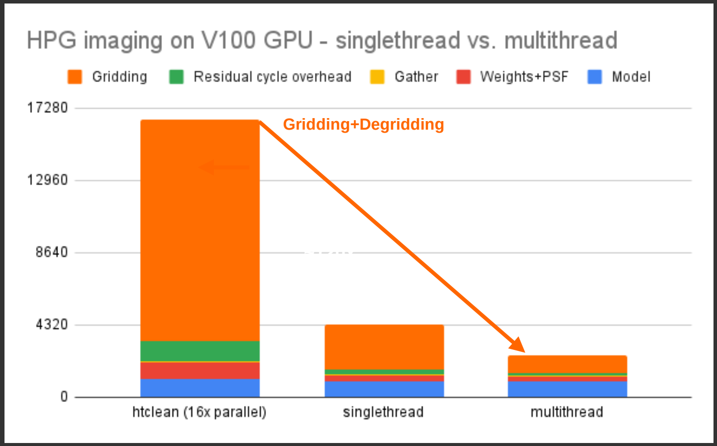
Multi-core CPU | Single CPU-core +GPU | Multi CPU-cores +GPU

ngVLA would need O(10^3)-way parallization!

Scaling in real-life

- A gridder on a GPU using Kokkos (NGVLA Memo #05)
- What does it mean in real-life application?
  - 200-pointing wide-band mosaic: 7-10 days vs 2.5hr

