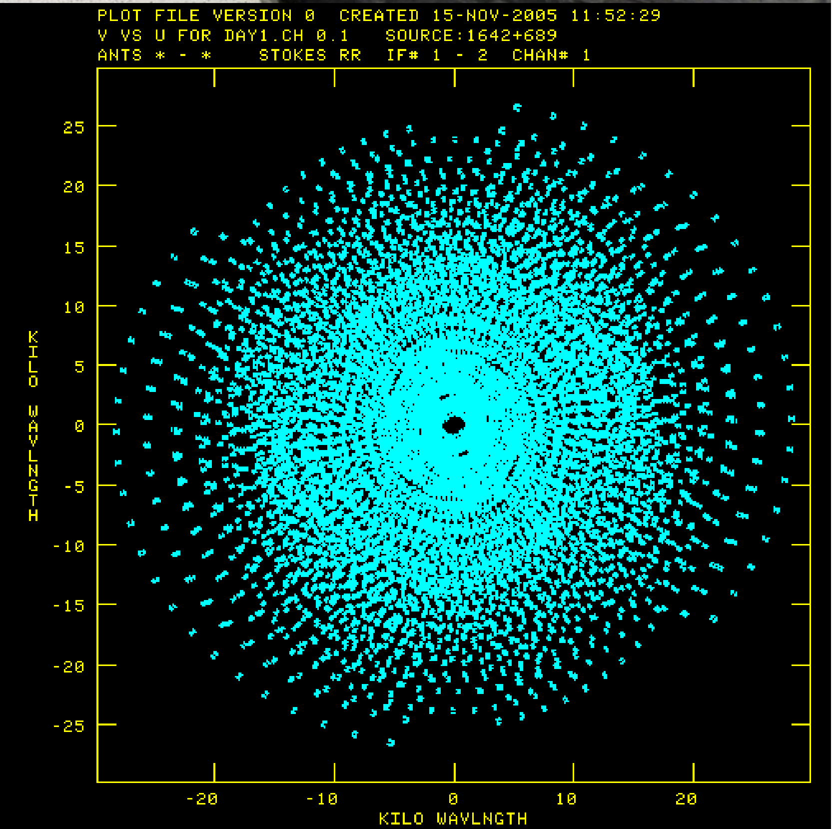


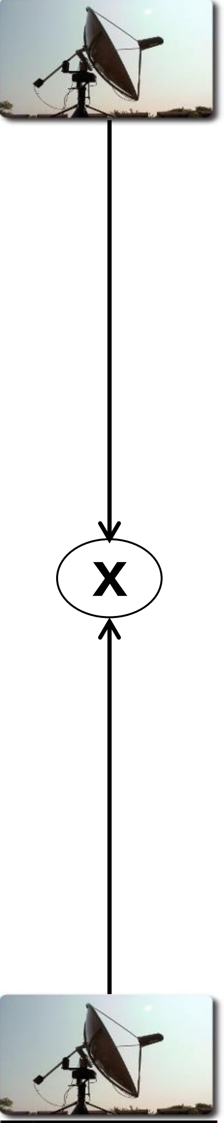
Cross correlators

for radio astronomy

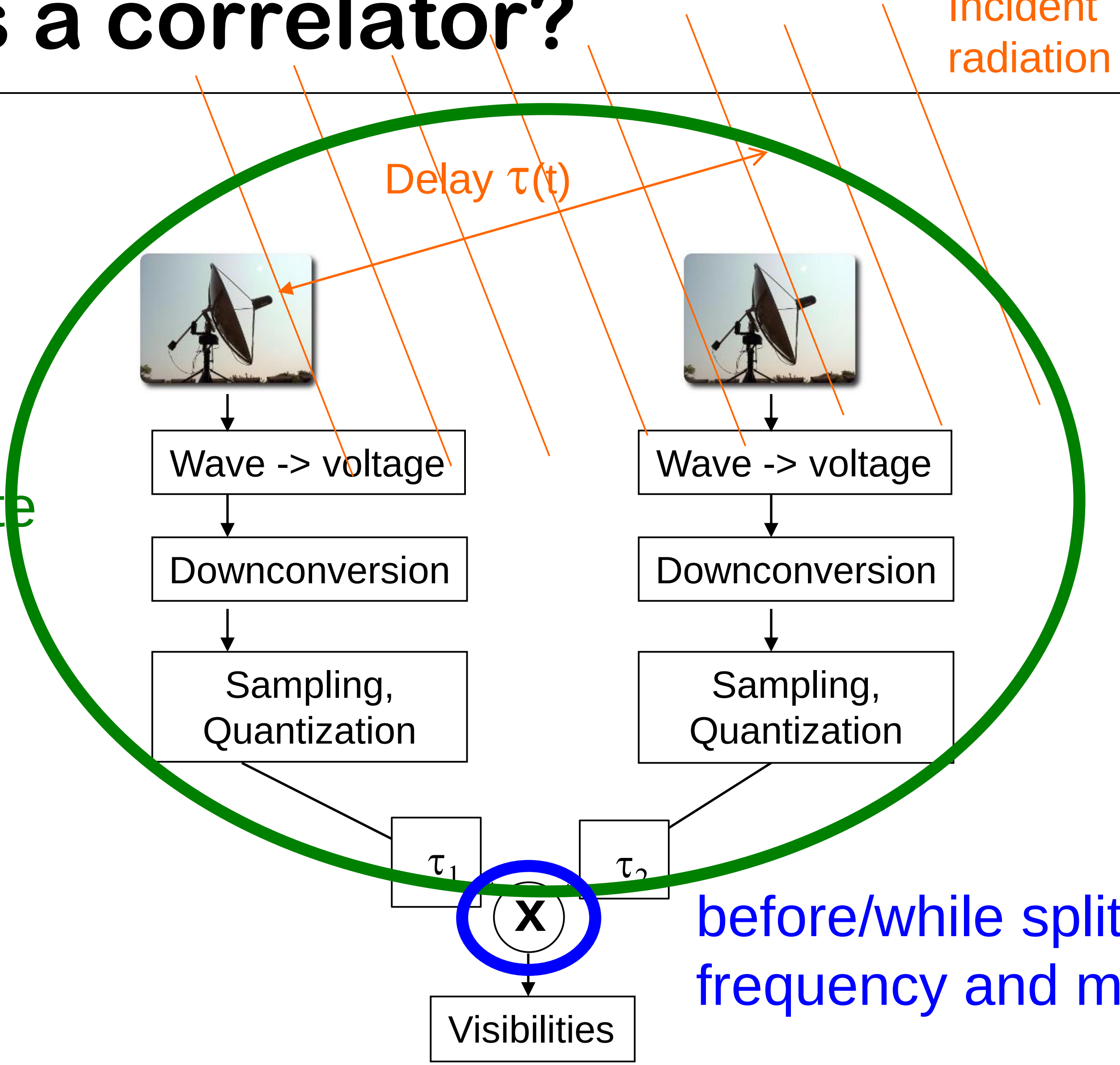
Adam Deller

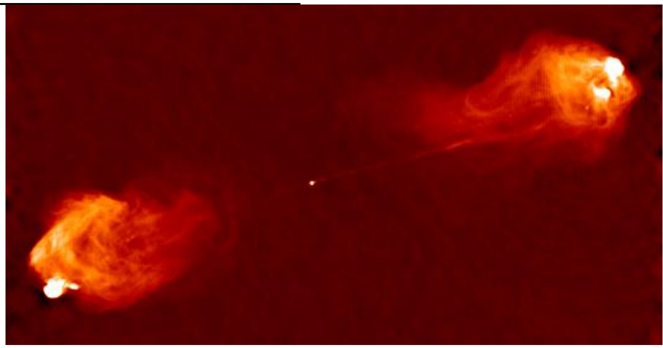
May 15, 2024

SWINBURNE UNIVERSITY OF TECHNOLOGY



What is a correlator?

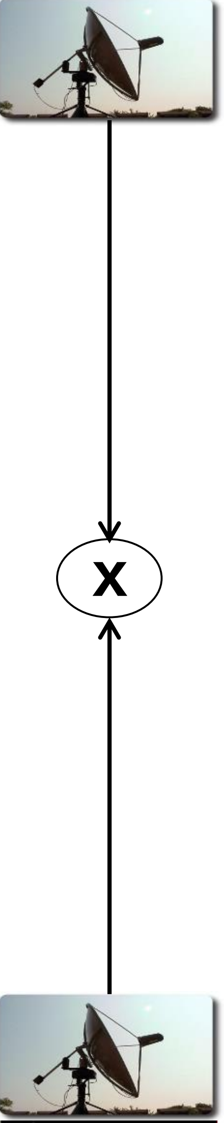




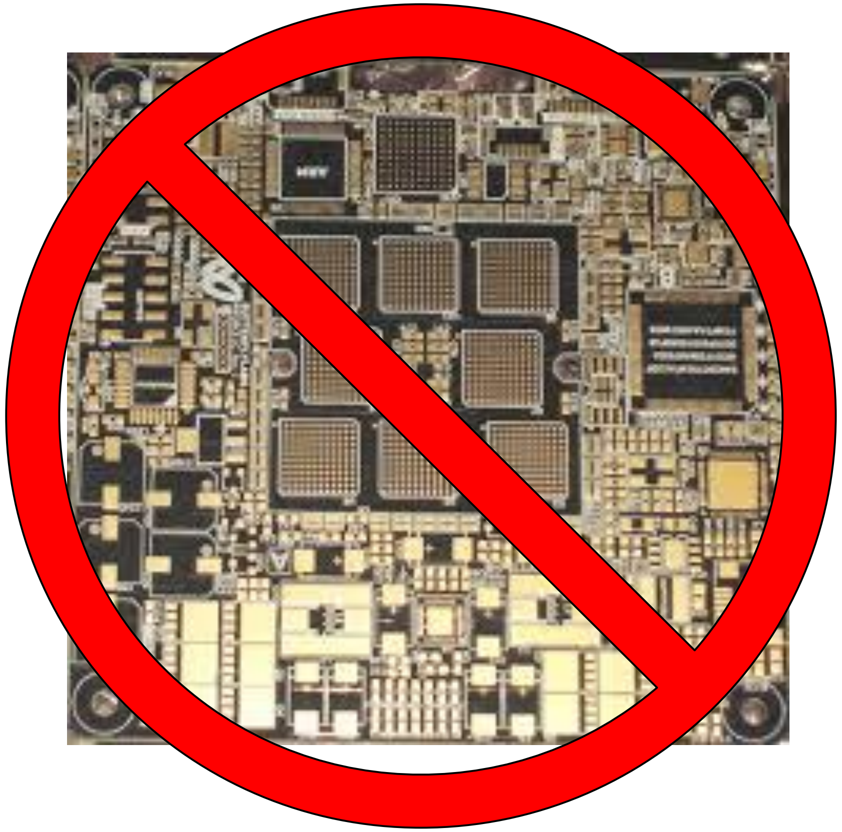
Compensate
for this…

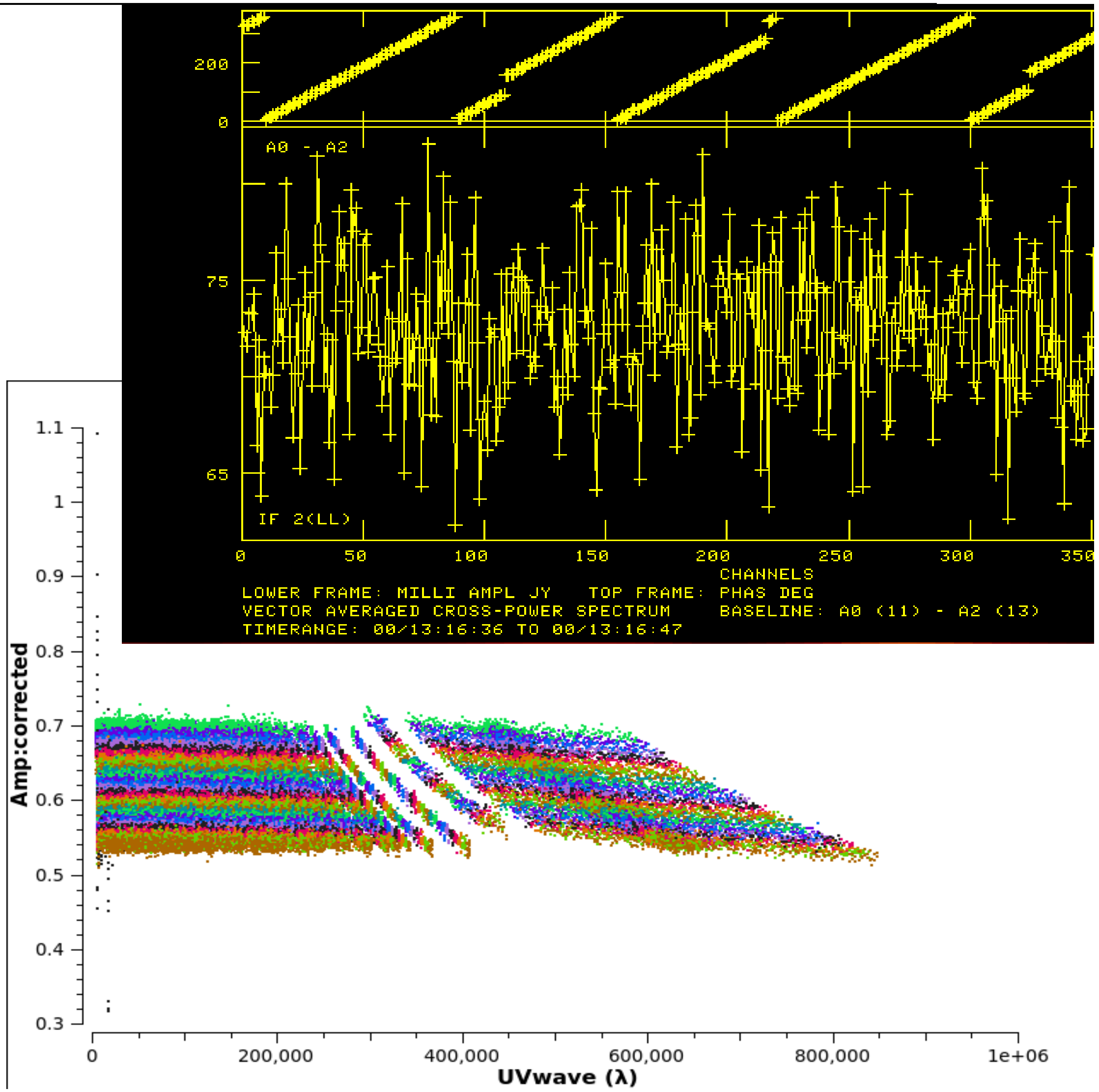
before/while splitting by
frequency and multiplying

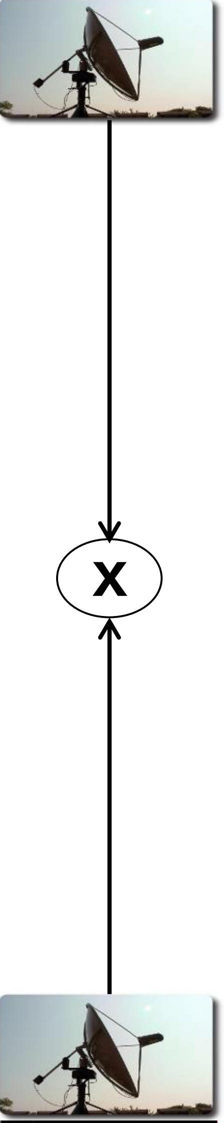
SWINBURNE UNIVERSITY OF TECHNOLOGY



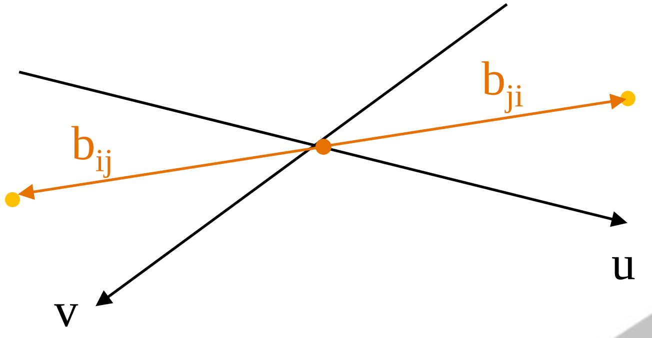
Why correlators matter to YOU

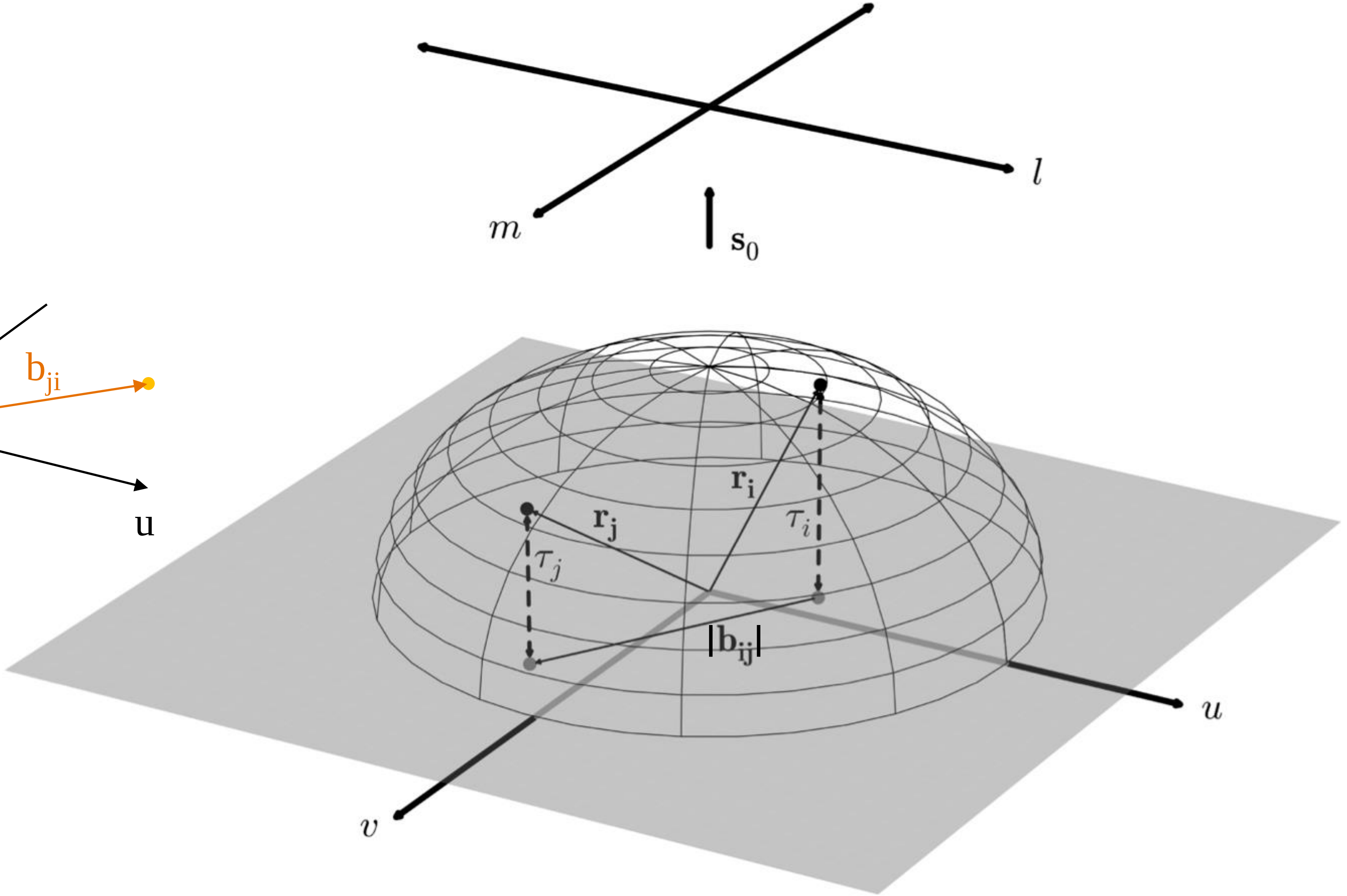


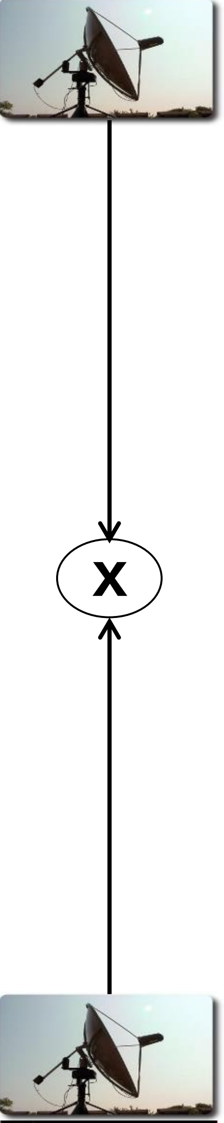




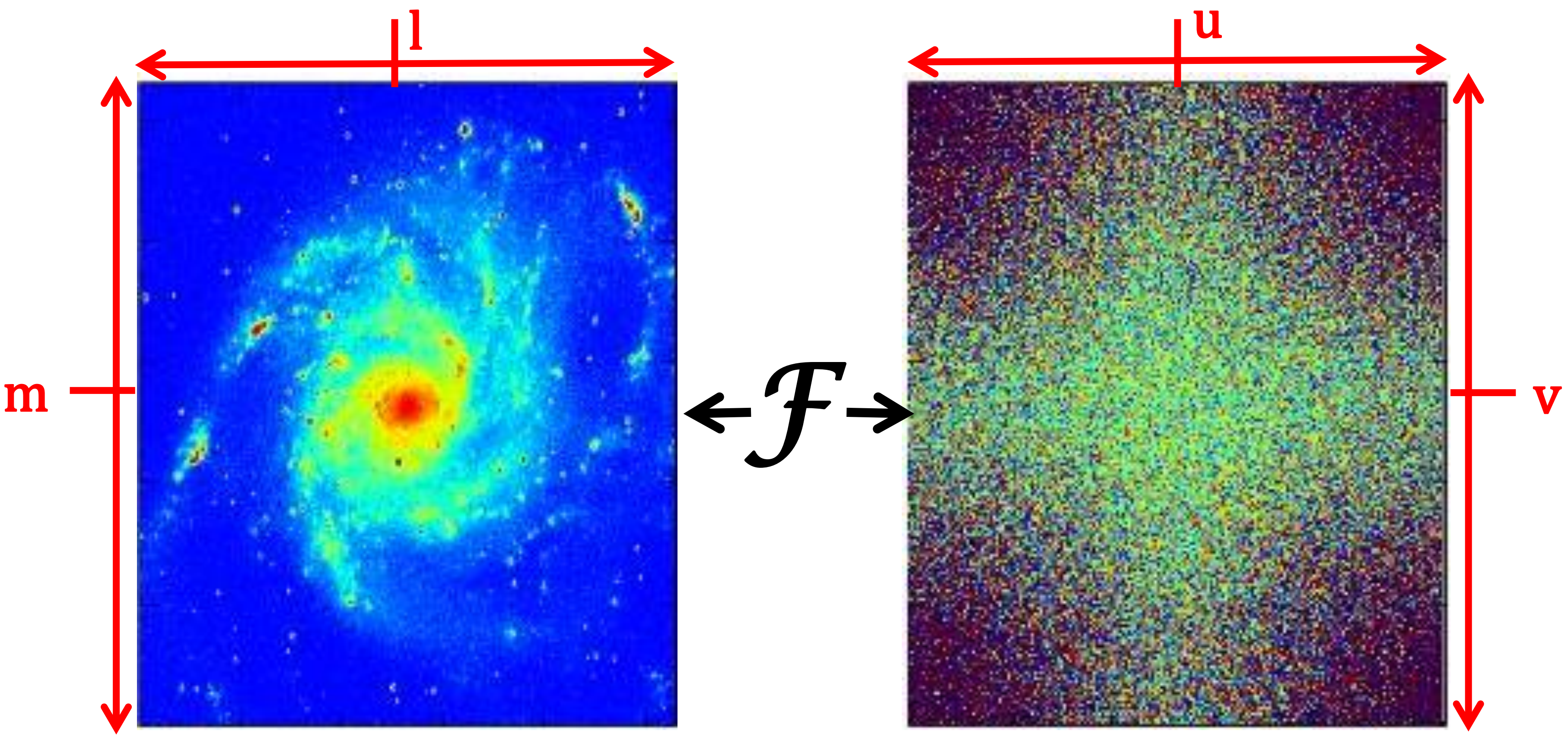
Correlators and Interferometry







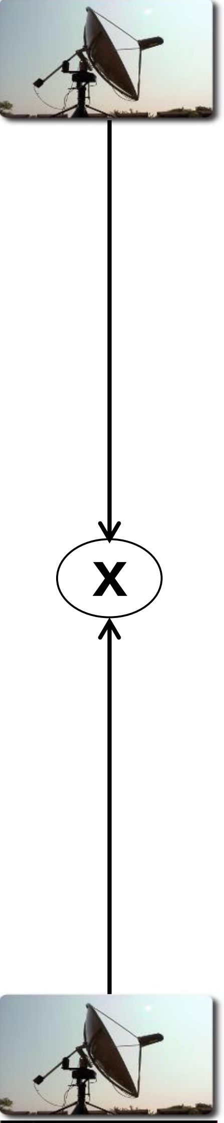
Correlators and Interferometry



Sky brightness at frequency ν_0

Visibilities (real component shown, unit is $\lambda_0 = c / \nu_0$)

SWINBURNE UNIVERSITY OF TECHNOLOGY

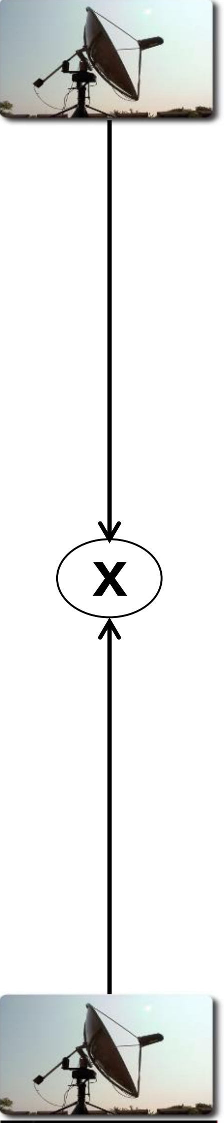


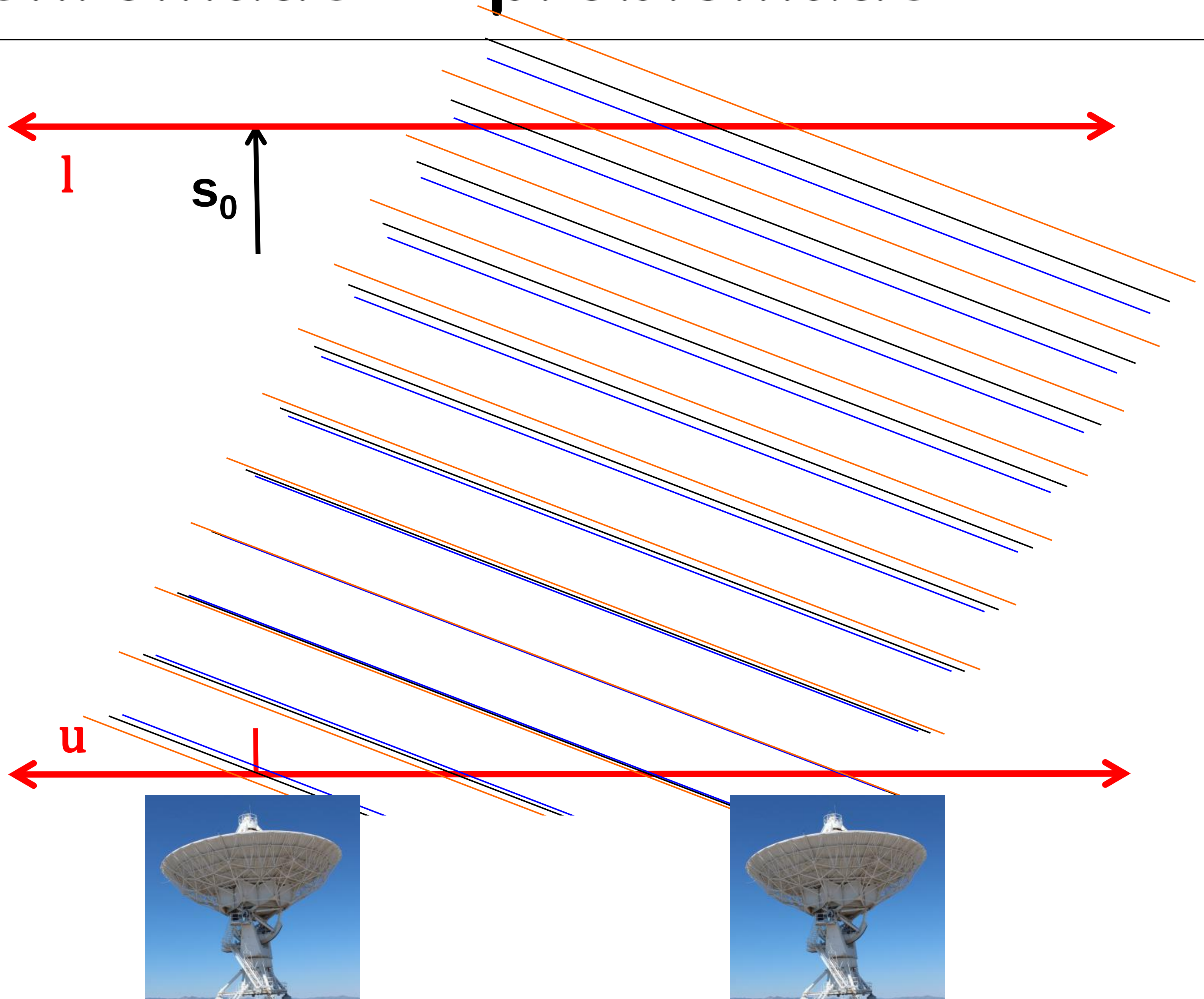
Monochromatic == problematic

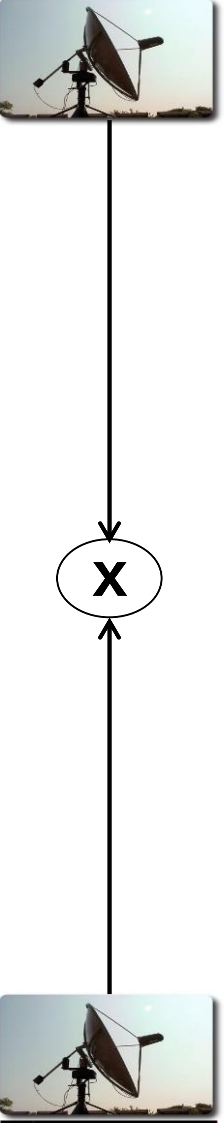
$$V(u,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \mathcal{A}(l,m) I(l,m) e^{-2\pi i(ul+vm)}\, dl\, dm\,.$$