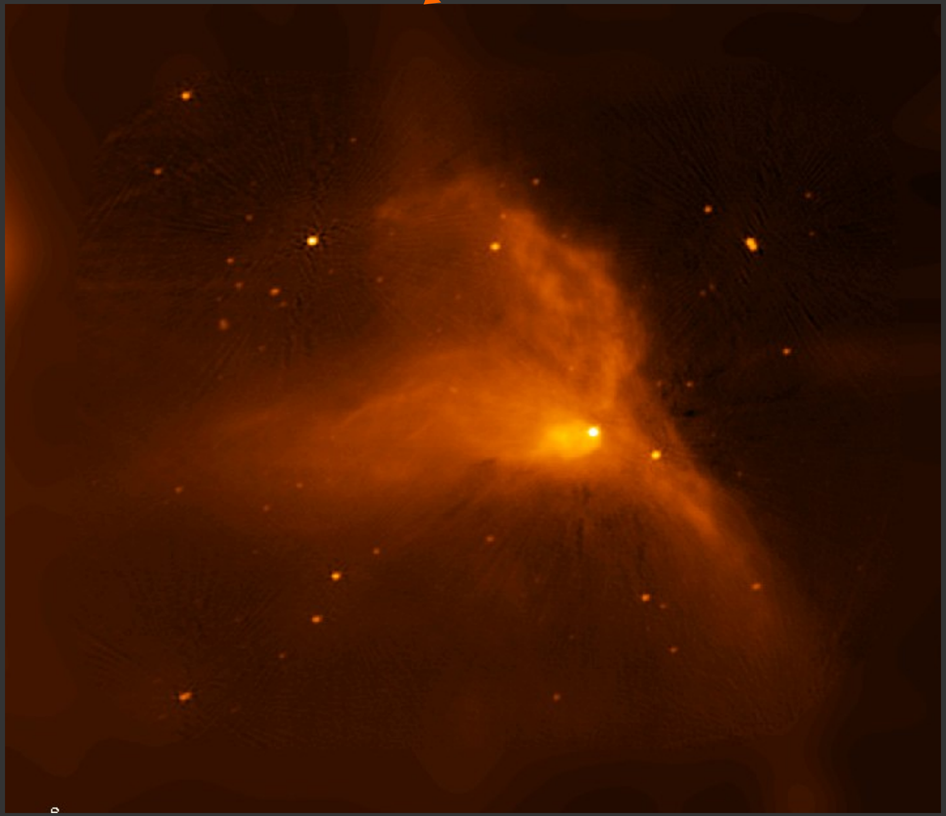




Multi-core CPU

Single CPU-core +GPU

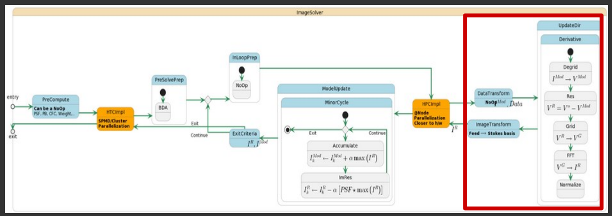
Multi CPU-cores +GPU



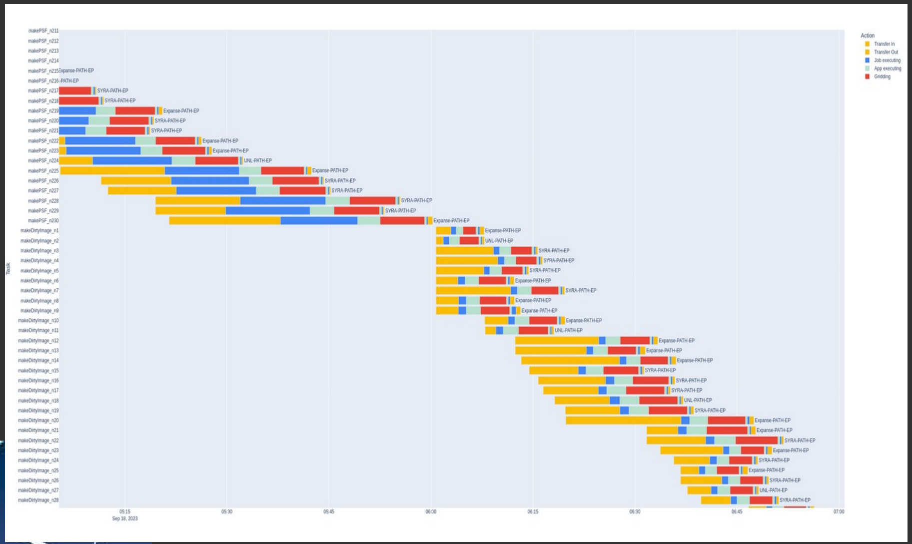
ngVLA would need O(10^3)-way parallization!

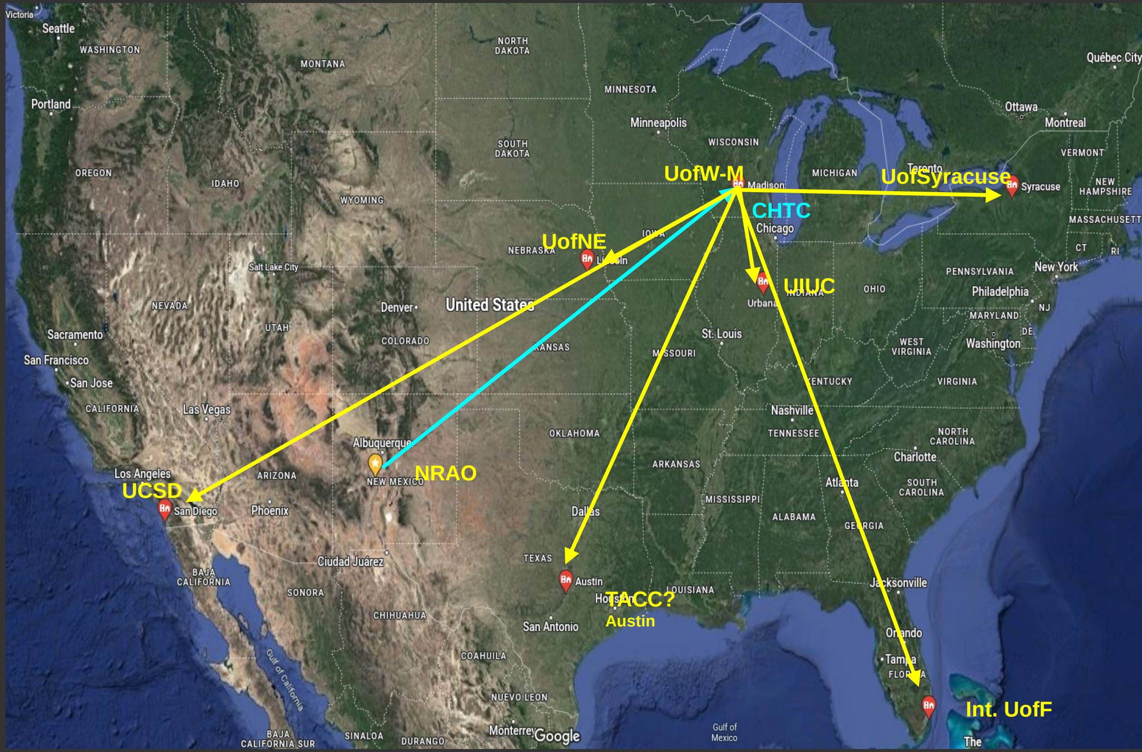
Scaling: On Wide-area network (OSG)