- $u \times l + v \times m$ is supposed to be constant, but both u and v depend on frequency
- No truly monochromatic radiation!
- Fortunately, “fairly narrow” band of $\Delta\nu$ (*quasi-monochromatic*) can suffice:
  - Real world viewpoint: different frequency components stay “in phase” as wavefront propagates from one antenna to the next

Monochromatic == problematic



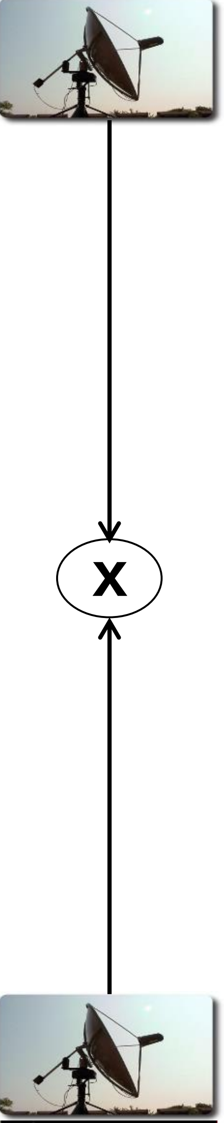




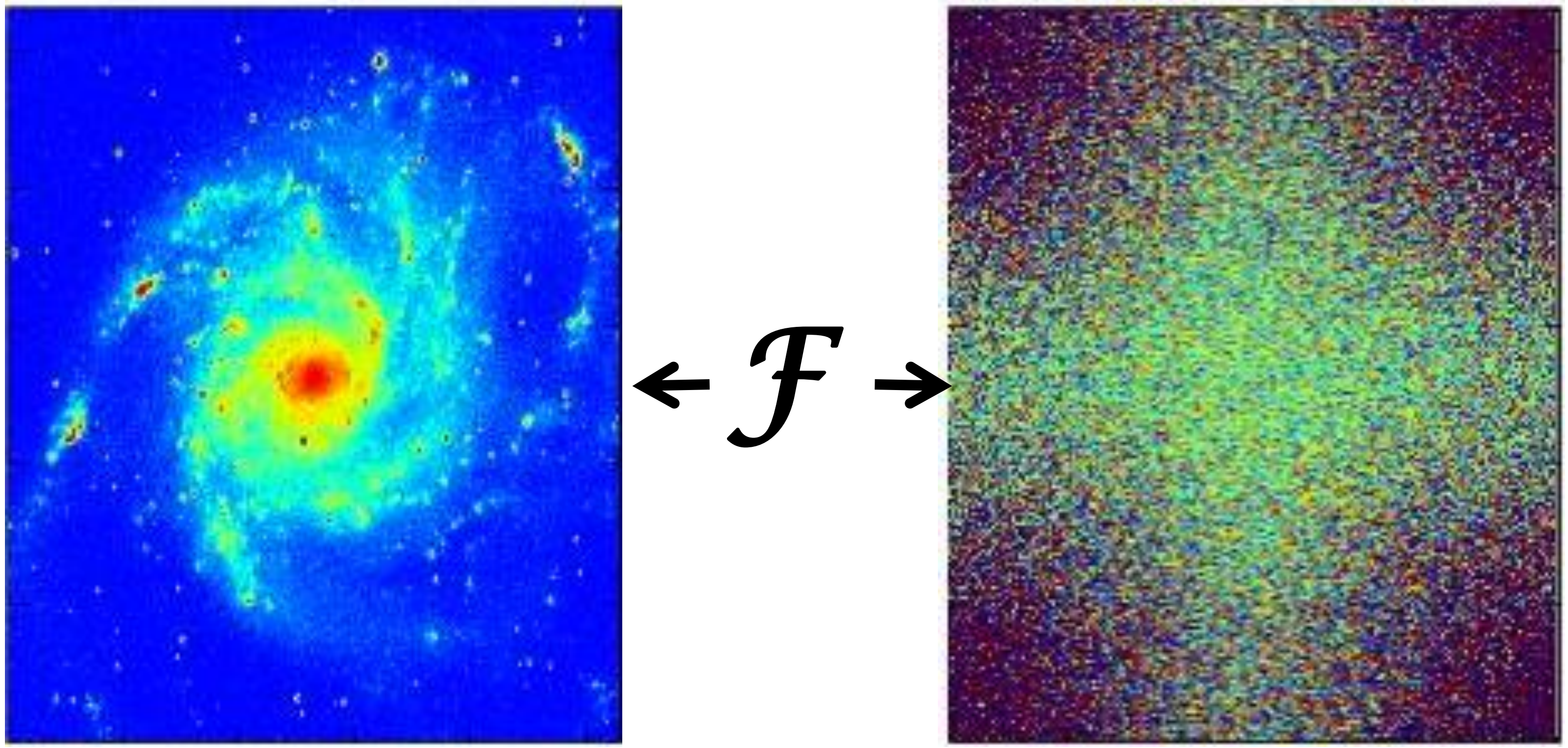
Monochromatic == problematic

$$V(u,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \mathcal{A}(l,m) I(l,m) e^{-2\pi i(ul+vm)}\, dl\, dm\,.$$

- $u \times l + v \times m$ is supposed to be constant, but both u and v depend on frequency
- No truly monochromatic radiation!
- Fortunately, “fairly narrow” band of $\Delta\nu$ (*quasi-monochromatic*) can suffice:
  - if $\Delta u \times l \ll 1$ and $\Delta v \times m \ll 1$ then the different frequency components stay in phase and we’re ok
  - Correlator needs to slice at least this finely



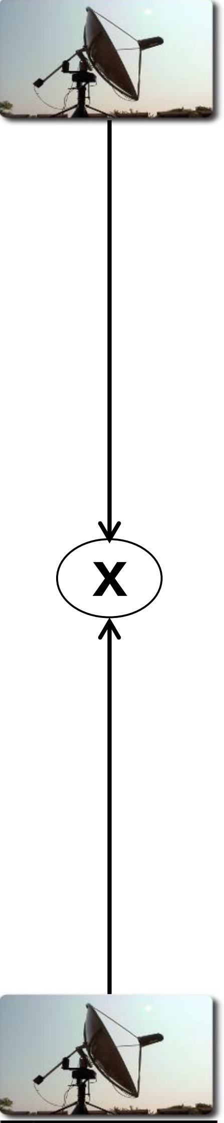
Correlators and Interferometry

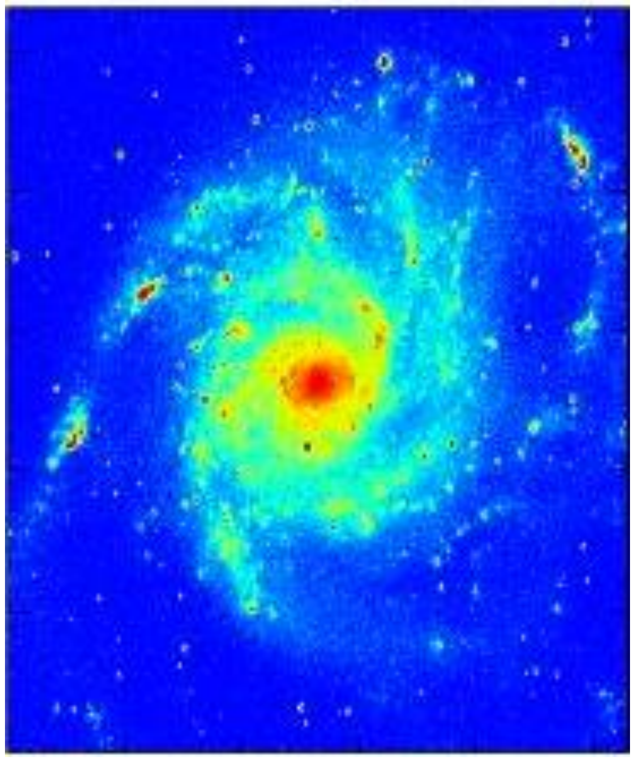


Sky brightness at frequency ν_0

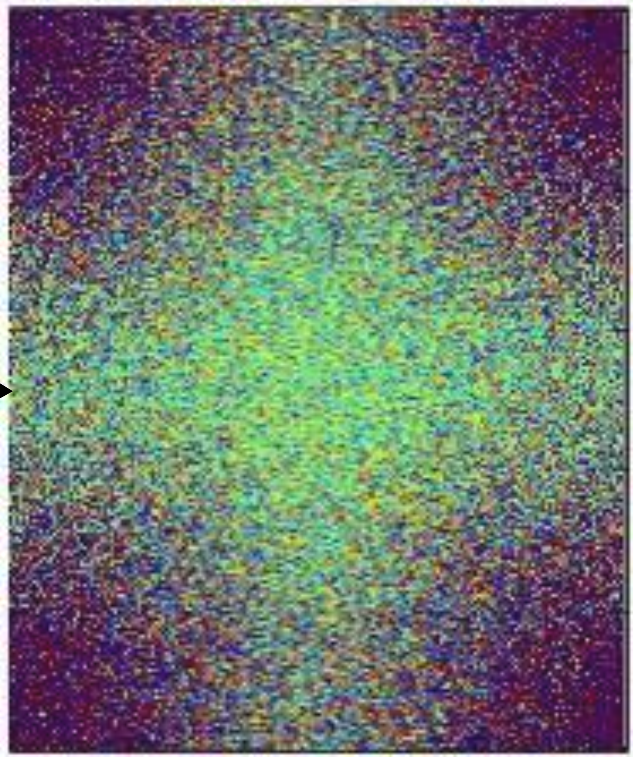
Visibilities (real component shown, unit is $\lambda_0 = c / \nu_0$)

Correlators and Interferometry





$\leftarrow \mathcal{F} \rightarrow$

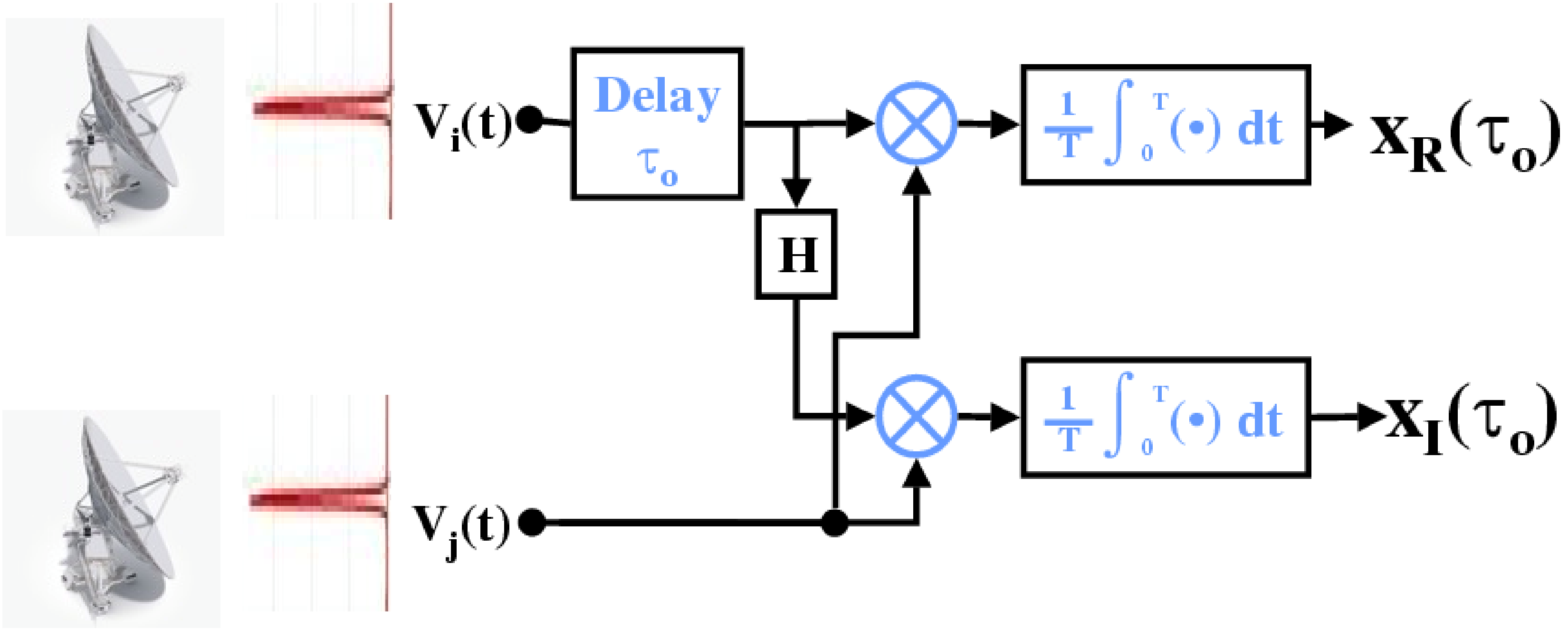


Sky brightness at frequency $\nu' = \nu_0 + \delta\nu$

Visibilities (real component shown, unit is $\lambda' = c / \nu'$)

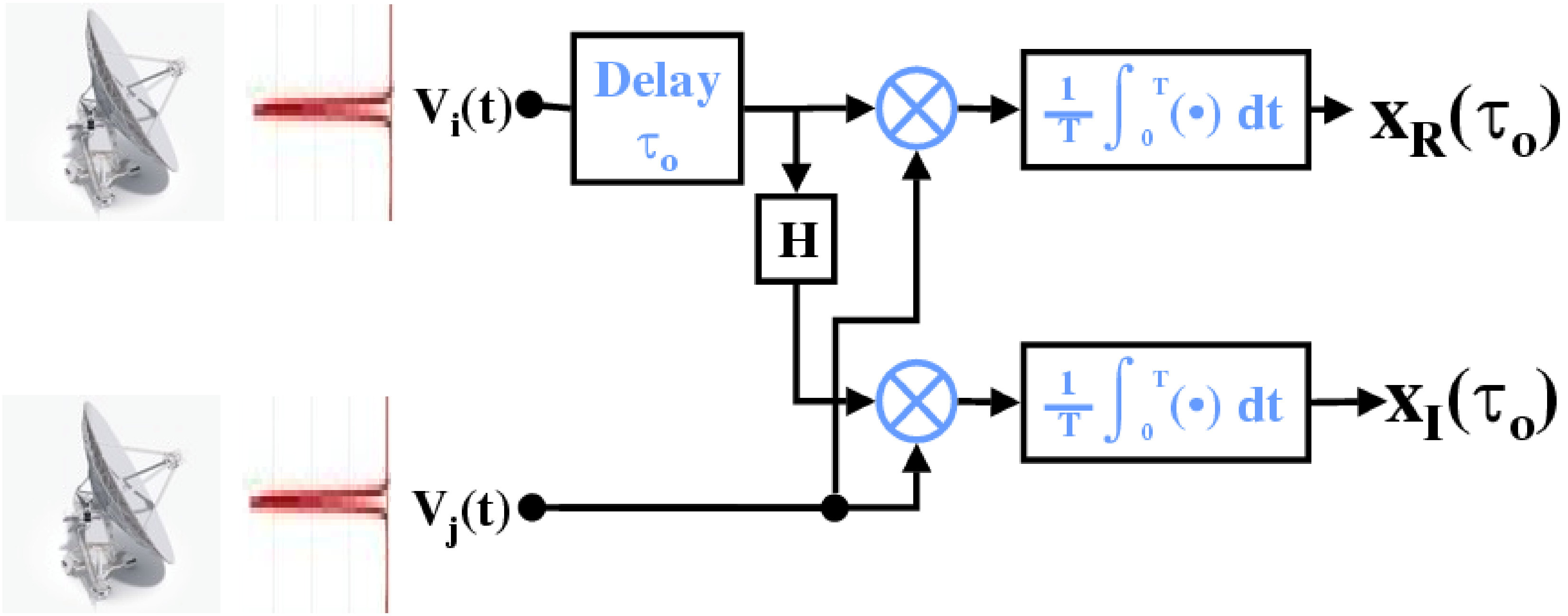
A “dumb” correlator

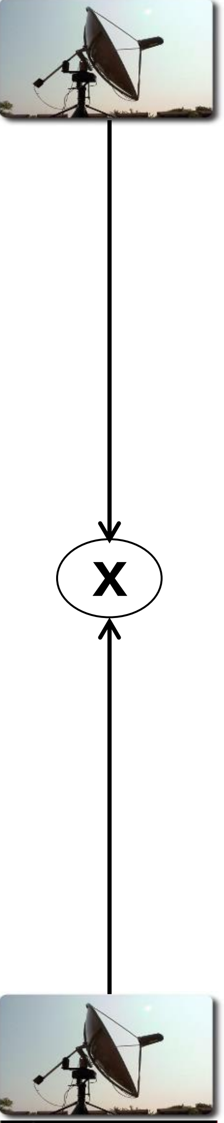
- Use many analog filters to make many narrow channels; correlate each one separately with a standard complex correlator:



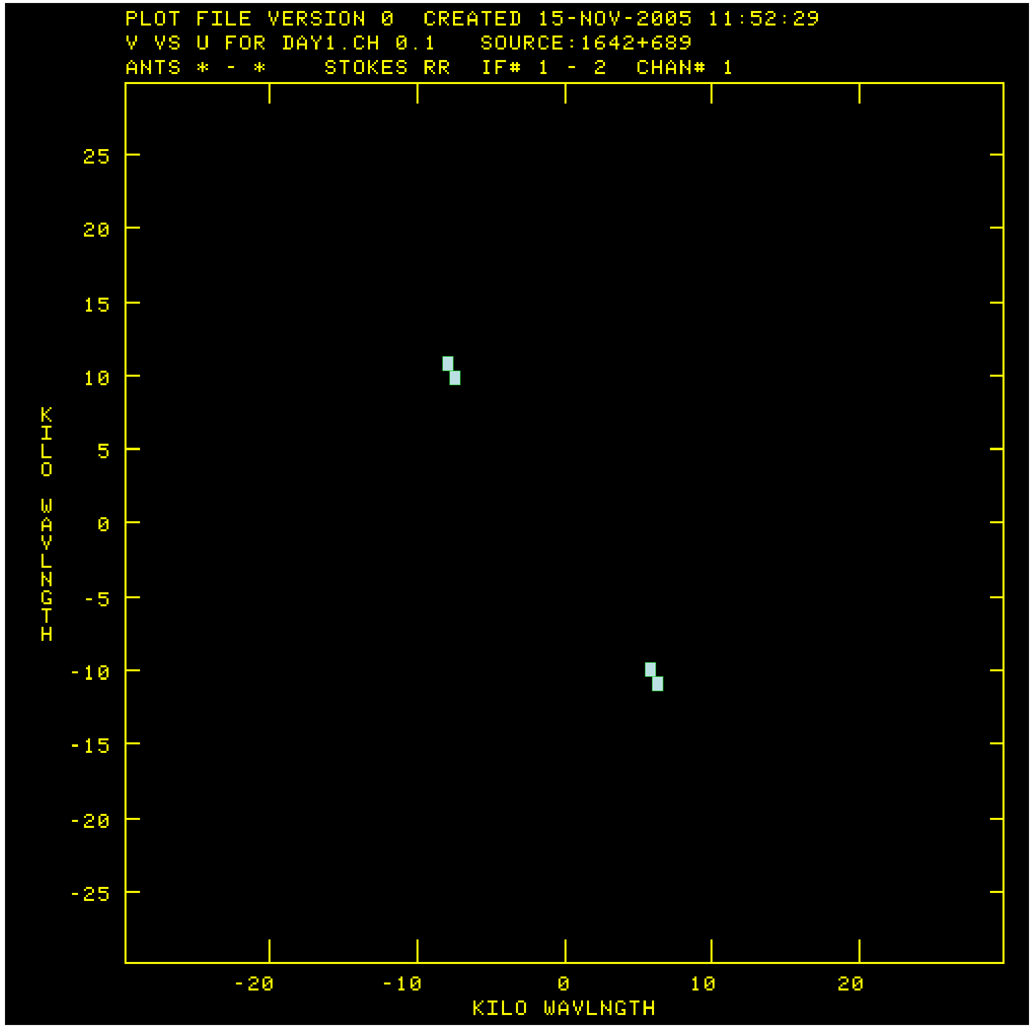
A "dumb" correlator

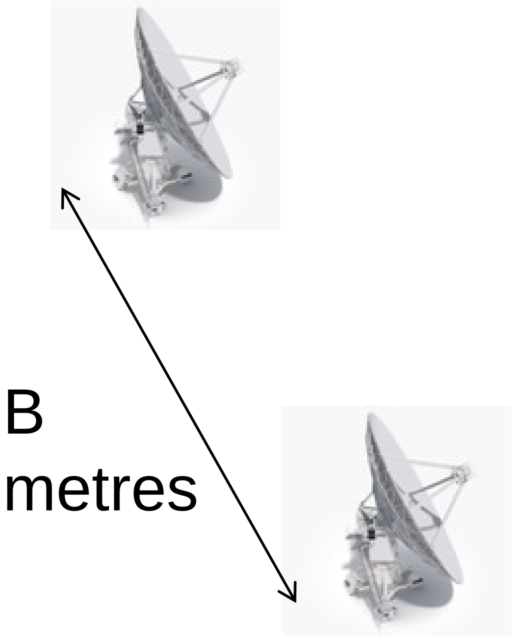
- Use many analog filters to make many narrow channels; correlate each one separately with a standard complex correlator:



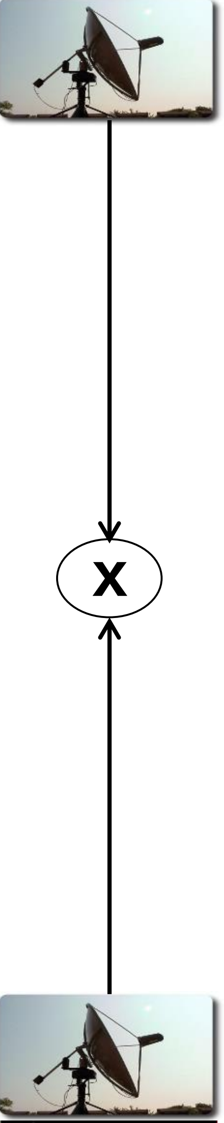

X

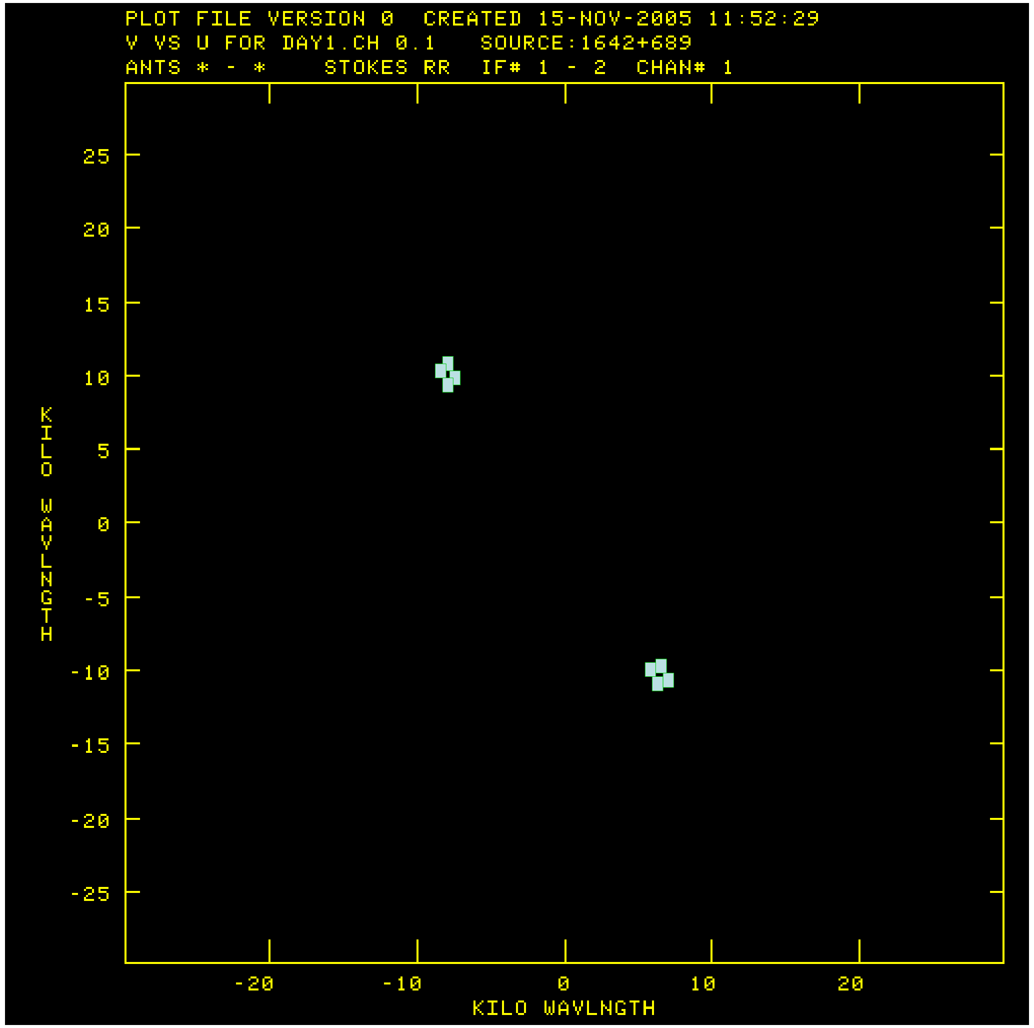
The output

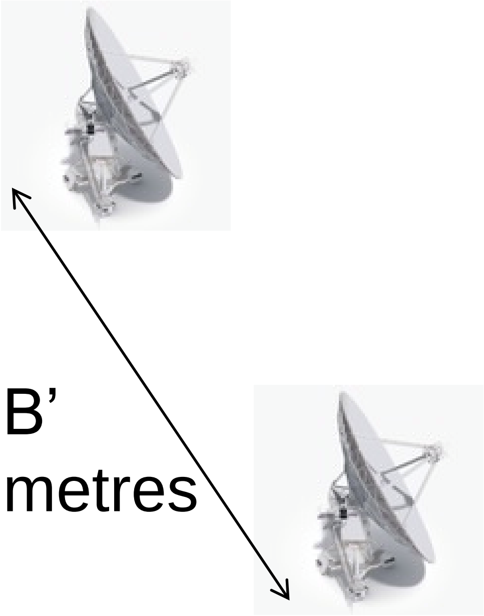

PLOT FILE VERSION 0 CREATED 15-NOV-2005 11:52:29
V VS U FOR DAY1.CH 0.1   SOURCE:1642+689
ANTS * - *   STOKES RR   IF# 1 - 2 CHAN# 1
KILO WAVLNGTH
KILO WAVLNGTH
25
20
15
10
5
0
-5
-10
-15
-20
-25
-20
-10
0
10
20


B
metres

The output



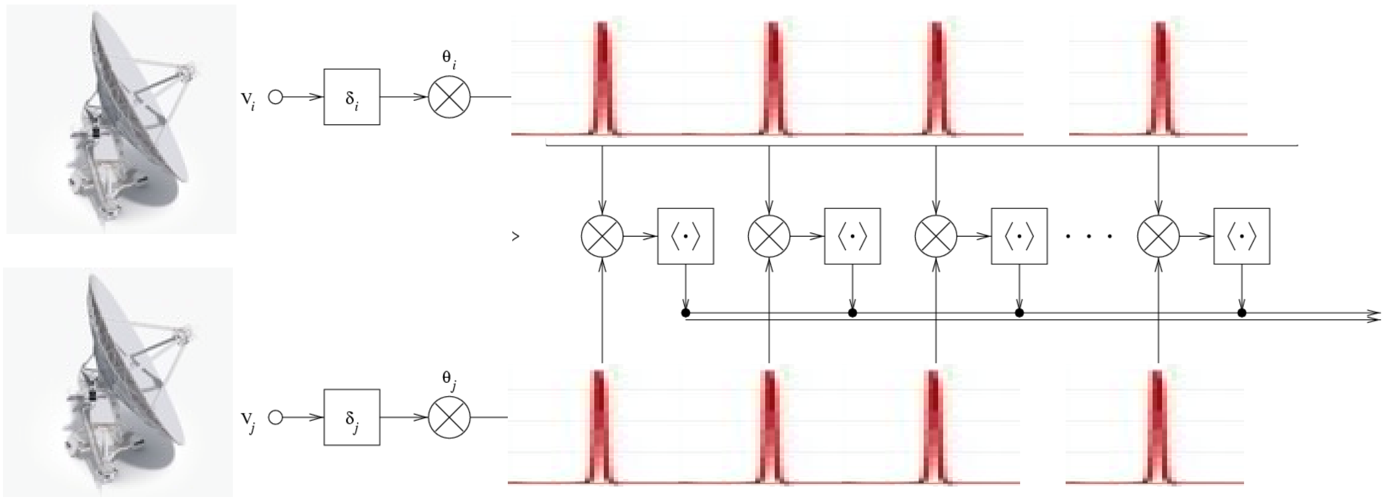




SWINBURNE UNIVERSITY OF TECHNOLOGY

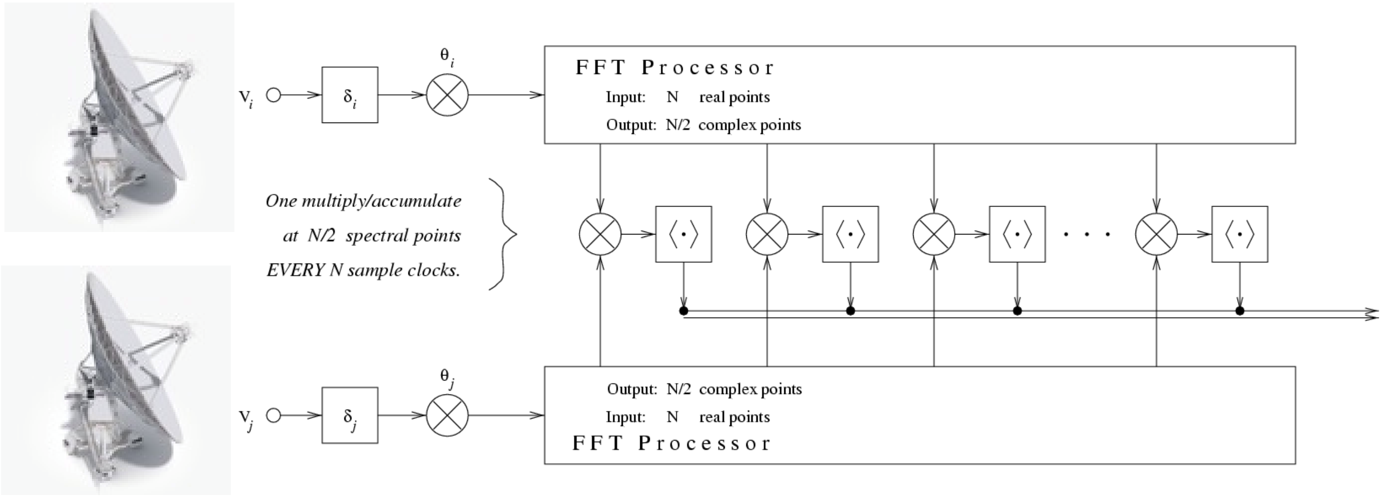
Making it feasible

- Analog filters are costly & finnicky; this would be expensive and temperamental

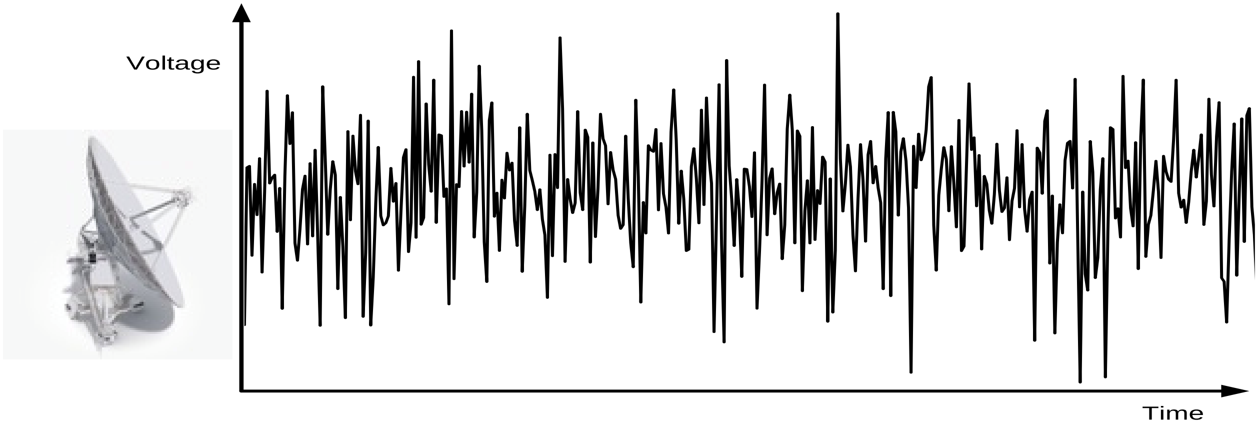


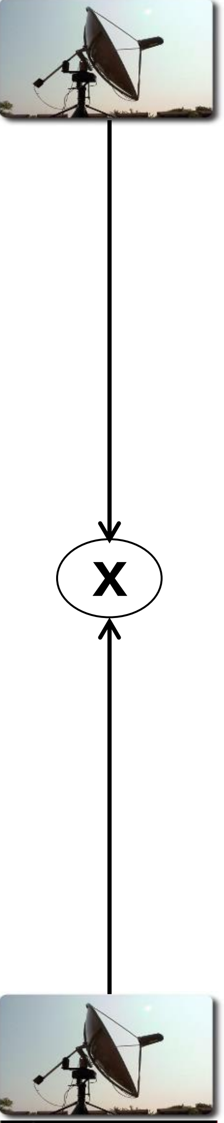
Making it feasible

- Analog filters are costly & finnicky; this would be expensive and temperamental
- Fortunately, we can (and do) digitize the signal – meaning we can use a digital substitute: **digital filterbank**

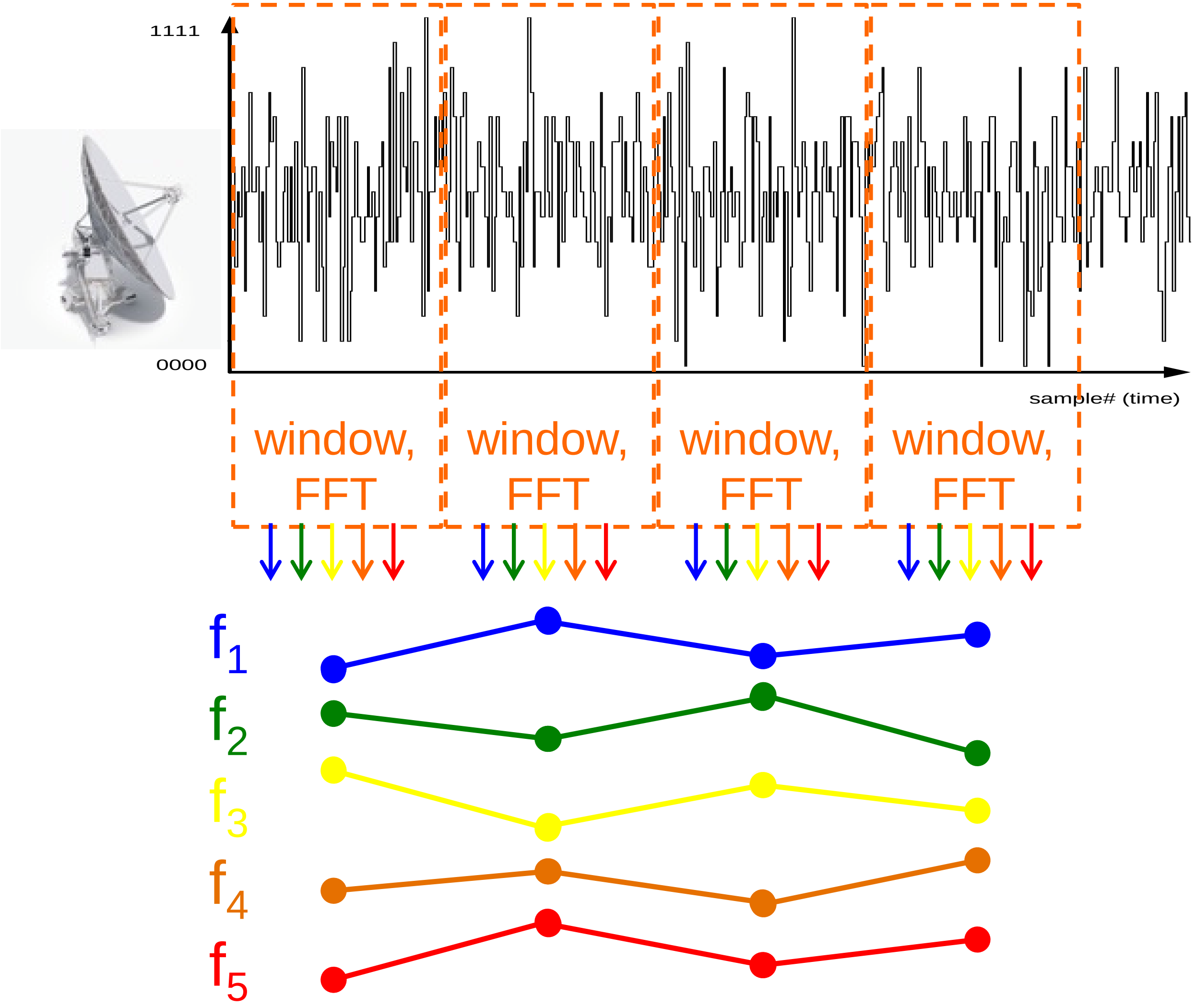


The “FX” correlator



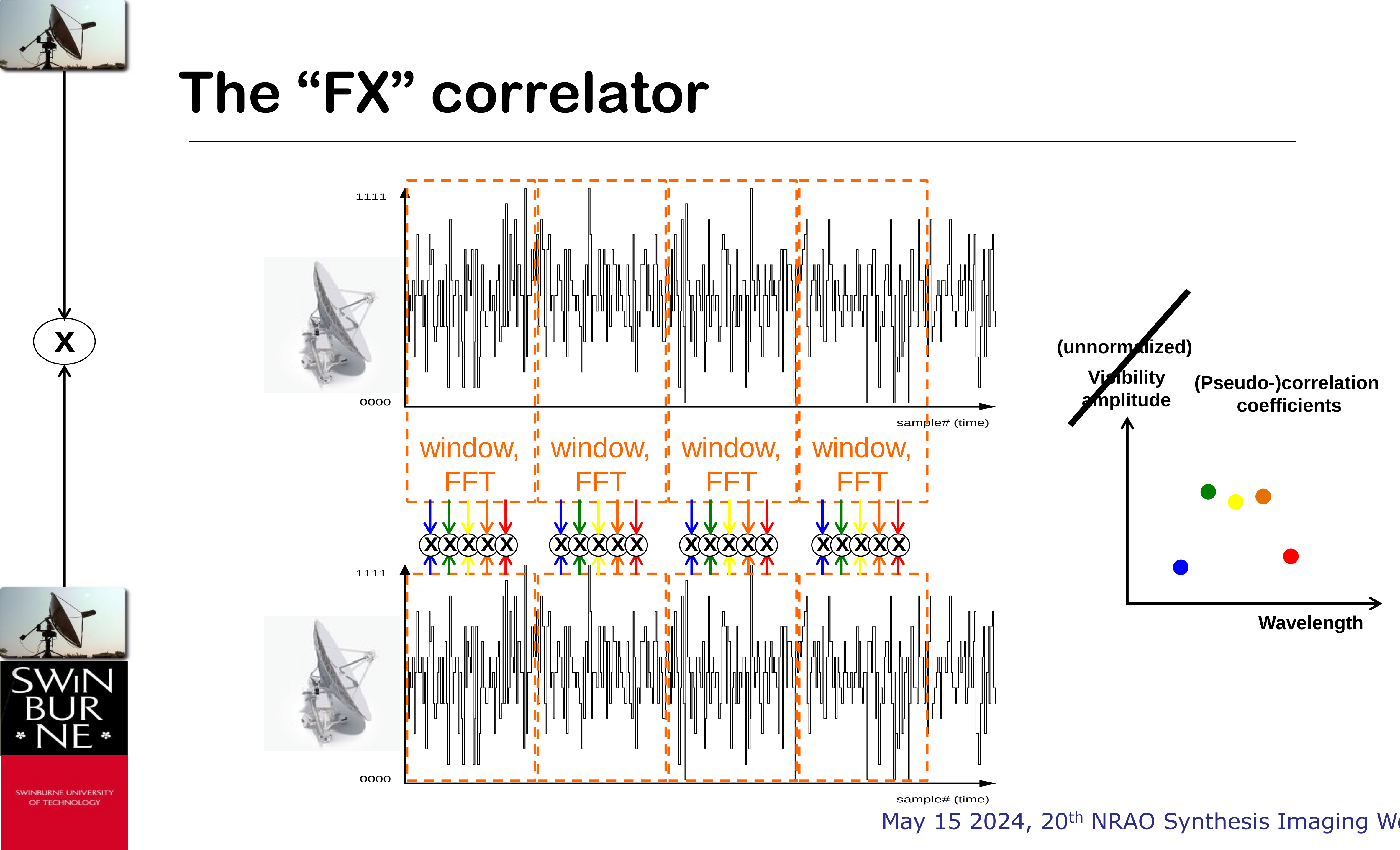


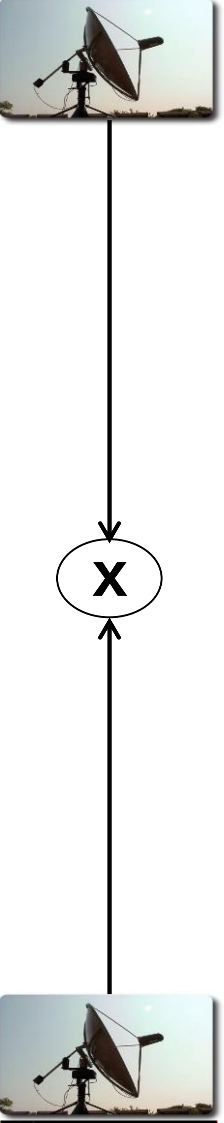
The “FX” correlator



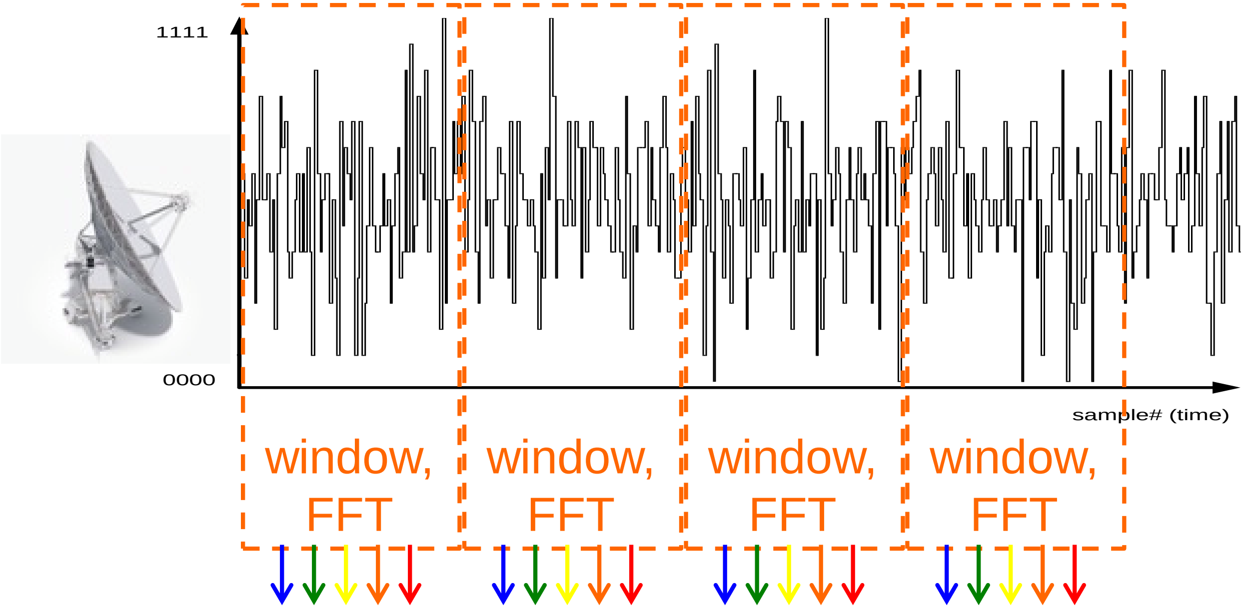
Each of these frequency channels is still a time series, representing a voltage that represents an electric field. It’s just that now, it represents a really **narrow** chunk of bandwidth (and is stored as a **complex** quantity – complex multiplication [conjugate antenna 2] rather than separate cosine + sine correlators)

The “FX” correlator



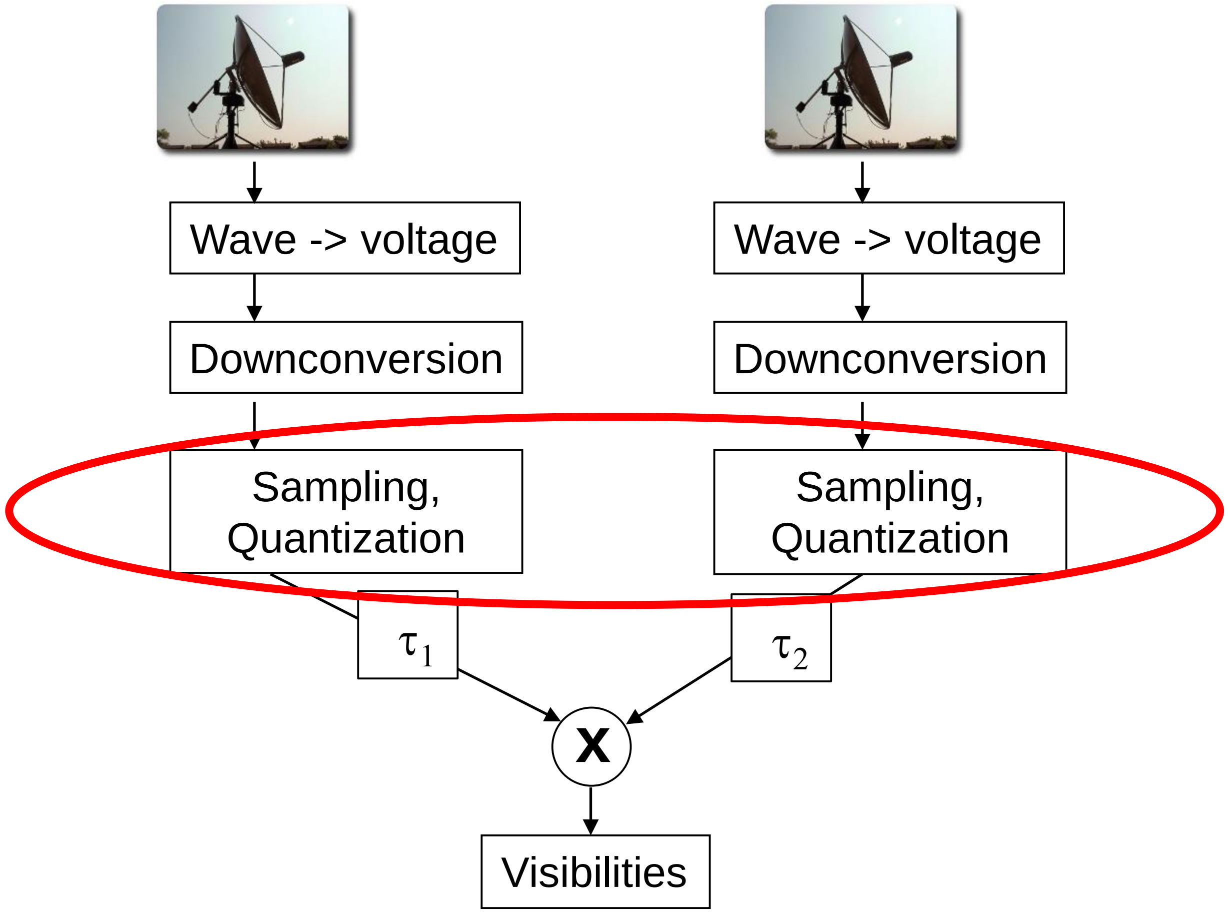


The “FX” correlator



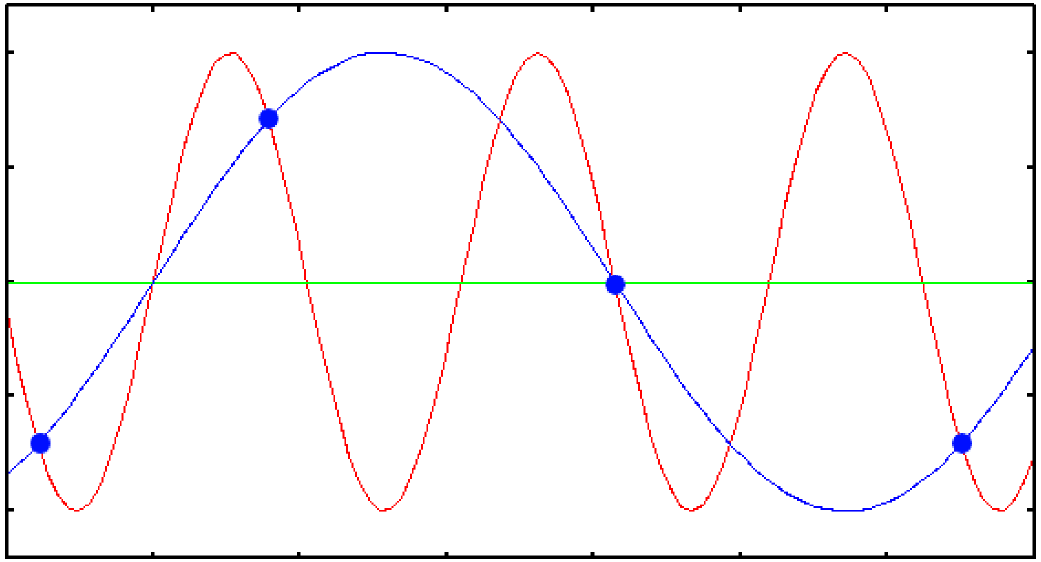
- Since this architecture consists of a <u>Fourier</u> transform (F) followed by <u>cross</u>-multiplication (X), we dub this the “FX” correlator

Righting the wrongs



Sampling

- Nyquist-Shannon sampling theorem:
  - real-valued signal is sampled every Δt sec
  - Original signal can be reconstructed perfectly so long as contains no power at frequencies ≥ 1 / (2 Δt) Hz (*band-limited*)

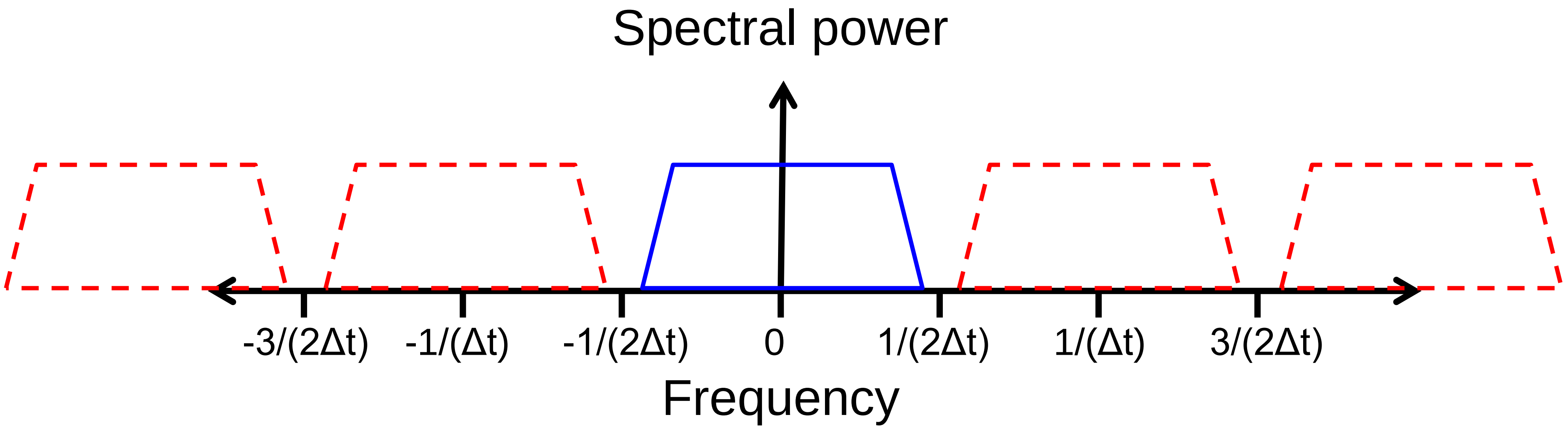


Adequately sampled

Undersampled, cannot be reconstructed

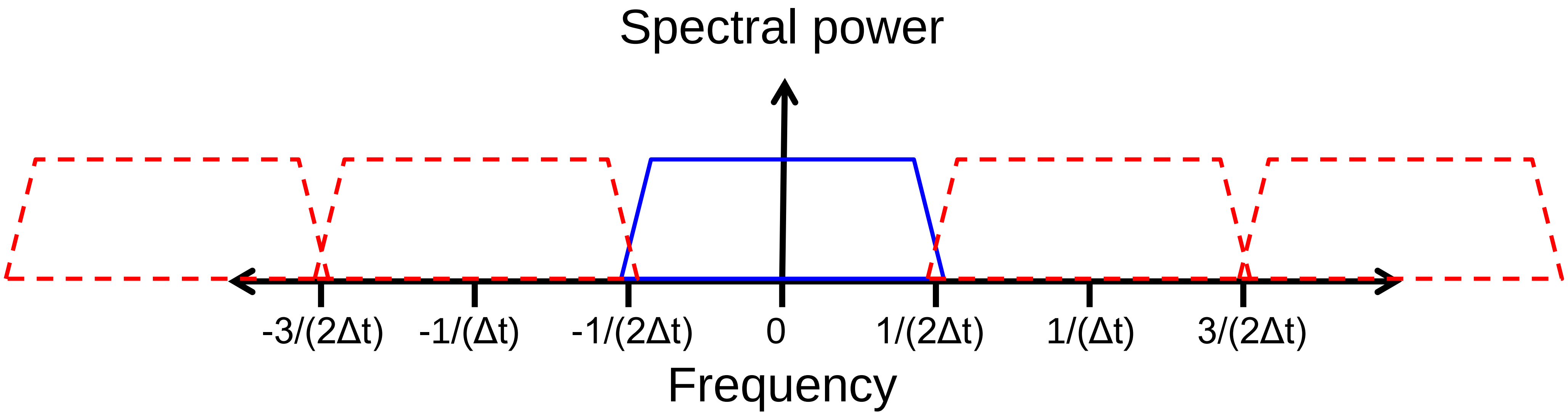
Sampling

- Nyquist-Shannon sampling theorem:
  - real-valued signal is sampled every Δt sec
  - Original signal can be reconstructed perfectly so long as contains no power at frequencies ≥ 1 / (2 Δt) Hz (*band-limited*)



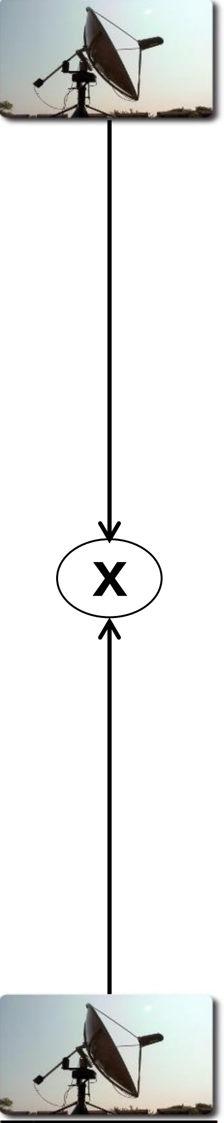
Sampling

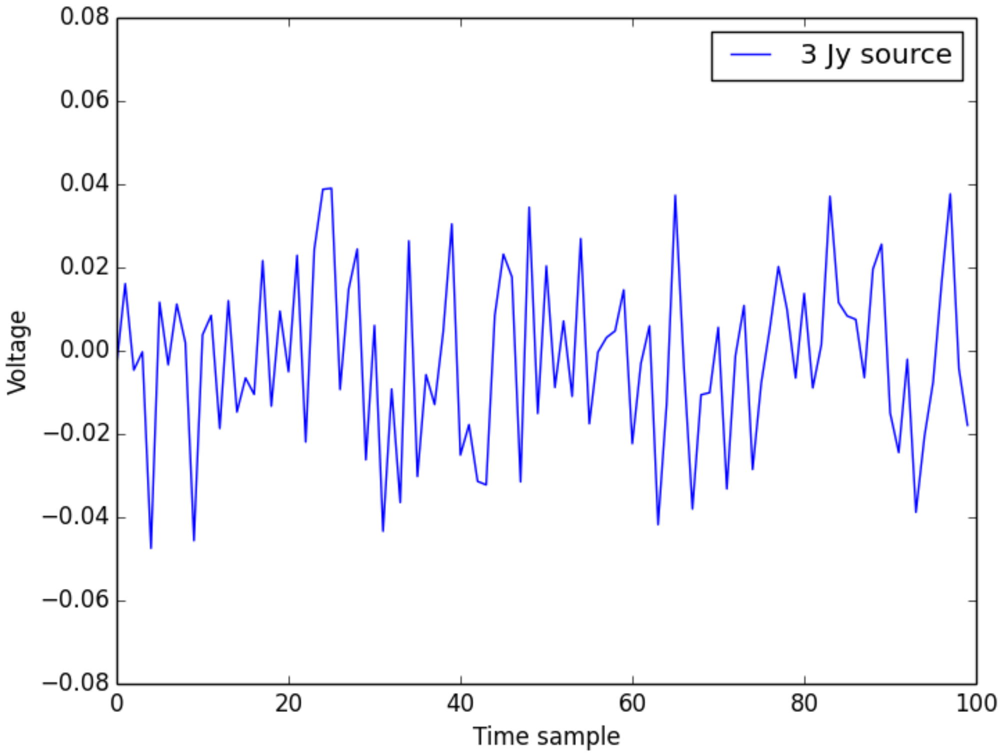
- Nyquist-Shannon sampling theorem:
  - real-valued signal is sampled every Δt sec
  - Original signal can be reconstructed perfectly so long as contains no power at frequencies ≥ 1 / (2 Δt) Hz (*band-limited*)

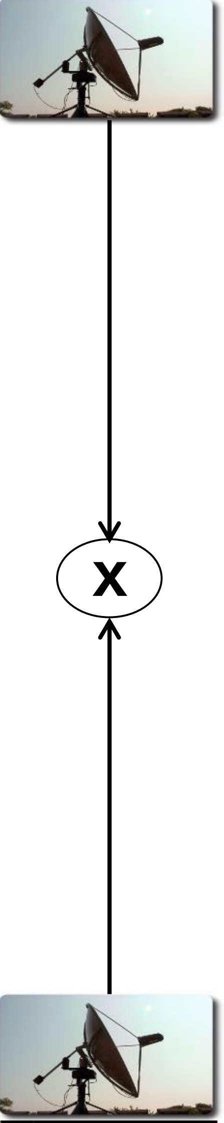


Quantization

- When correlation is low (almost always – sources are faint) even very coarse quantization is ok!