- <u>Distributed High Throughput Computing:</u>
  - Center for High Throughput Computing, U of W-M.
    - » PATh: A GPU cluster at a national scale
  - AWS: CPU cluster



- Opportunistic computing + Edge-caching
- <u>Work in progress</u>
  - Currently effort is resource-limited!
  - More human and computing resources
  - International resources

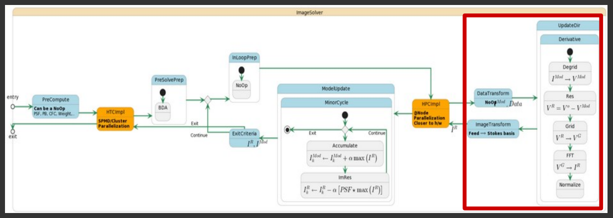




Madsen, et al.

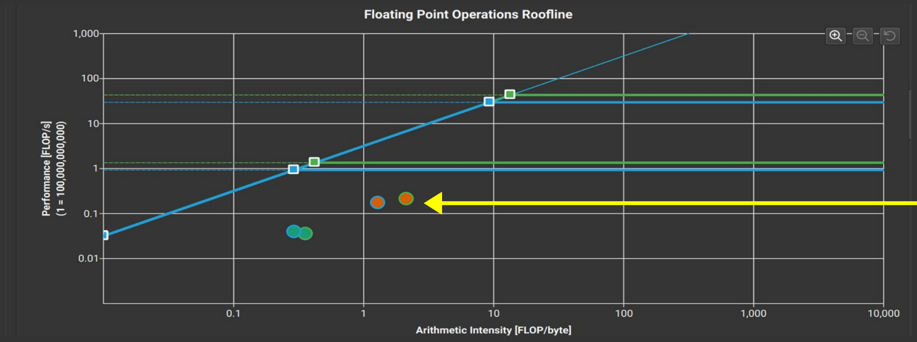
Optimization: Hardware generations

- Scaling on the GPU: View from the "inside"



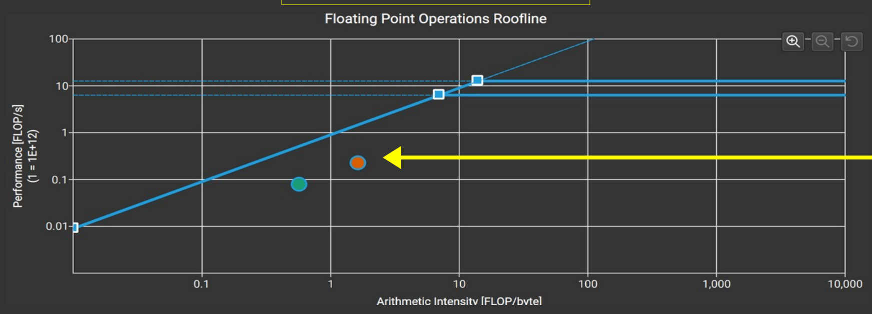
[Courtesy Jagannathan, Heriart]

NVIDIA T4 GPU

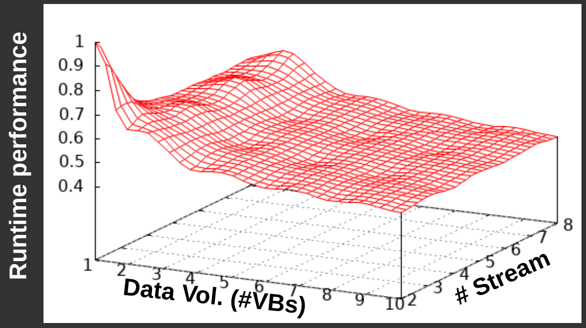


HPG is compute bound

NVIDIA V100 GPU



HPG is memory-bandwidth bound



Issues, future work

- Ported one compute hot-spot using Kokkos
  - O(100x) improvement compared to a CPU core, but still need $O(10^3)$ GPUs!
  - GPU occupancy remains low: < 50%

- Scaling with data volume (in GPU memory)
  - Runtime remains unchanged with data volume, No. of Streams
- I/O bandwidths
  - Data store → Compute nodes → GPU
- Move more compute to GPU
  - Calibration : Multiple iterations on data in GPU memory
  - Compute CF in the GPU: OTF numerically, Analytical

- Kokkos for logically partitioned GPUs (H100)?
- Performance of the same code on GPU and CPU?
- Decorations for Roofline model of a code segement?