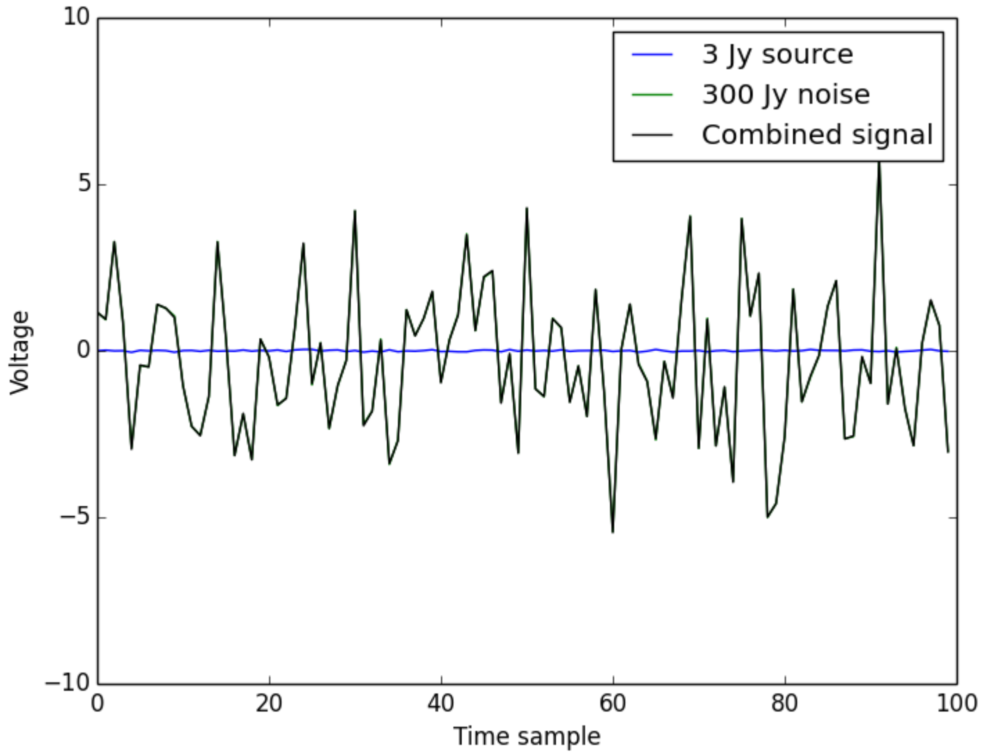






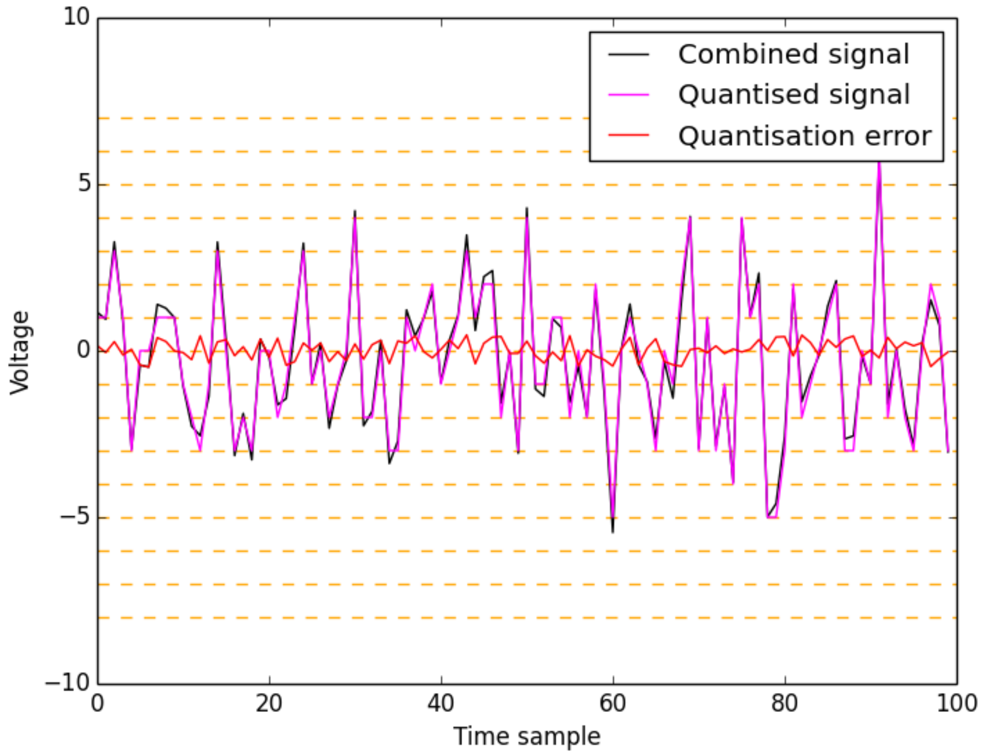
Quantization

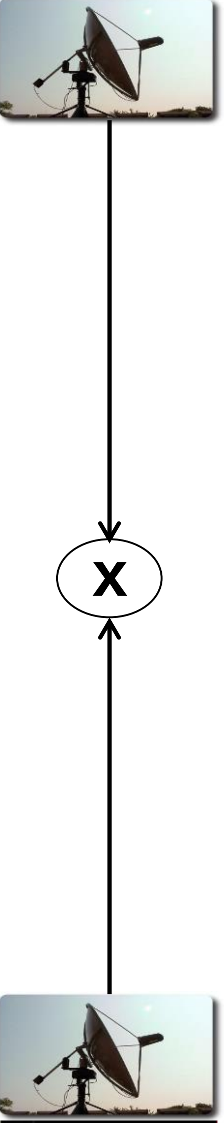
- When correlation is low (almost always, sources are faint) even very coarse quantization is ok!



Quantization

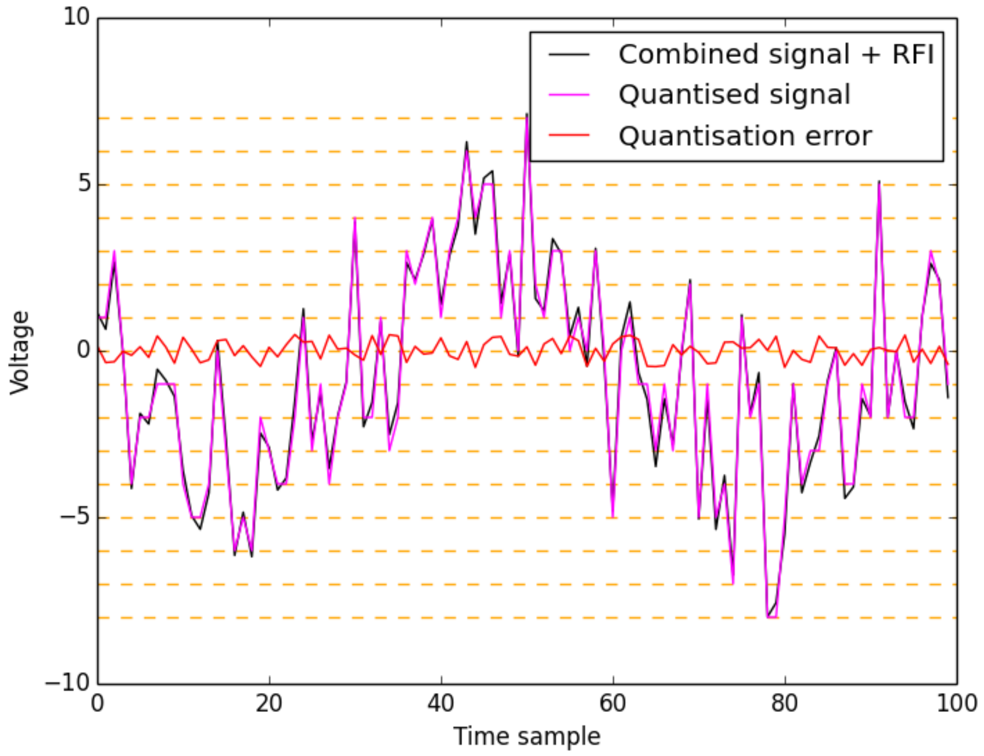
- When correlation is low (almost always, sources are faint) even very coarse quantization is ok!





Quantization

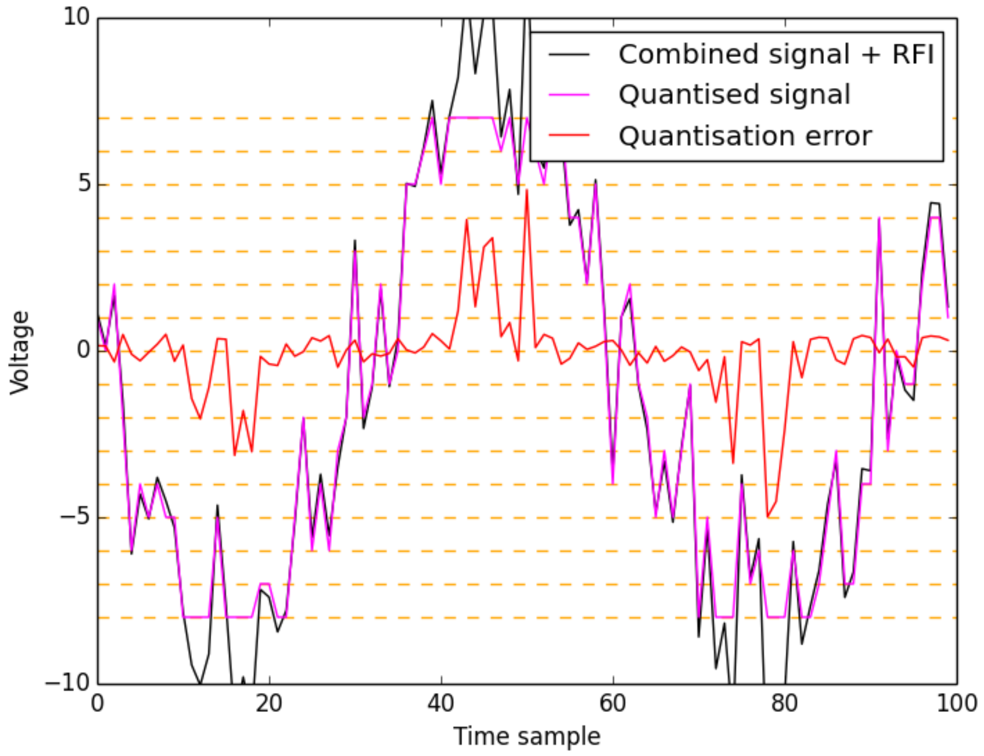
- When correlation is low (almost always, sources are faint) even very coarse quantization is ok!



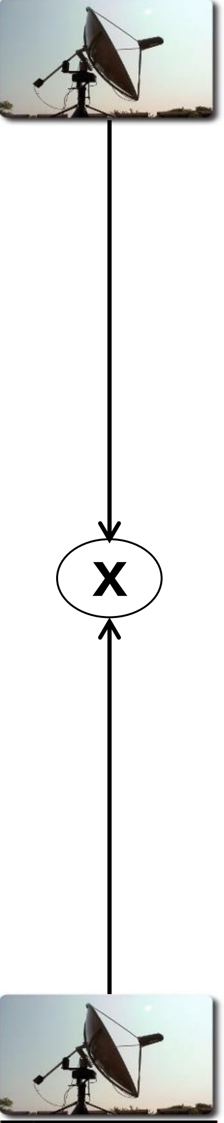
SWINBURNE UNIVERSITY OF TECHNOLOGY

Quantization

- When correlation is low (almost always, sources are faint) even very coarse quantization is ok!



until the headroom runs out…

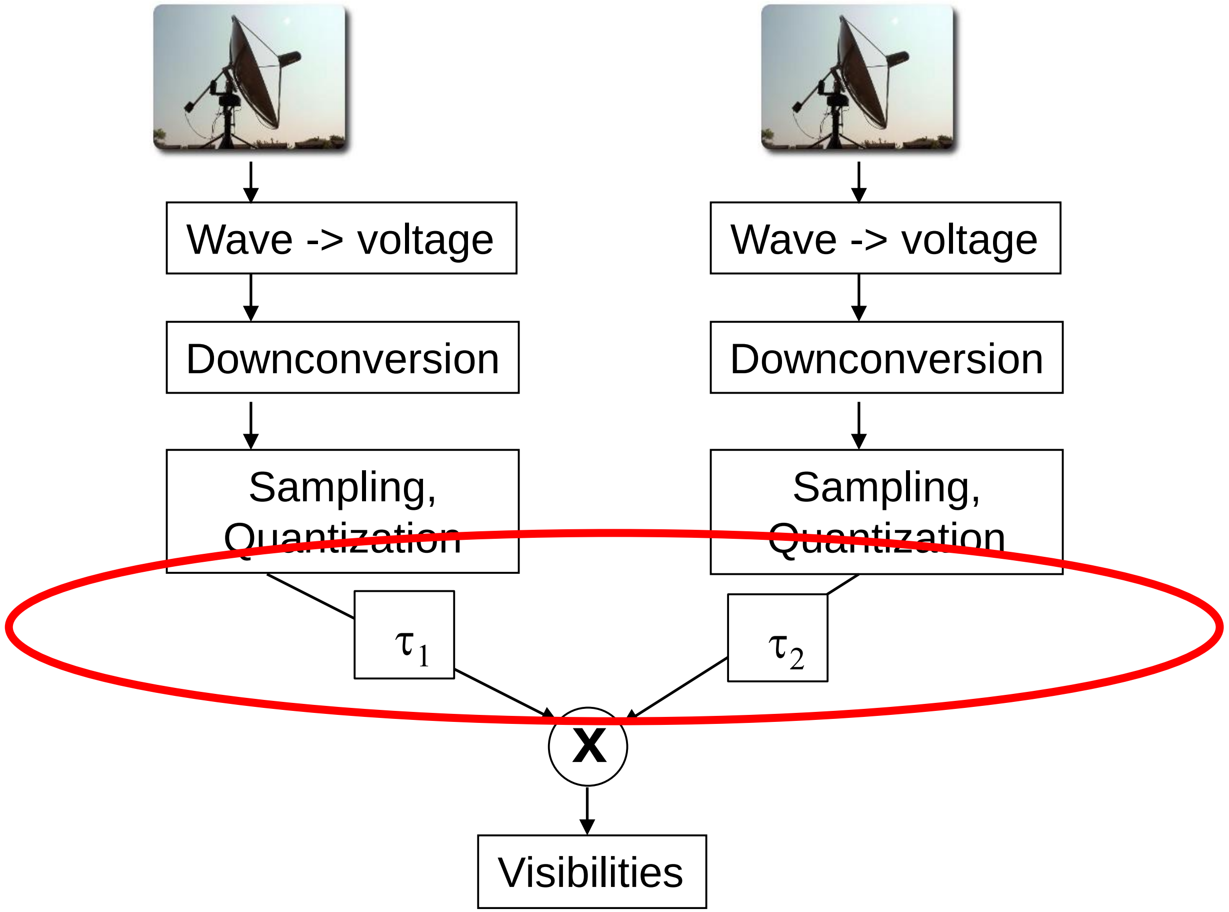


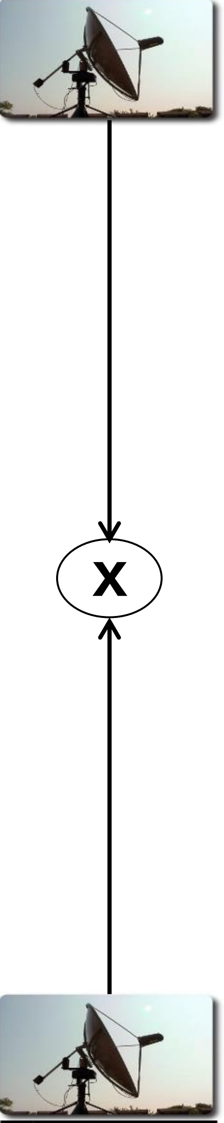
Quantization

- When correlation is low (almost always, sources are faint) even very coarse quantization is ok!
- Sensitivity loss due to quantisation:
  - 8 bit: 0.1%
  - 4 bit: 1.3%
  - 2 bit: 12%
  - 1 bit: 36%
- In correlator or after correlator: correct visibility amplitudes for this sensitivity loss

SWINBURNE UNIVERSITY OF TECHNOLOGY

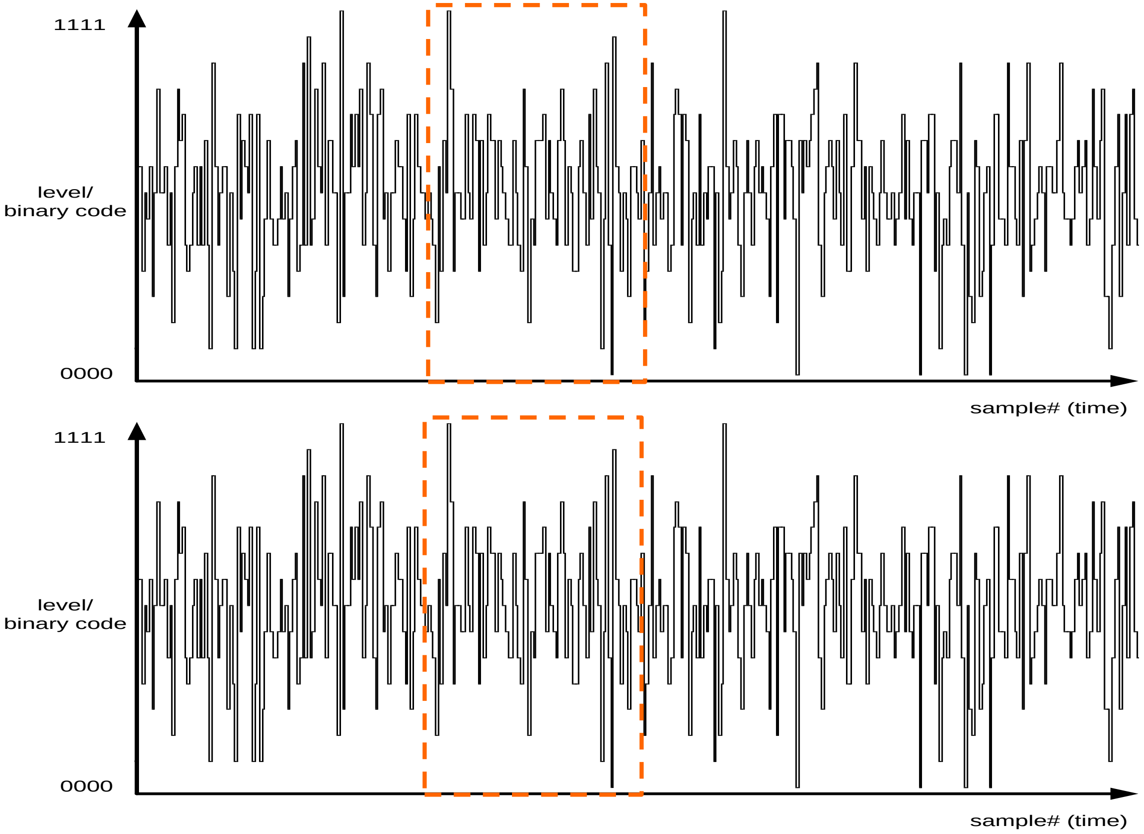
Righting the wrongs

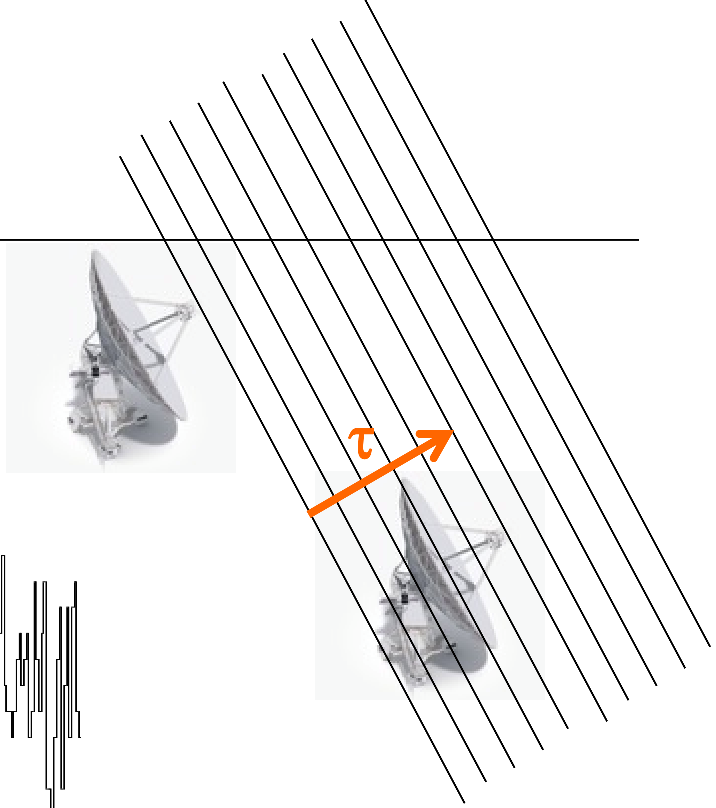


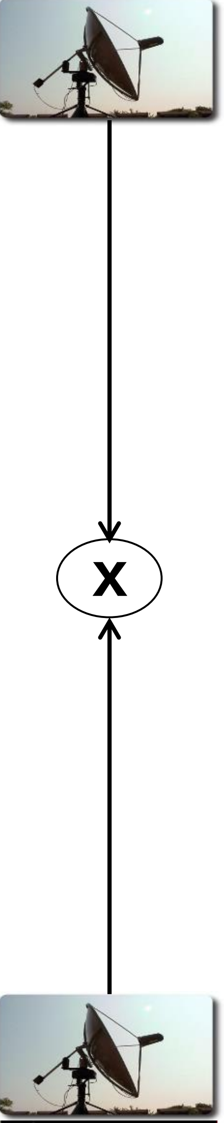


Delay compensation

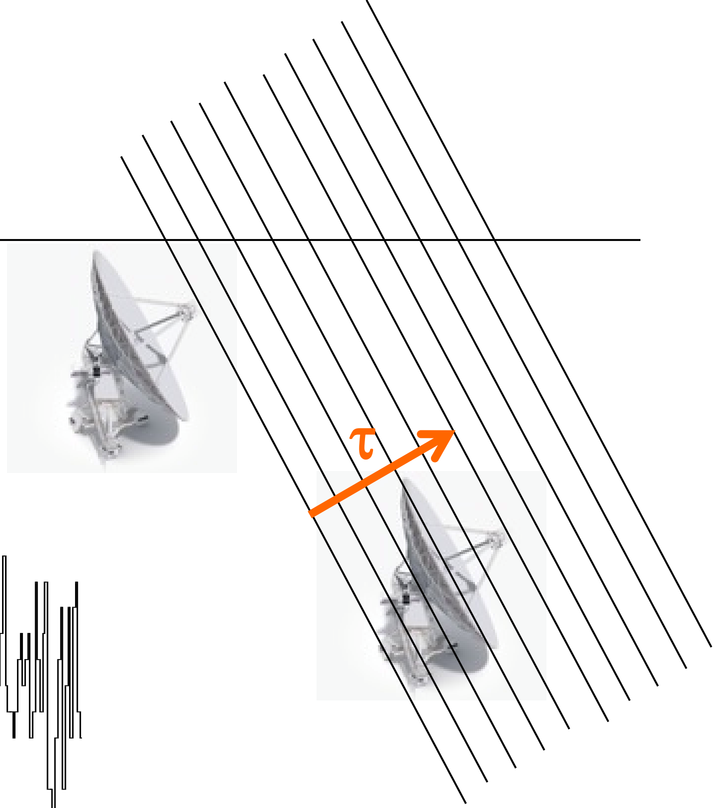
- Delay to the nearest sample is easy:



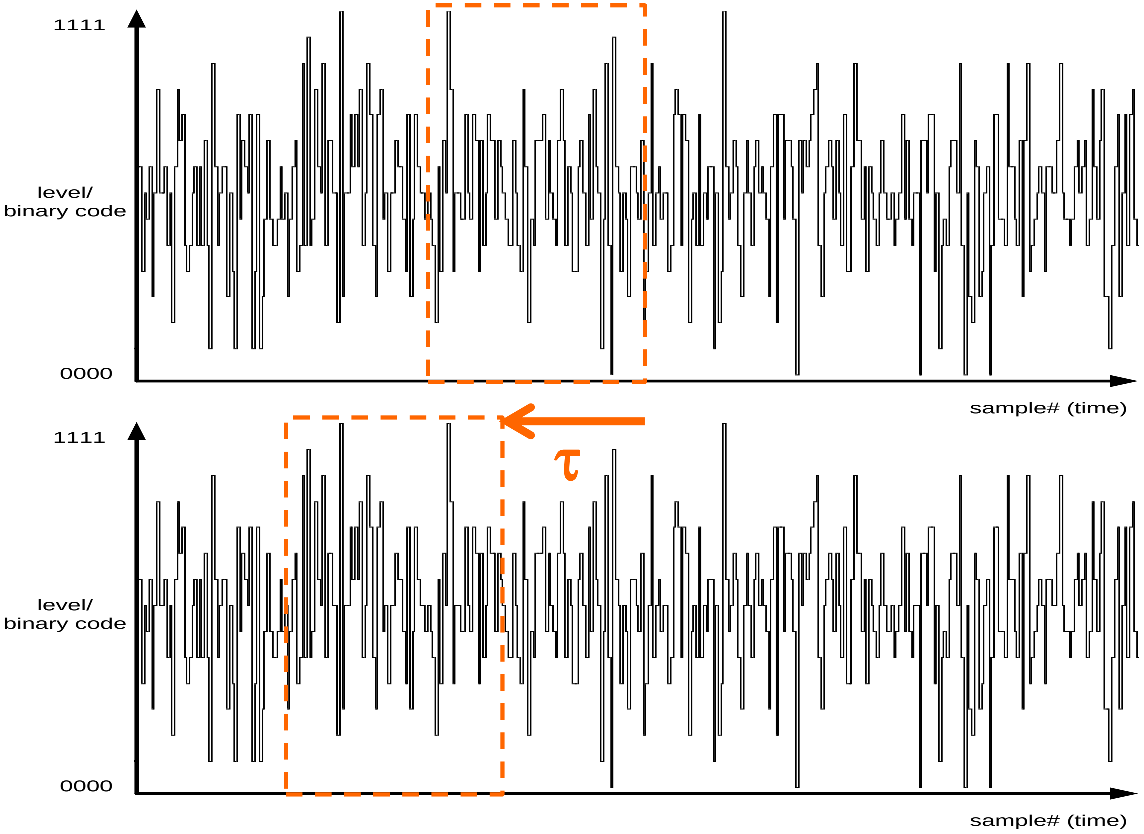




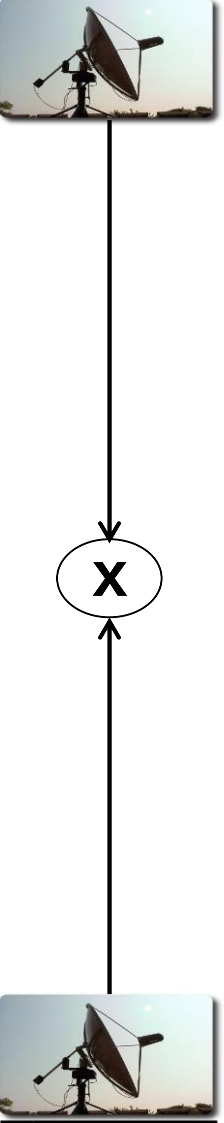
Delay compensation



- Delay to the nearest sample is easy:

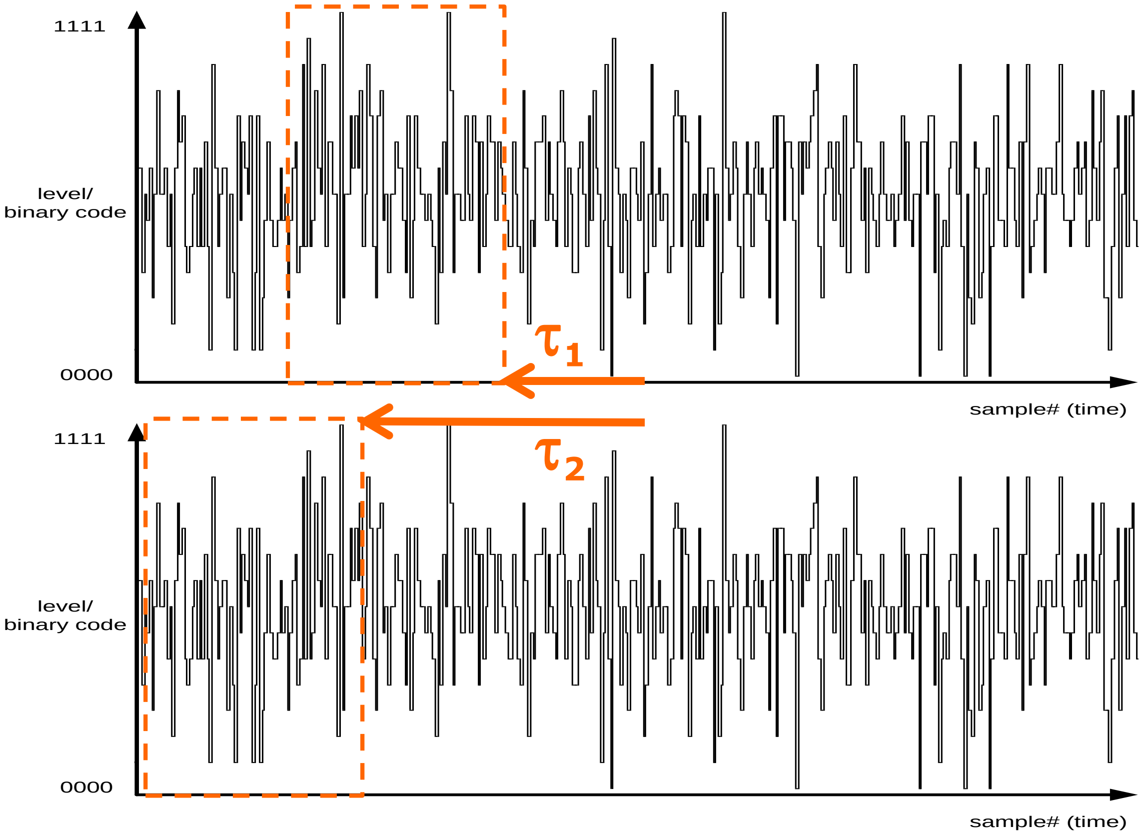


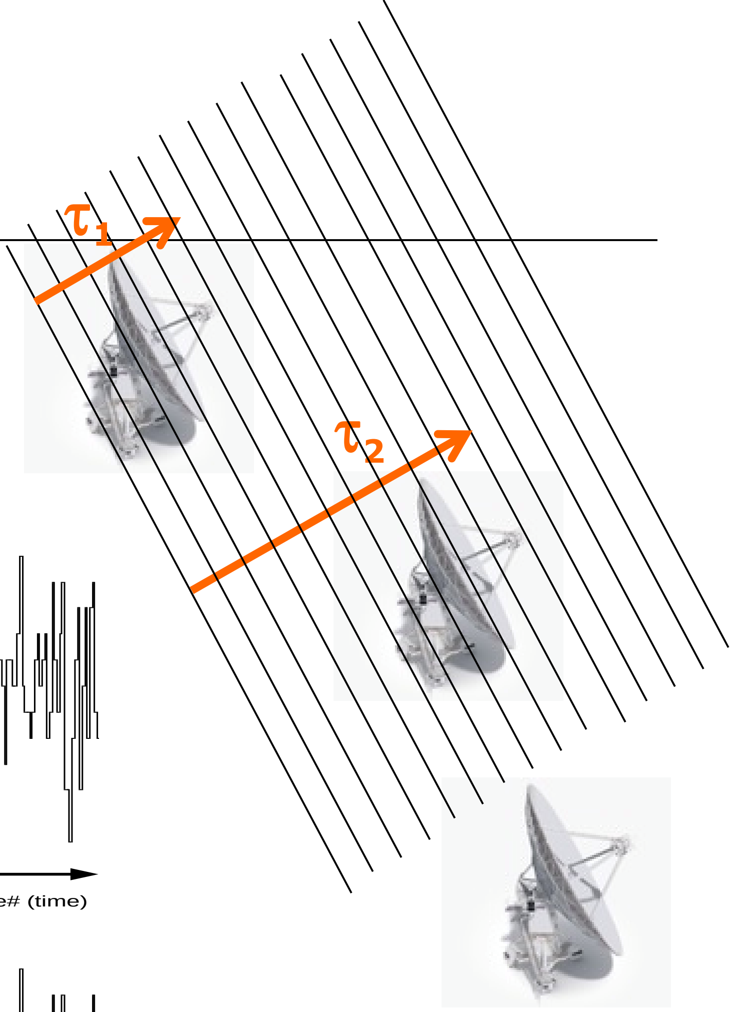
SWINBURNE UNIVERSITY OF TECHNOLOGY



Delay compensation

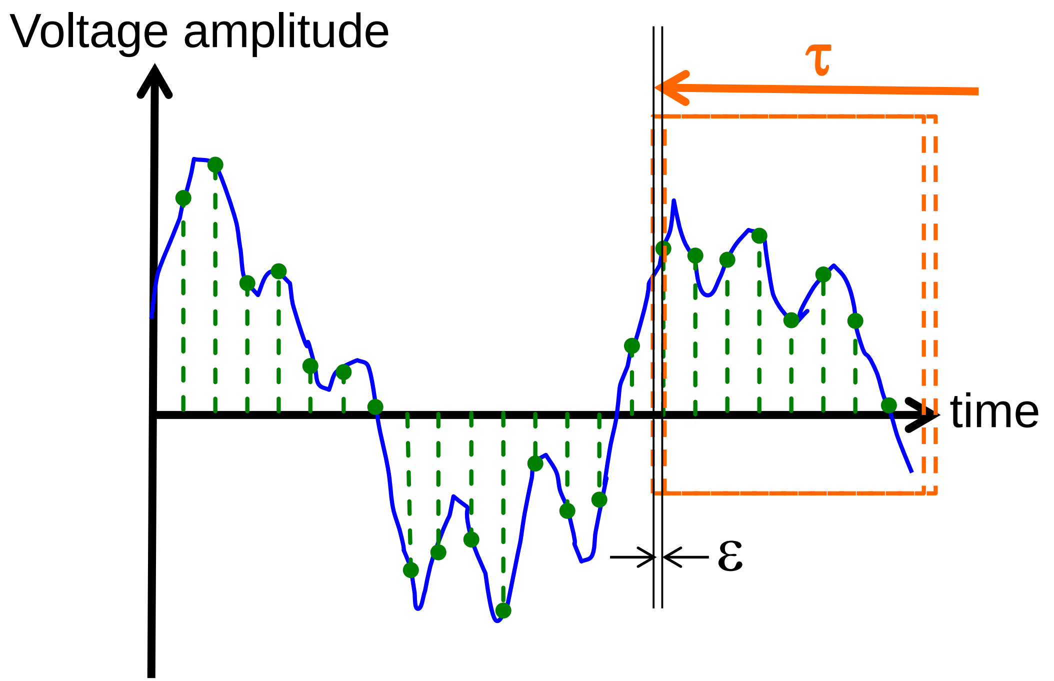
- In practise, delay all to common reference





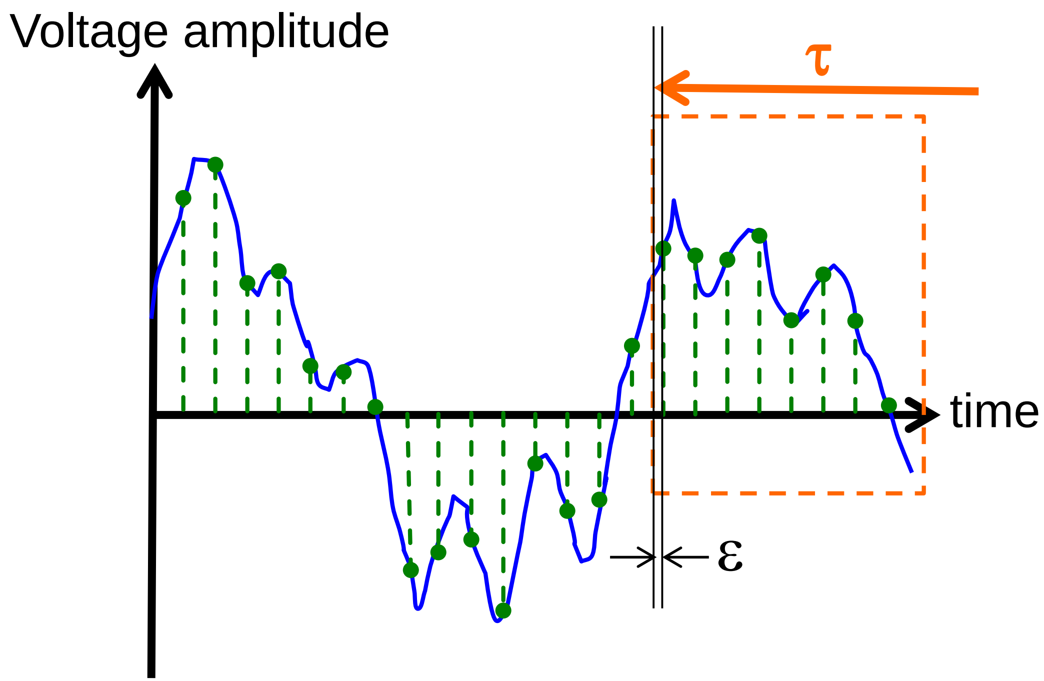
Fractional-sample correction

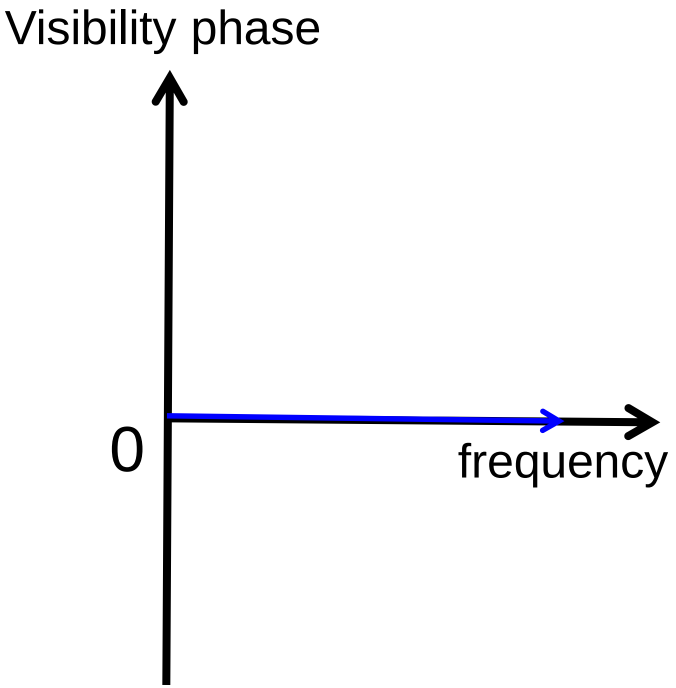
- Sampling prevents perfect alignment of datastreams; always a small error

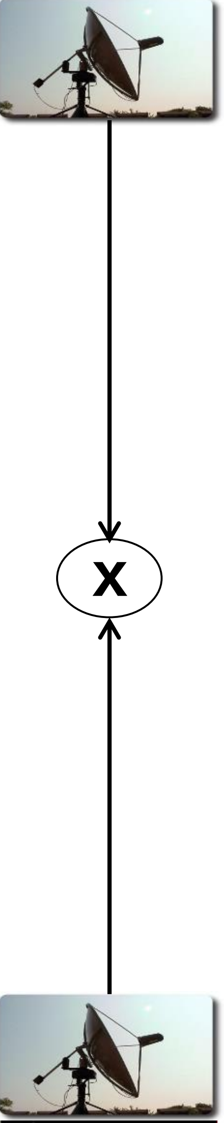


Fractional-sample correction

- Sampling prevents perfect alignment of datastreams; always a small error

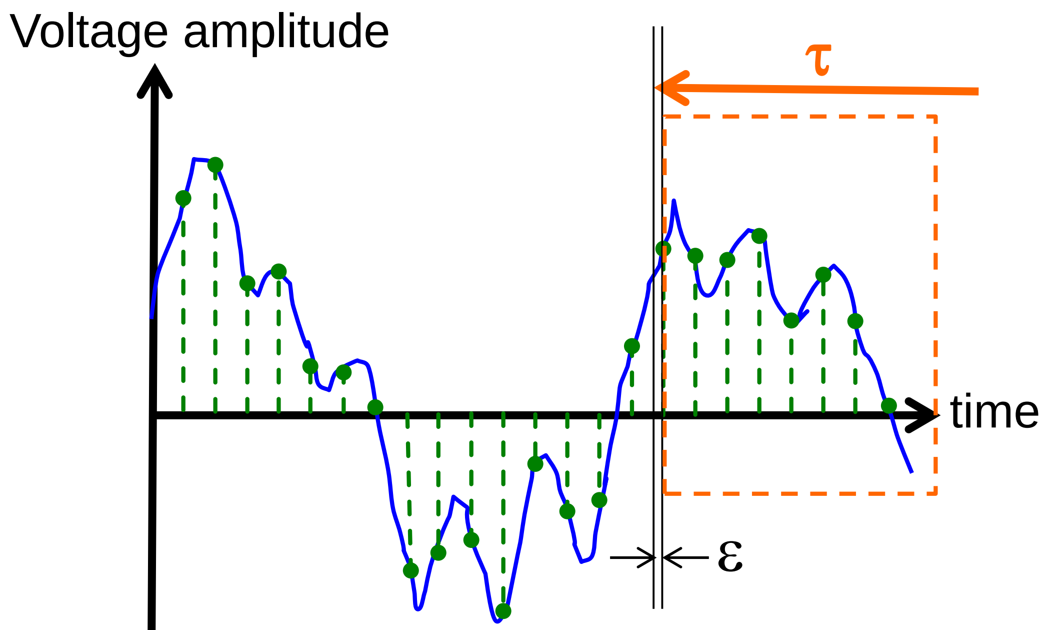


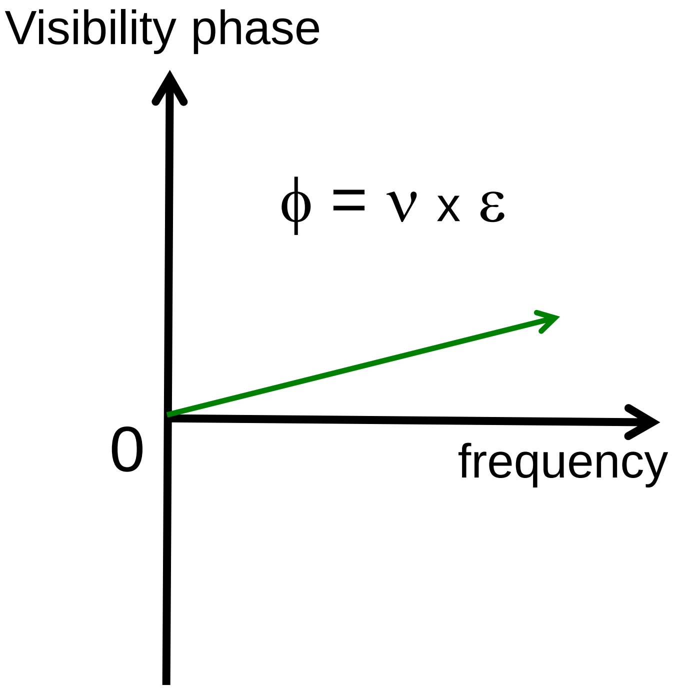




Fractional-sample correction

- Sampling prevents perfect alignment of datastreams; always a small error

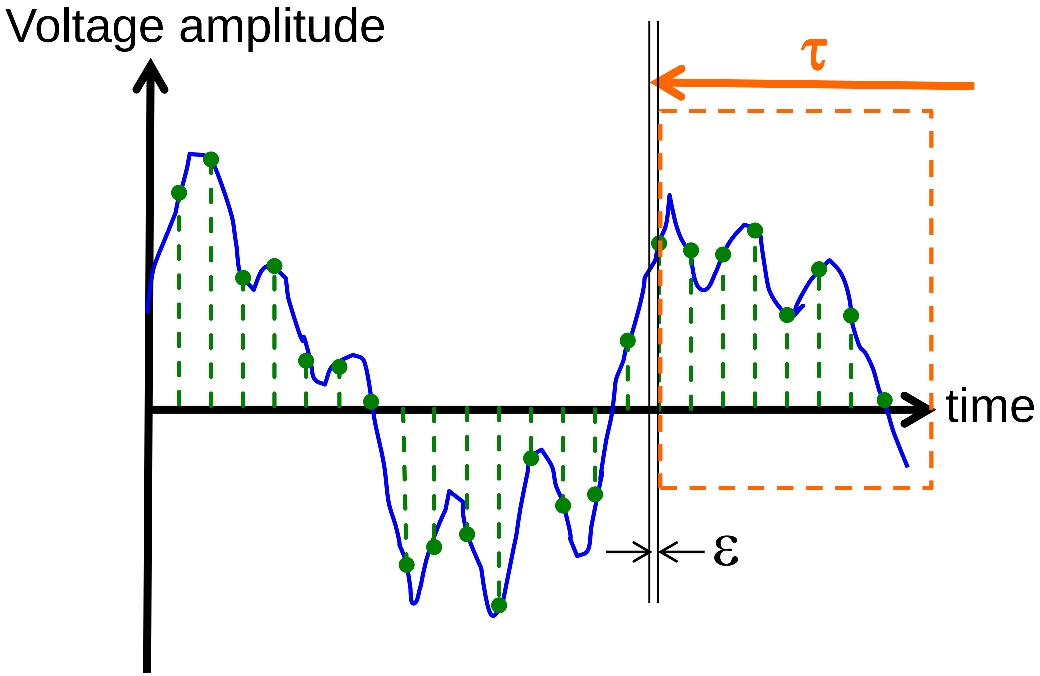


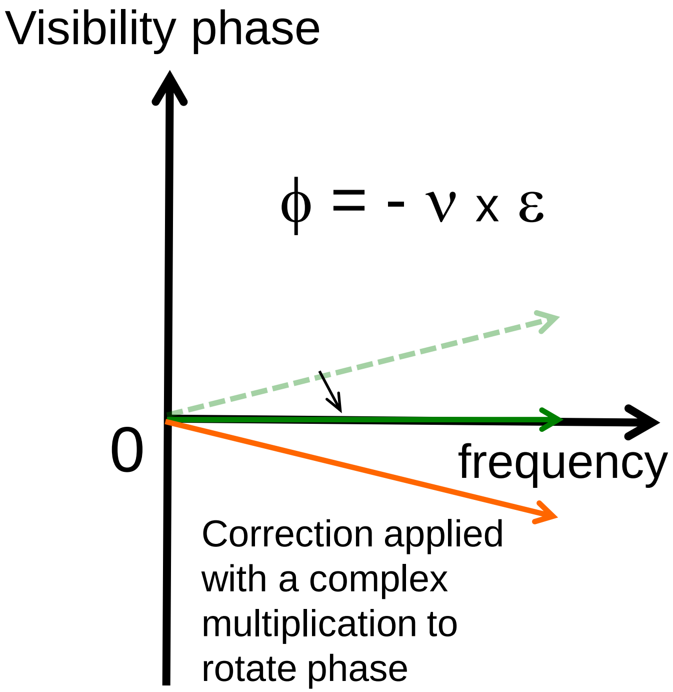


SWINBURNE UNIVERSITY OF TECHNOLOGY

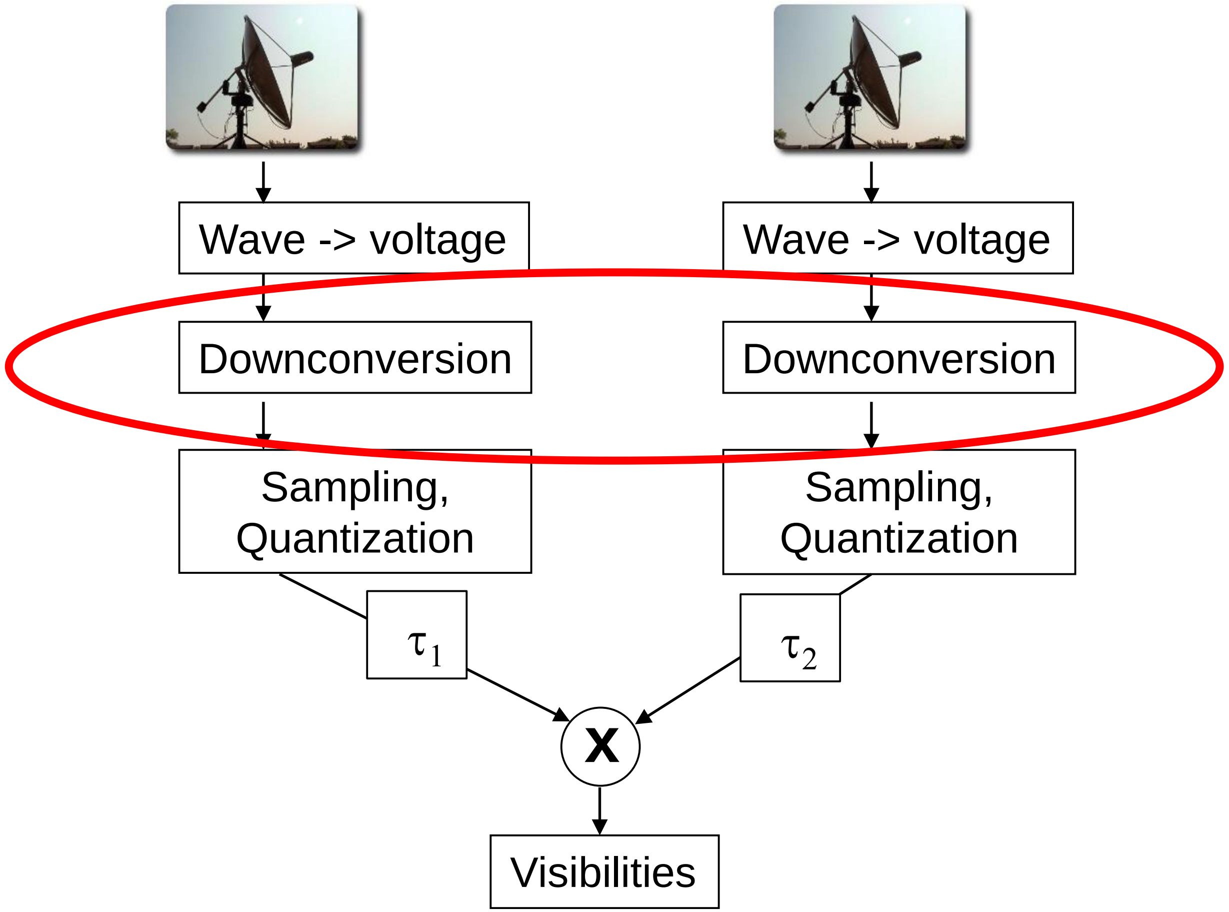
Fractional-sample correction

- Sampling prevents perfect alignment of datastreams; always a small error

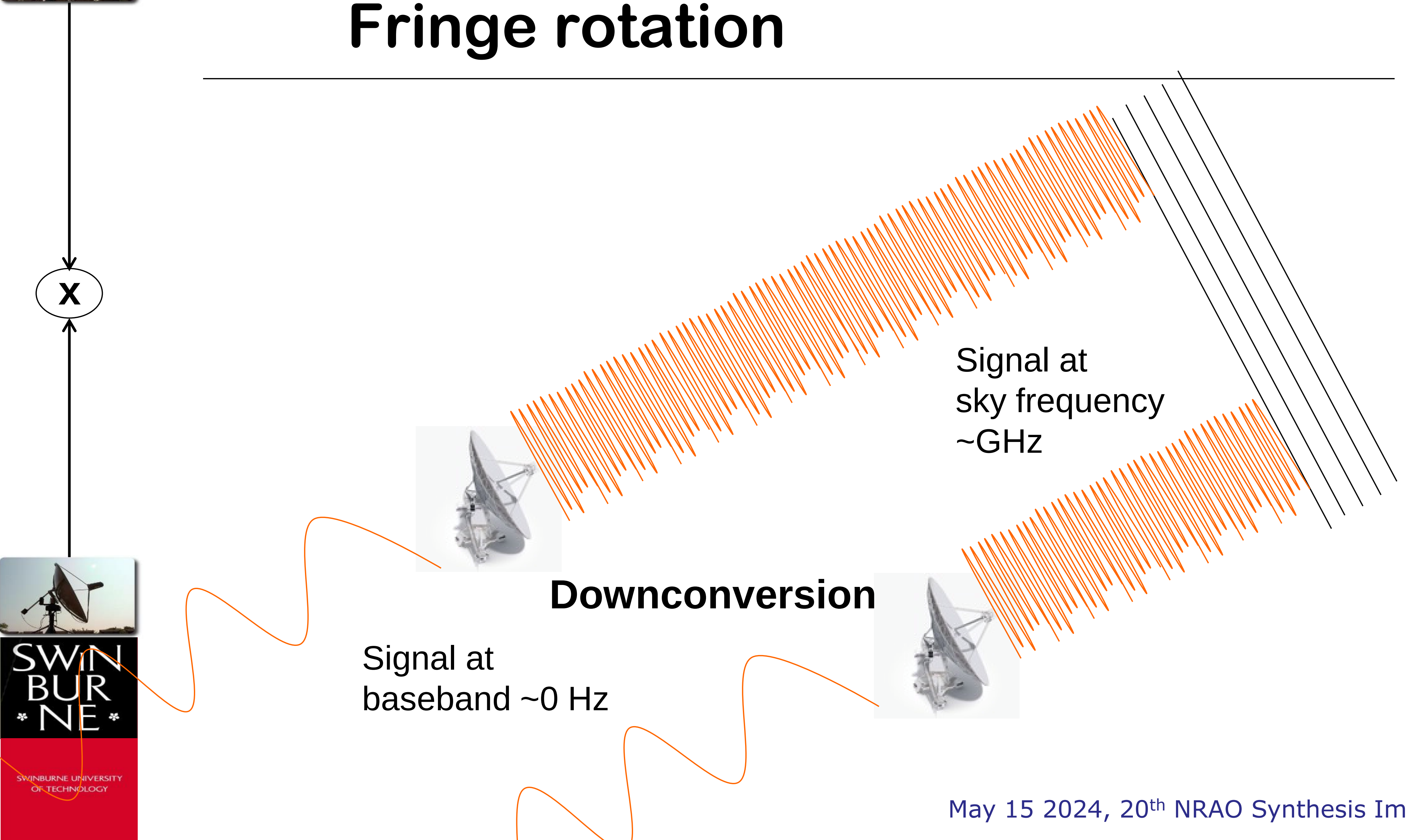


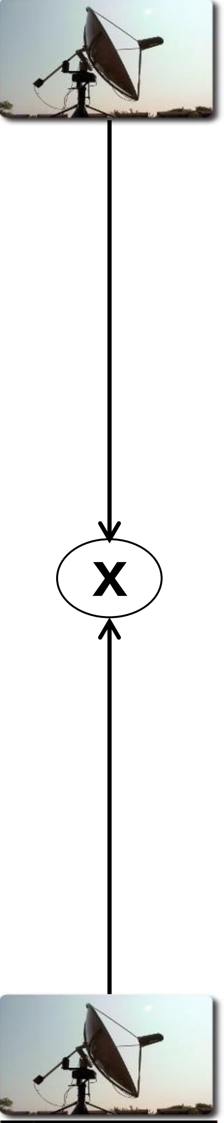


Righting the wrongs



Fringe rotation





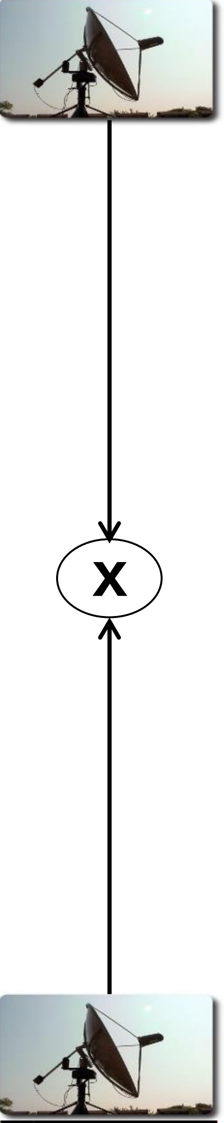
Fringe rotation

- Implementation: treat voltage as a complex number, rotate phase via complex multiplication
- $\Delta\phi = 2\pi \times \nu_{lo} \times \tau_g$

ν_{lo} = local oscillator frequency (how much the signal was shifted in frequency);
τ_g = applied delay

- How often do we need to recompute $\Delta\phi$?:
  - If τ_g is changing fast, correct every recorded sample individually (before the FFT)
  - For shorter baseline instruments (or at lower frequencies, where ν_{lo} is much smaller), can do less frequently – post-channelisation or even post-accumulation

SWINBURNE UNIVERSITY OF TECHNOLOGY

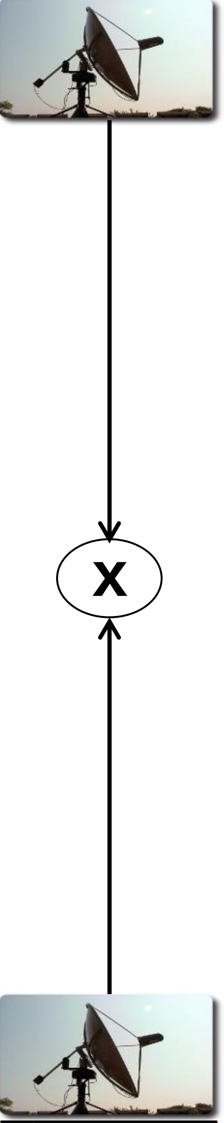


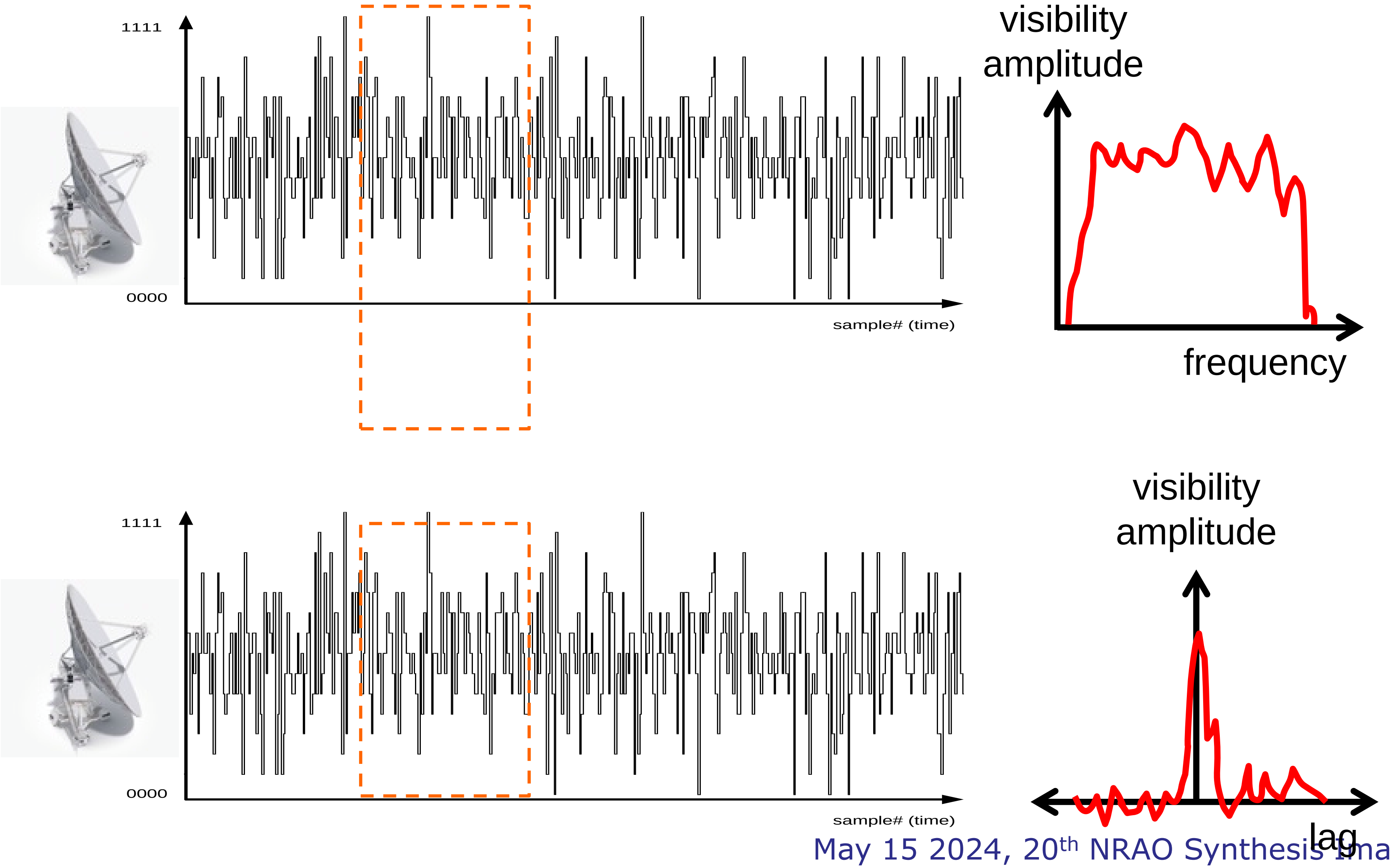
Alternate implementation

- We have shown how to build a practical FX correlator, which first Fourier transforms and then multiplies
- Convolution theorem: **Multiplication** in the frequency domain is equivalent to **convolution** in the time domain
- It is mathematically equivalent to convolve the two signals in the time domain and then Fourier transform

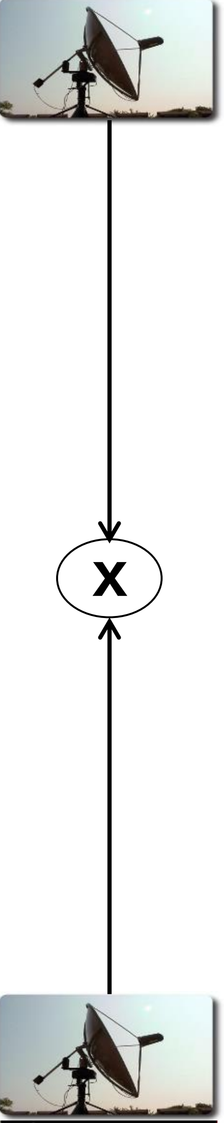
SWINBURNE UNIVERSITY OF TECHNOLOGY

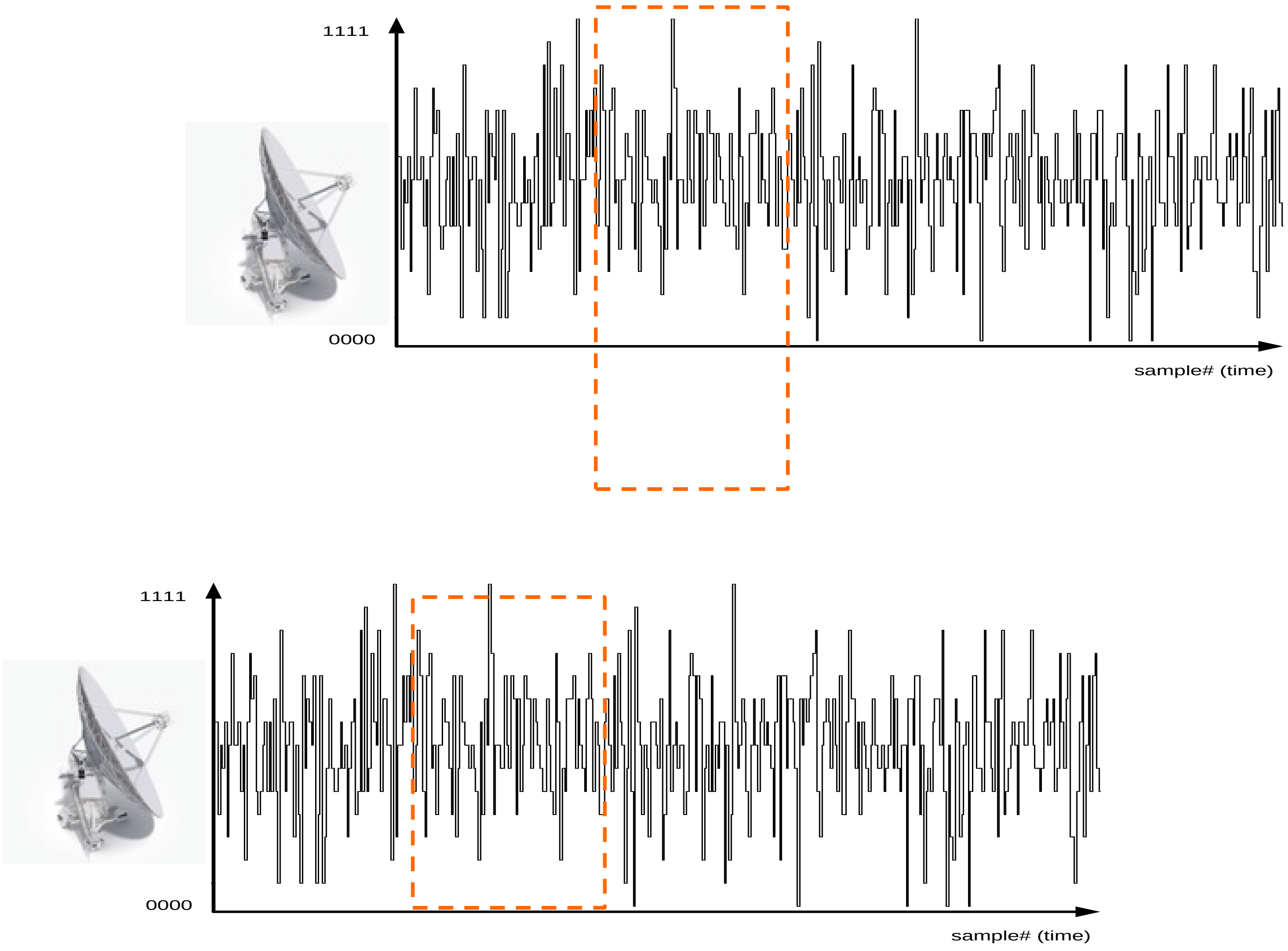
An equivalent "XF" correlator



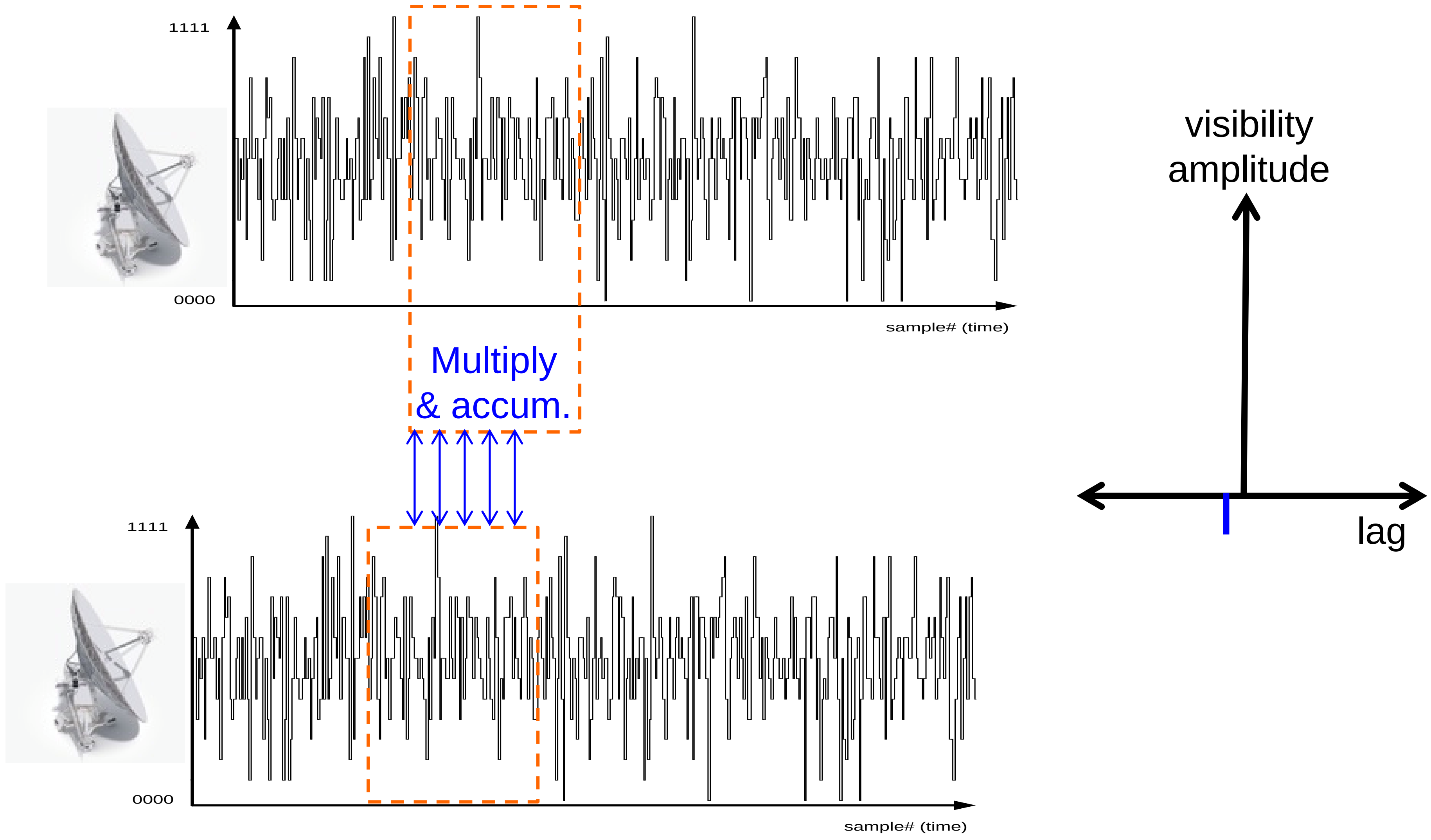


An equivalent "XF" correlator

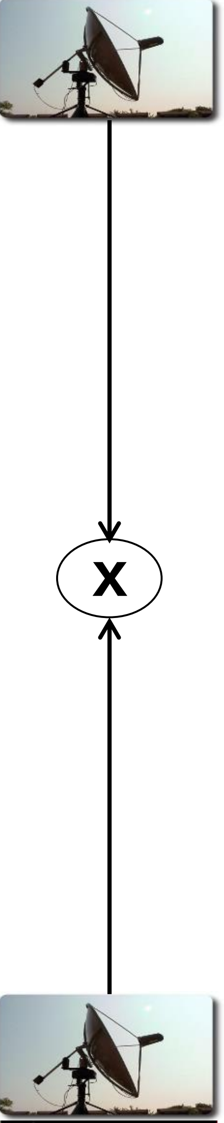


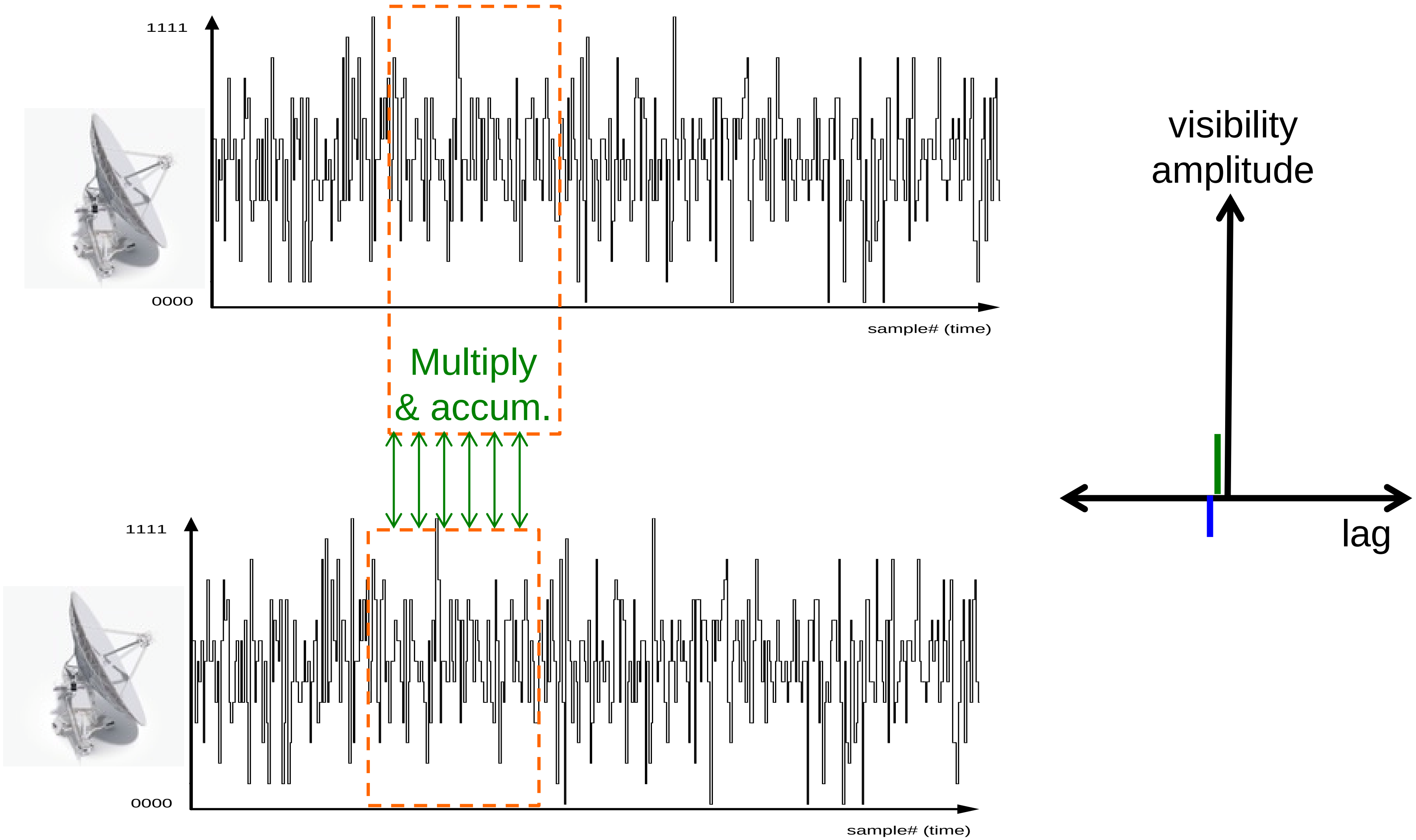


An equivalent “XF” correlator

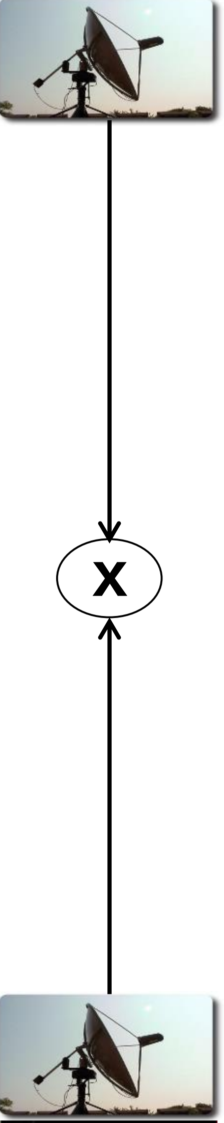


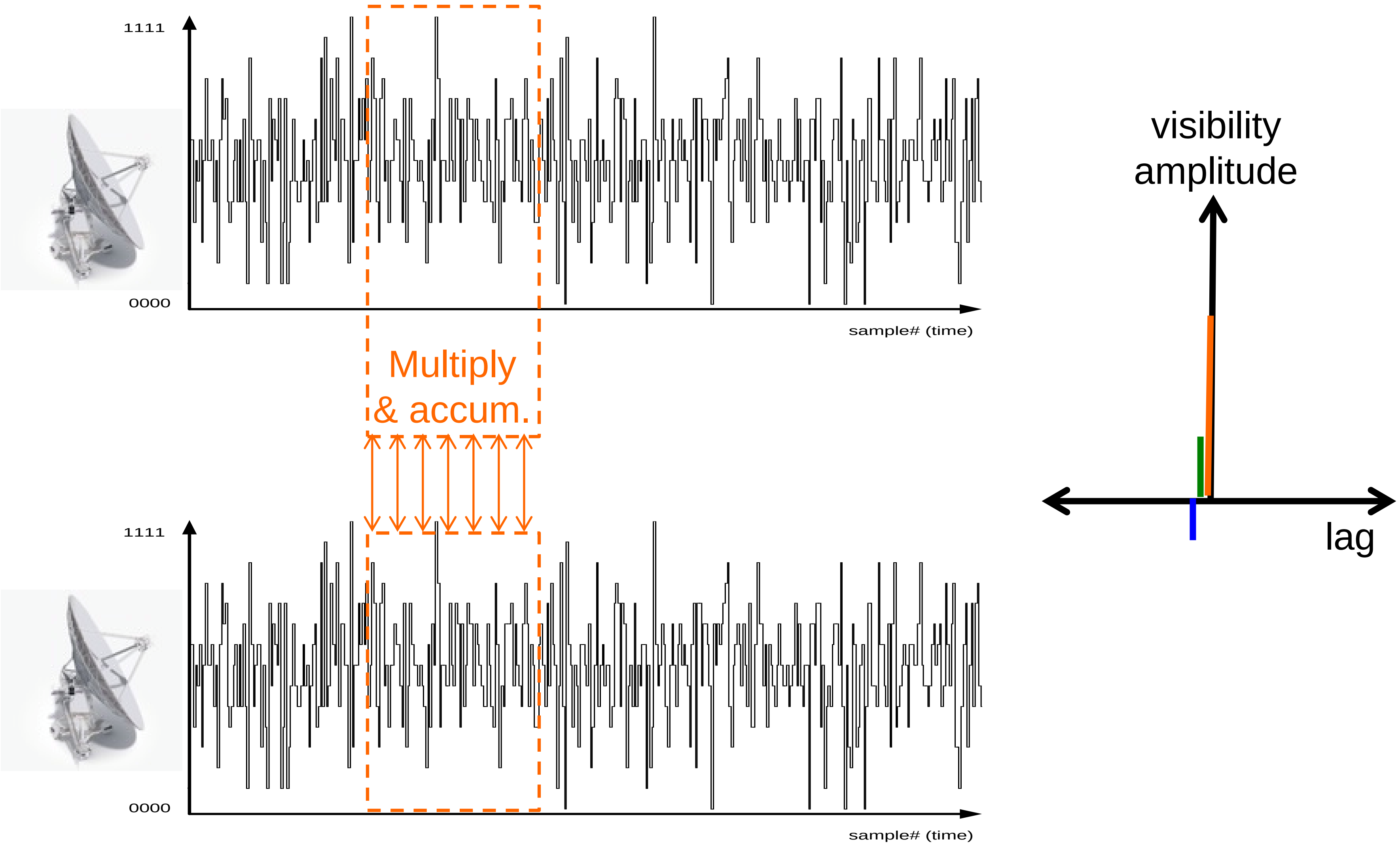
An equivalent "XF" correlator

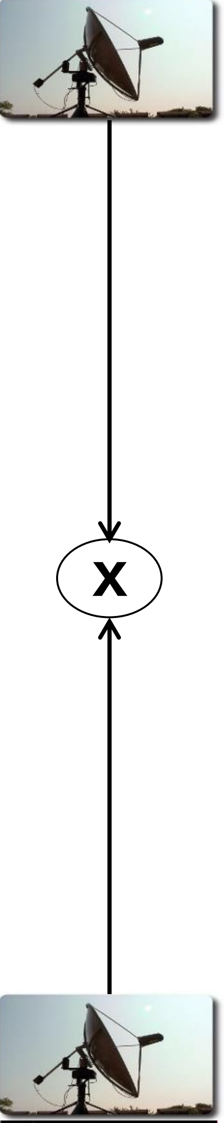




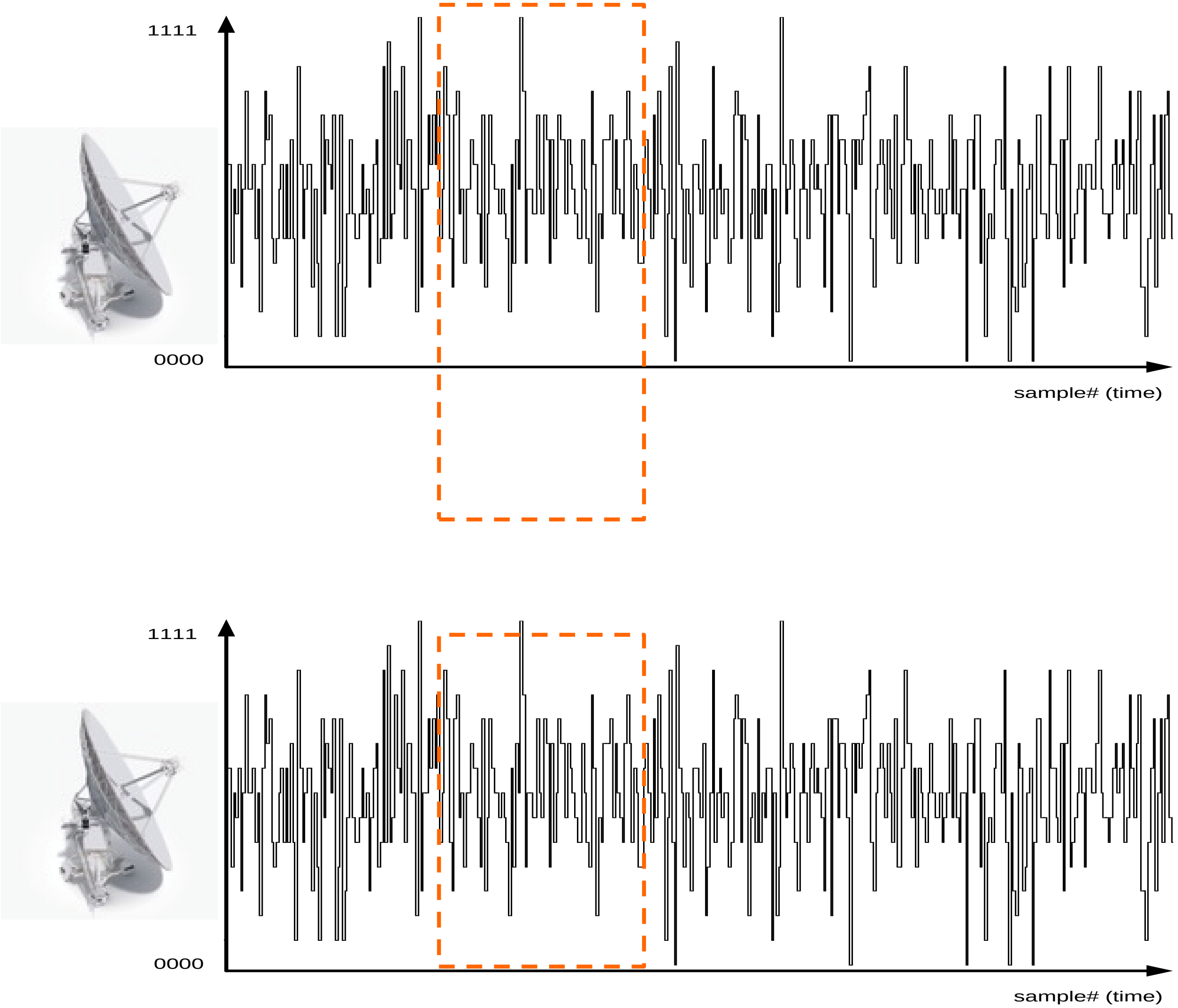
An equivalent "XF" correlator

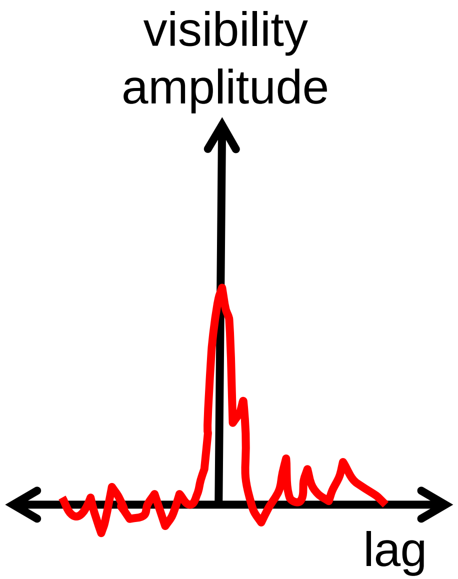






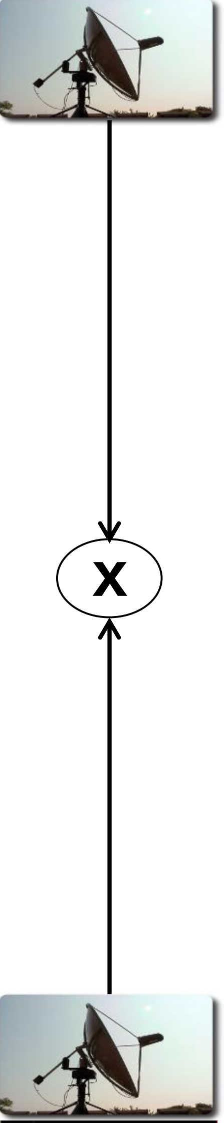
An equivalent “XF” correlator

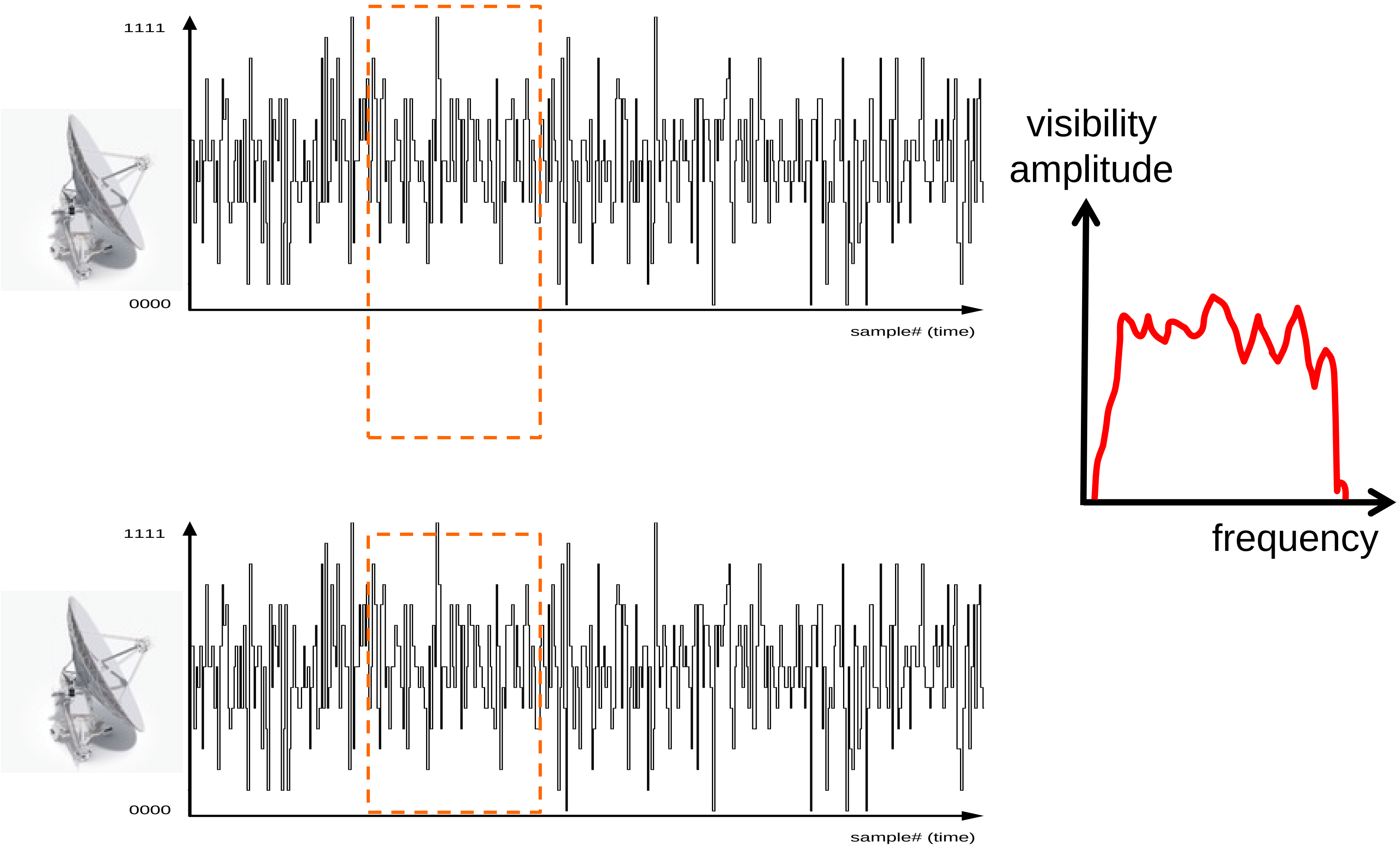




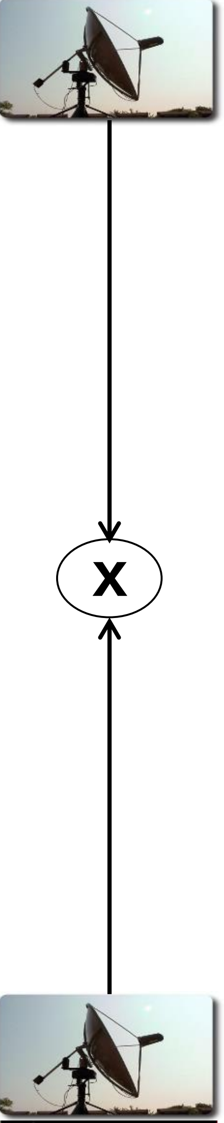
SWINBURNE UNIVERSITY OF TECHNOLOGY

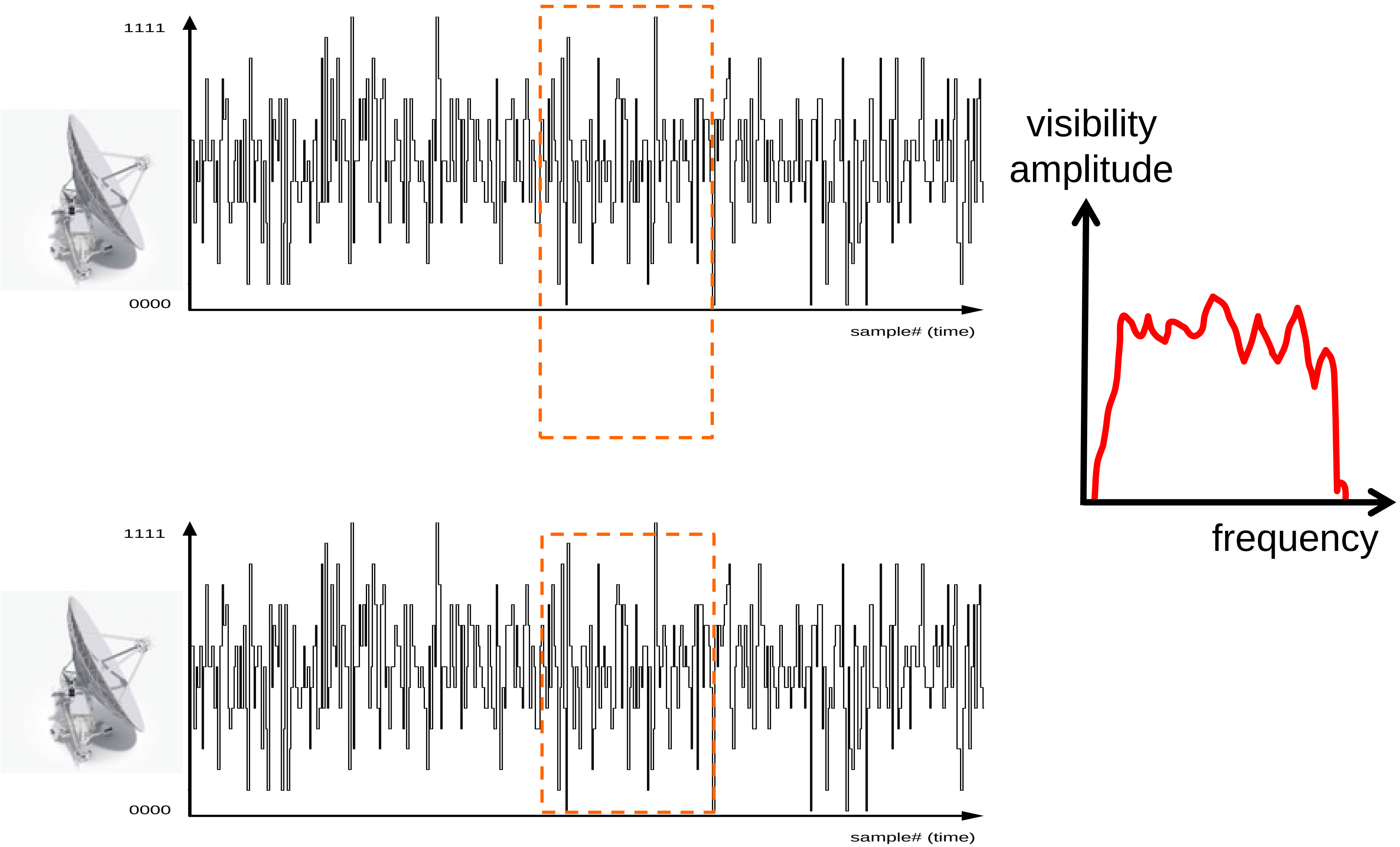
An equivalent "XF" correlator



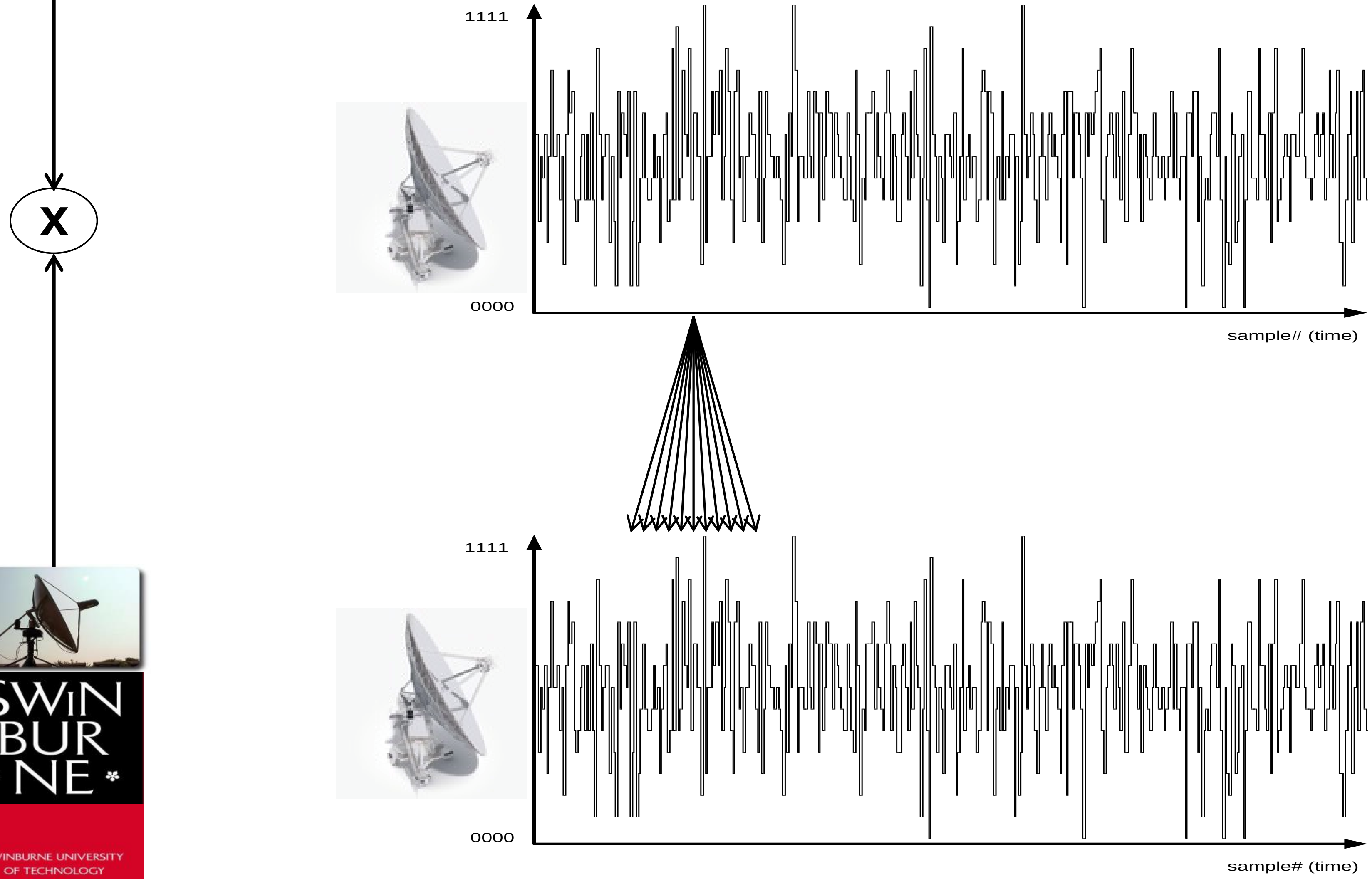


An equivalent “XF” correlator

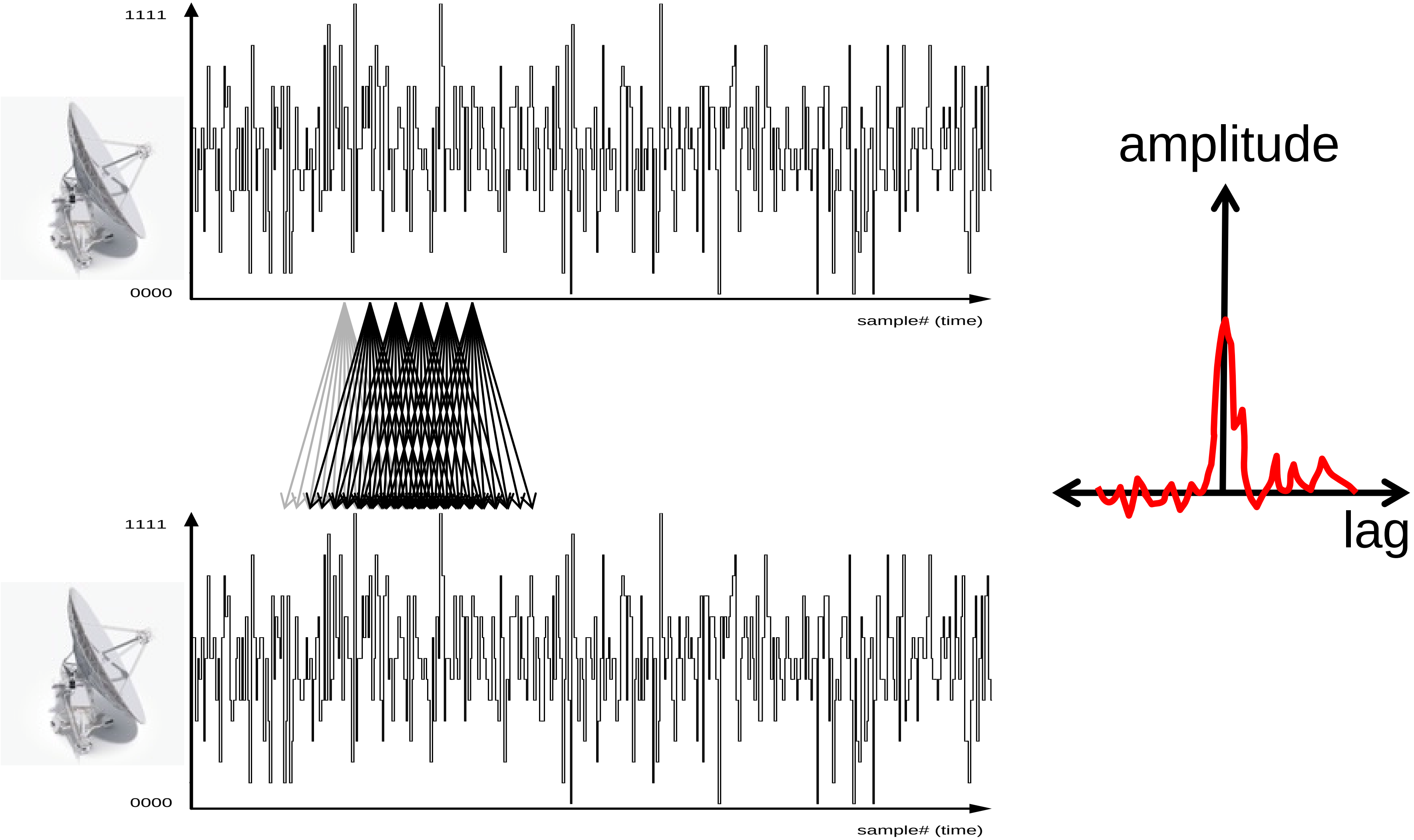




A realistic XF correlator

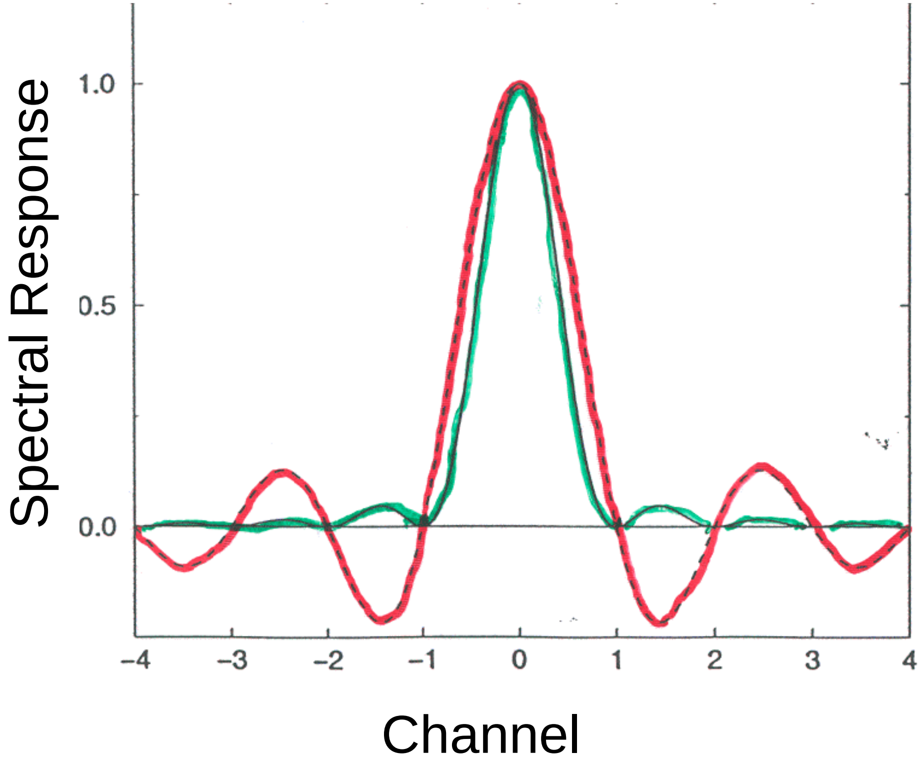


A realistic XF correlator

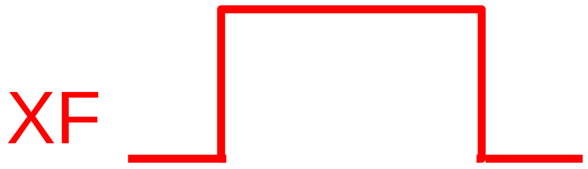


XF vs FX

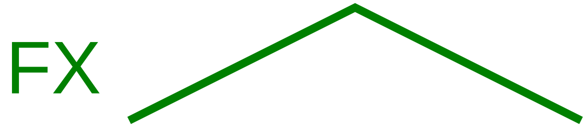
- Different windowing in time domain gives different spectral response



22% sidelobes! Reduce with Hanning smoothing

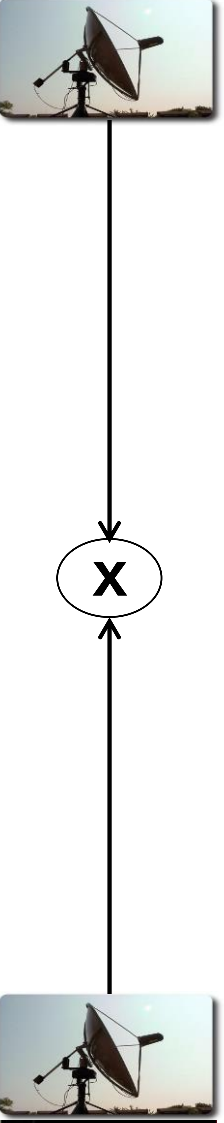


5% sidelobes. If possible, produce higher spectral resolution and then average.



Lag weighting

Modern FX correlators may use overlapping, non-boxcar window functions, even polyphase filterbanks: all give different lag weightings, ask later for details.

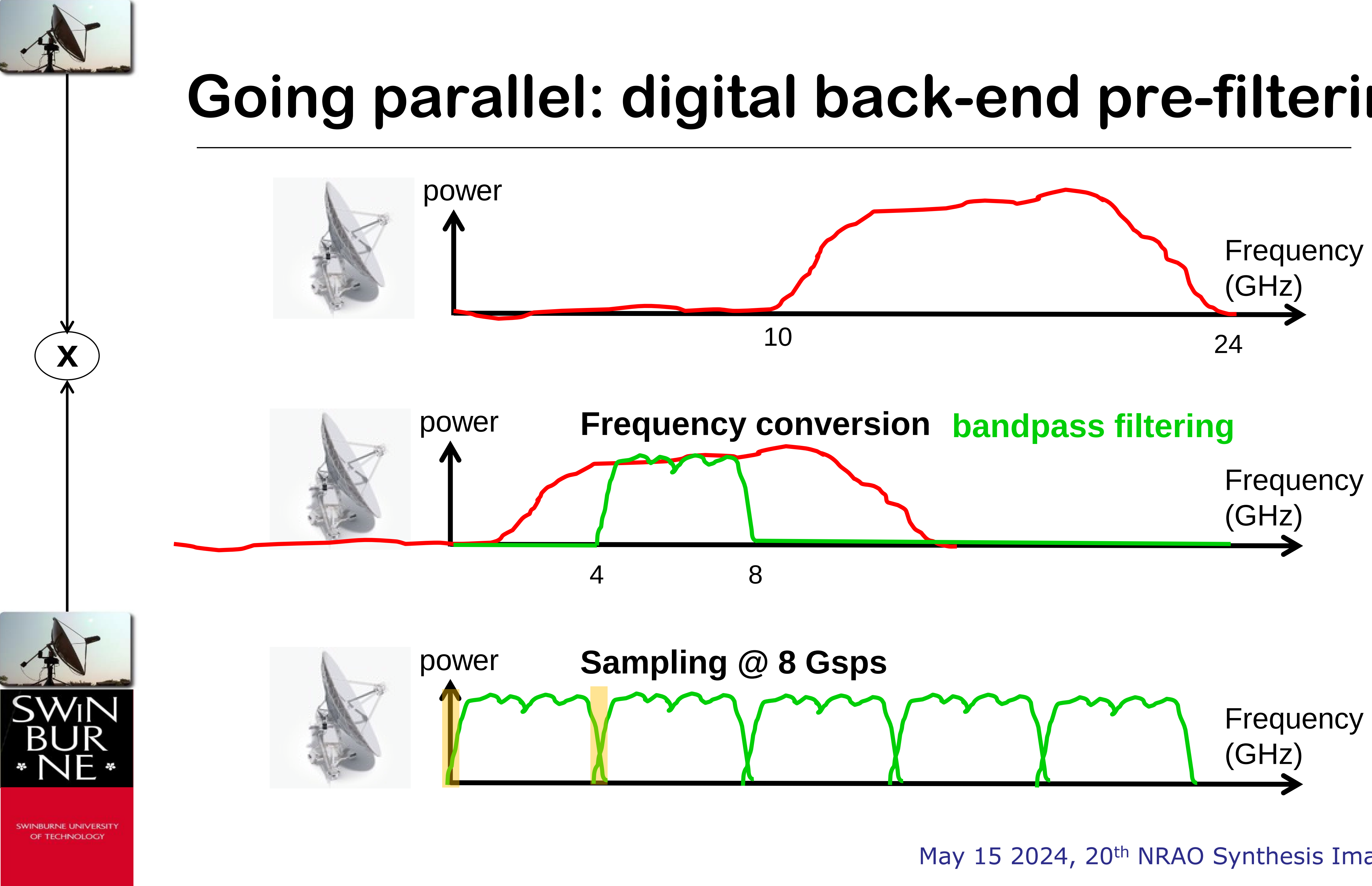


XF vs FX: which is better?

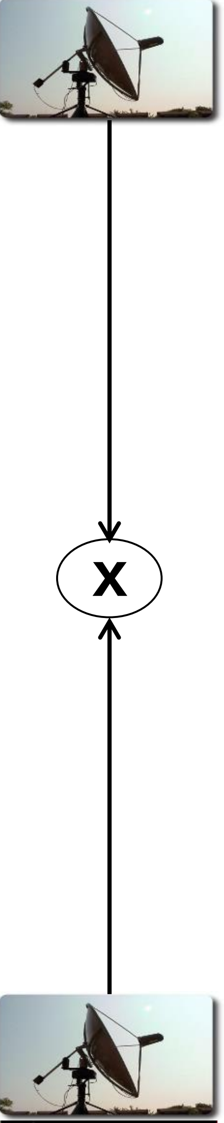
- Desire for reduced artifacts favours FX
  - Main advantage of XF: can use very efficient low-precision integer multipliers up-front
  - But FX many fewer operations overall, unaffected by trend to higher bit depth
  - FX also: access to frequency domain at short timescale allows neat tricks and higher precision correction of delay effects
  - Modern correlators mostly FX-style, and often have multiple cascaded filter steps (~GHz recorded band chopped into ~100 MGz chunks and correlated separately)

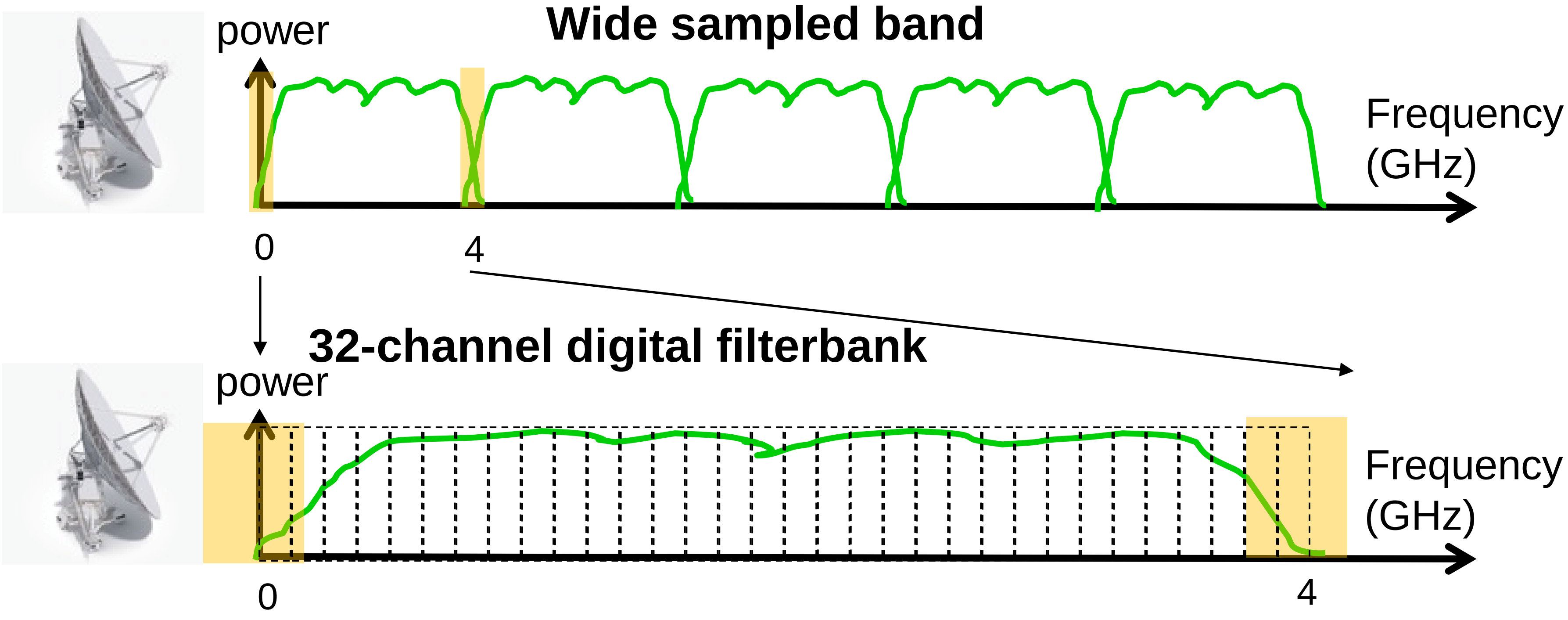
SWINBURNE UNIVERSITY OF TECHNOLOGY

Going parallel: digital back-end pre-filtering

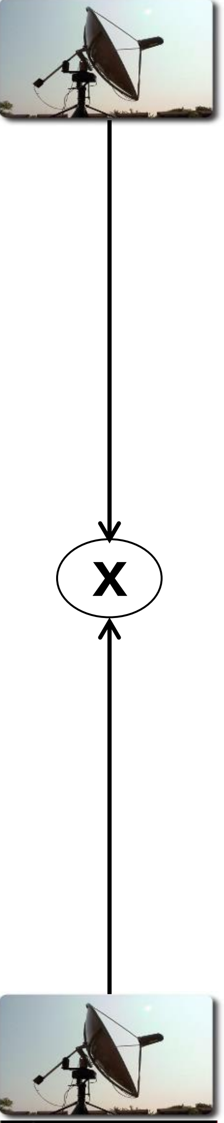


Going parallel: digital back-end pre-filtering





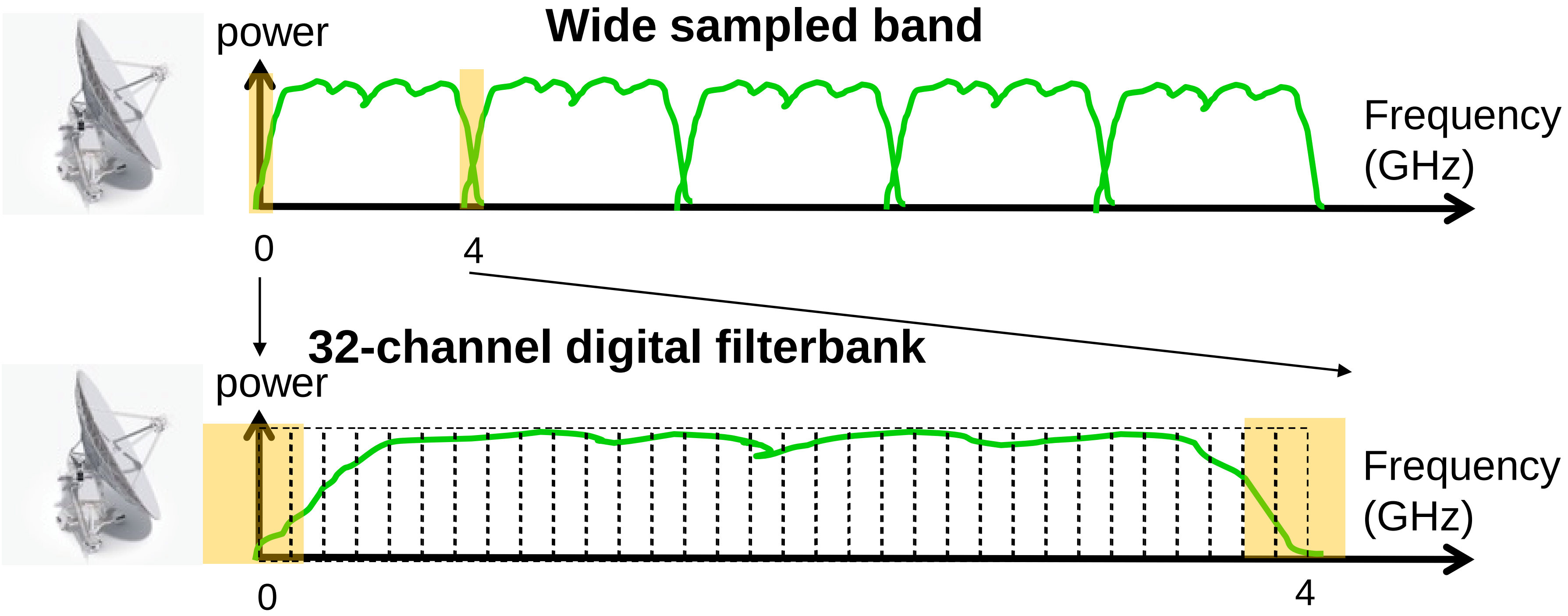
Each of these 128 MHz chunks can then be treated by separate FX (or XF!) style correlator in parallel: fringe rotation, channelization, delay compensation, and cross-multiplication



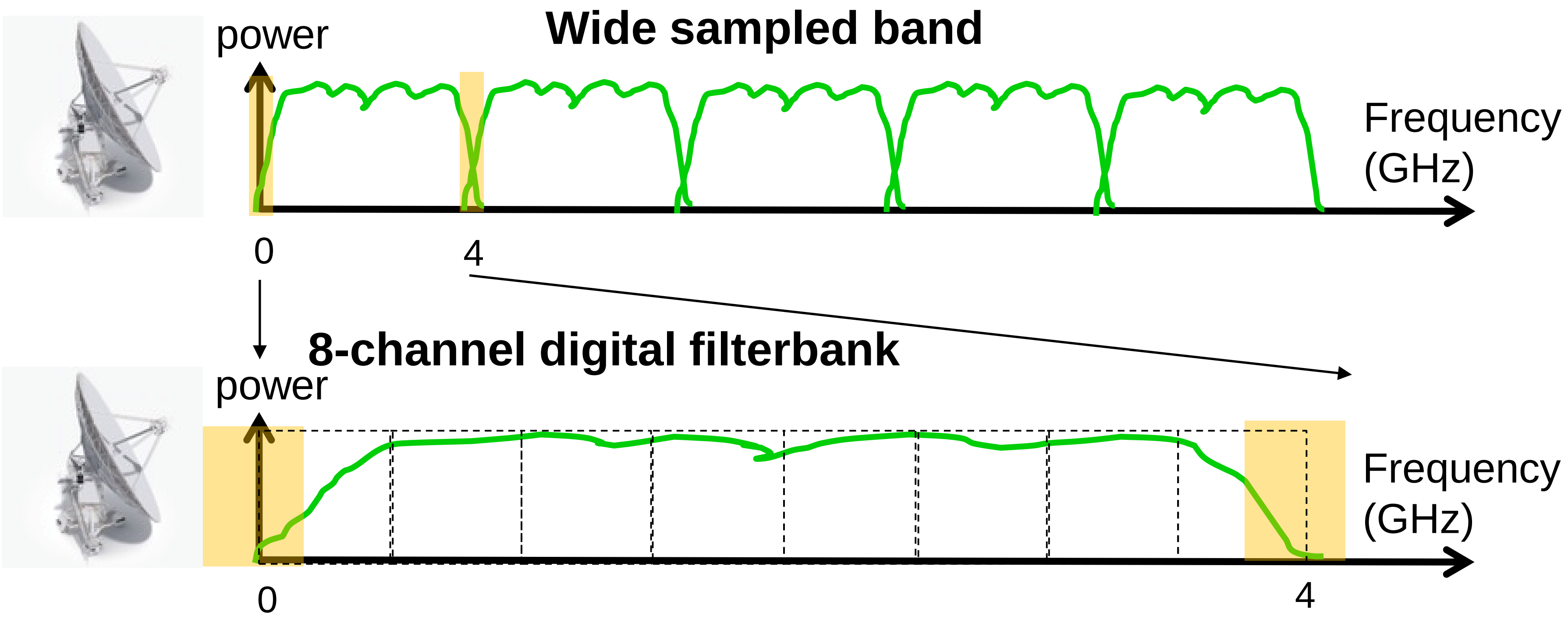
Filter it your way

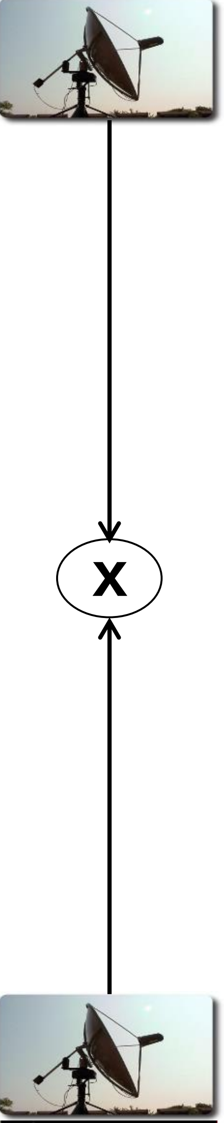
- Once the voltage signal is digitized, until we detect it (multiply it against another voltage), we can slice it and dice it however we like! Just keep track of the losses/amplitude reductions to correct later.
- Benefits:
  - Parallelisation (maps well to hardware, more in a minute)
  - Spectral isolation (keep RFI in its place)
  - Re-quantisation – save on bits! (Enabled by RFI isolation)
- Digital filterbank outputs are natively complex signals (real + imaginary) - *ask in the discussion if you're interested in analytic signal representations.*

Filter it your way

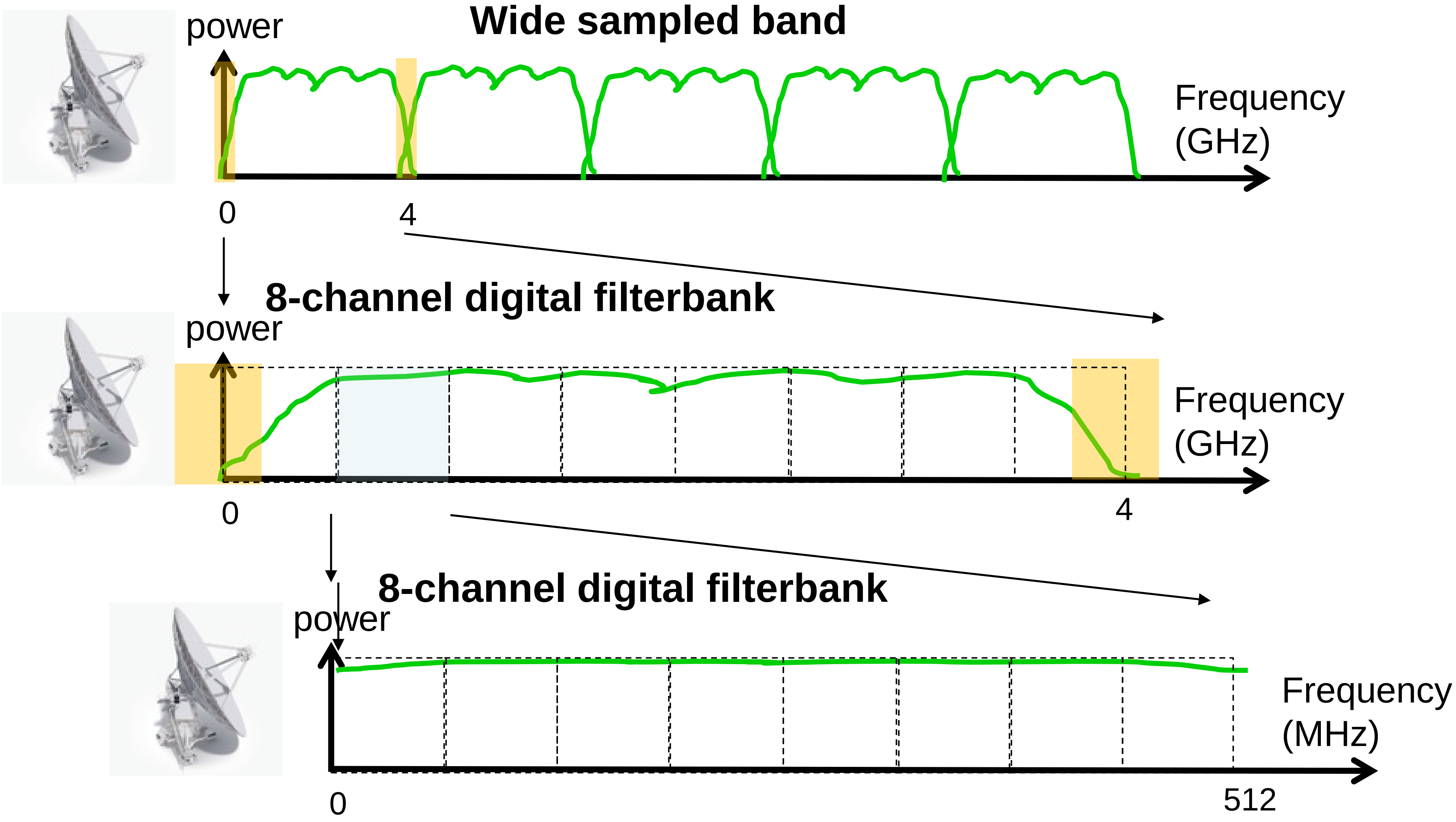


Filter it your way

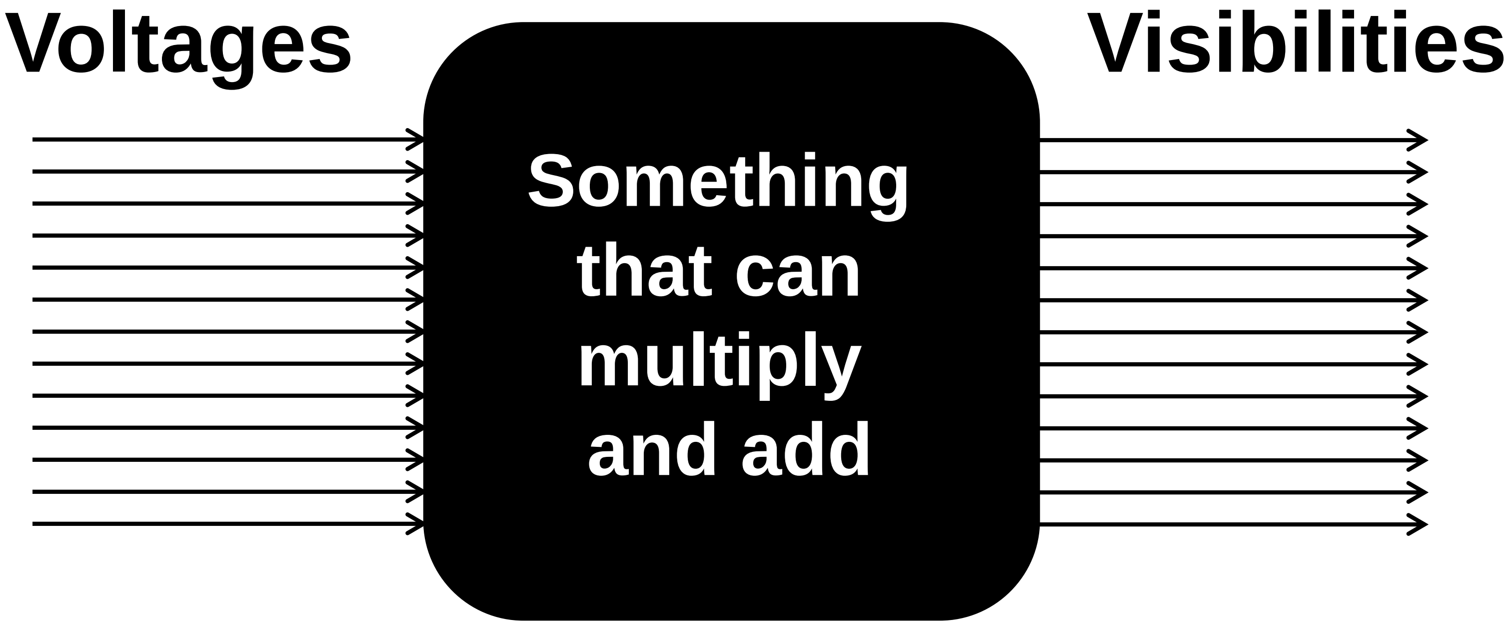


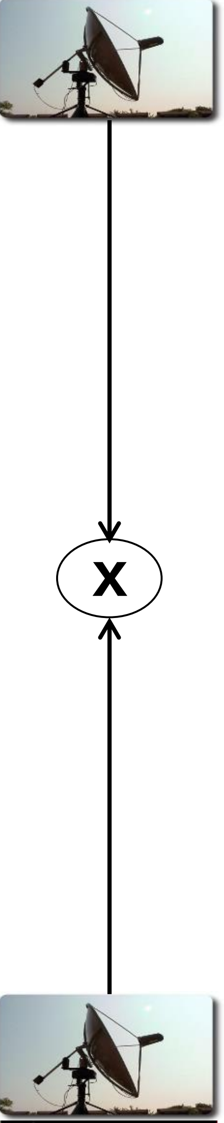


Filter it your way



Correlator platforms



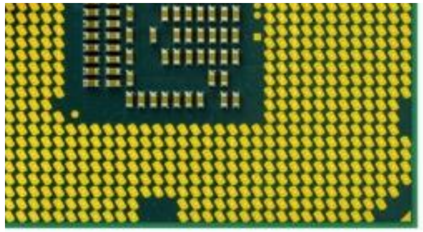


Correlators on CPUs

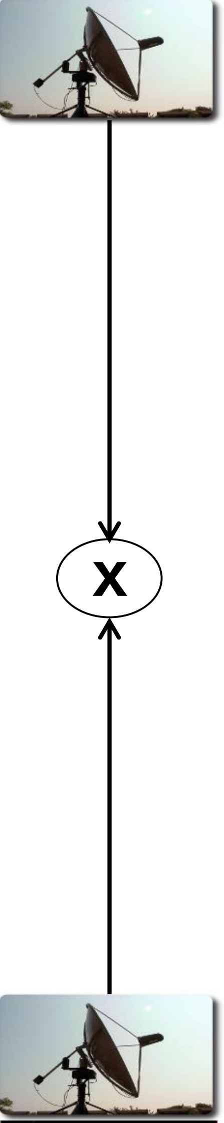

```
status = vectorFFT_CtoC_cf32(complexunpacked, fftd, pFFTSpecC, fftbuffer);
if(status != vecNoErr)
  csevere << startl << "Error doing the FFT!!!" << endl;

      ...

status = vectorAddProduct_cf32(vis1, vis2, &(scratchspace->threadcrosscorrs[resultindex+outputoffs
```



SWINBURNE UNIVERSITY OF TECHNOLOGY



Correlators on CPUs

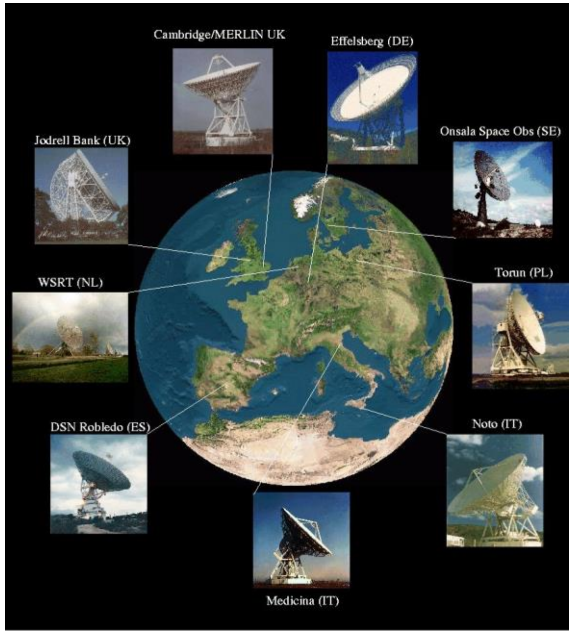
- Many positive points:
  - Can implement in “normal” code (e.g., C++); maintainable, many skilled coders
  - Development effort transferrable across generations of hardware
  - Incremental development is trivial
  - Natively good at floating point (good for FX), no cost to do high precision
- One major disadvantage:
  - CPUs not optimised for correlation; teraflop++ systems would take **many** CPUs.

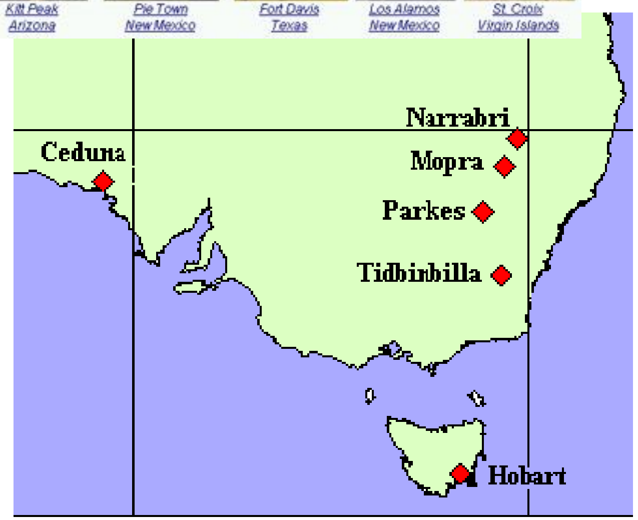
SWINBURNE UNIVERSITY OF TECHNOLOGY

Correlators on CPUs

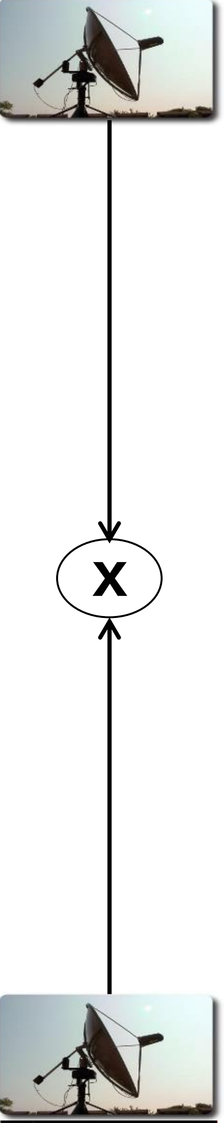
The Very Long Baseline Array, 10 stations

The European VLBI Network, ~30 stations





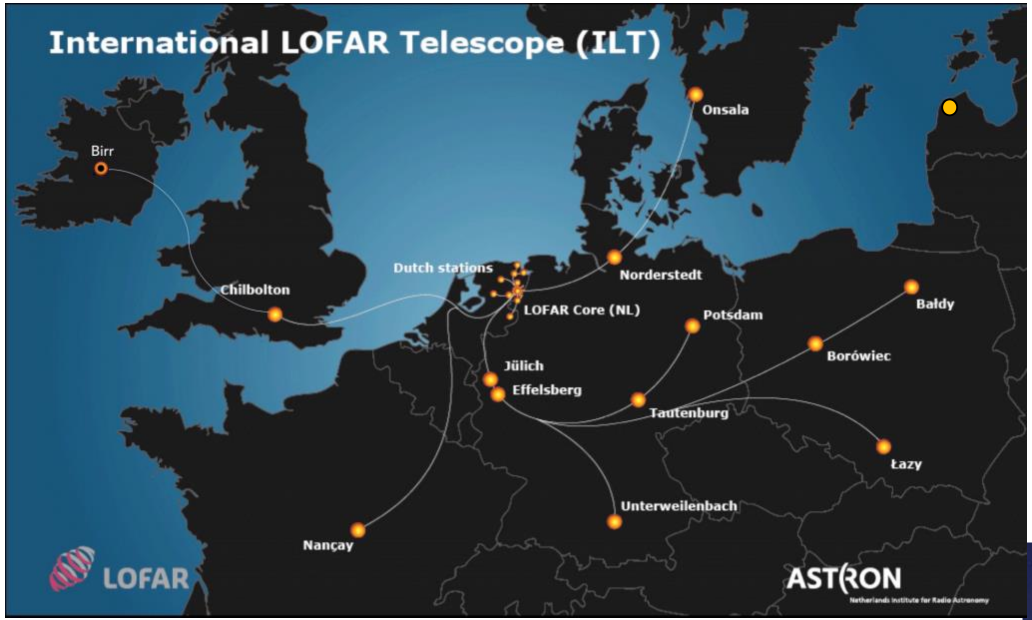
The Long Baseline Array, Australia, ~6 stations



Correlators on GPUs

- Advantages:
  - More powerful/efficient than CPUs
  - Also good at floating point
  - Made for vector maths!
- Disadvantages:
  - Writing code is more difficult (GPUs are more specialized, less flexible: need to carefully manage data transfers)
  - Fewer expert GPU programmers available
  - Transfer-ability of code across hardware generations (was) harder (capabilities change faster, need new code to use)
  - Compete for hardware against bitcoin miners, AI LLMs

Correlators on GPUs

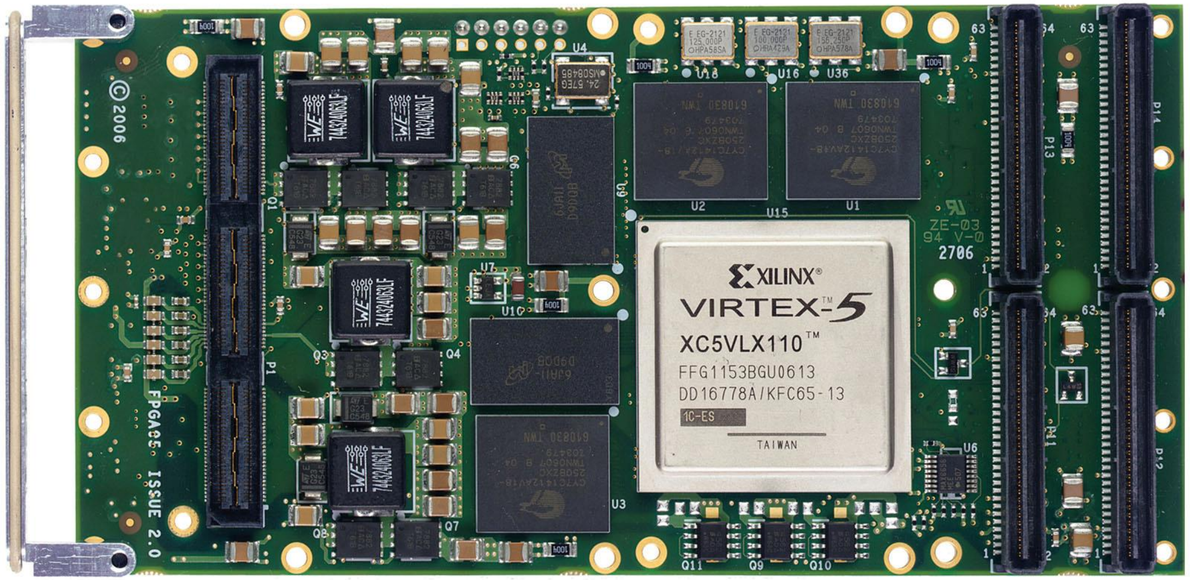


The Low Frequency Array (LOFAR), 76 stations

GMRT, India, 30 stations

Now underway: adding GPU acceleration to "general purpose" software correlators

Correlators on FPGAs



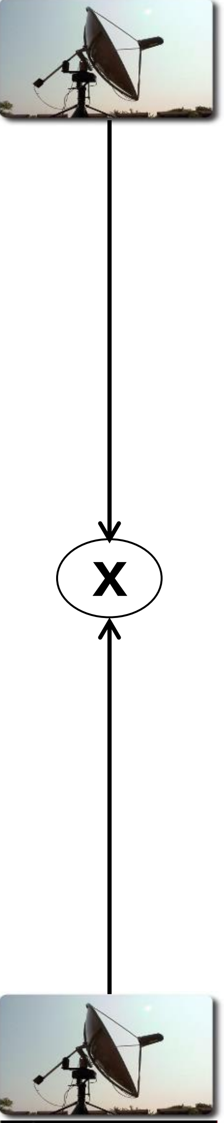
- Advantages:
  - More efficient than CPUs or GPUs, particularly for integer multiplication – big power savings
- Disadvantages:
  - Programming is harder again (especially debugging), yet fewer experts
  - Transfer-ability across hardware generations even more limited
  - Synchronous (clocked) system, less robust to perturbations c.f. CPUs/GPUs

Correlators on FPGAs

“Roach” reconfigurable FPGA board used for correlation

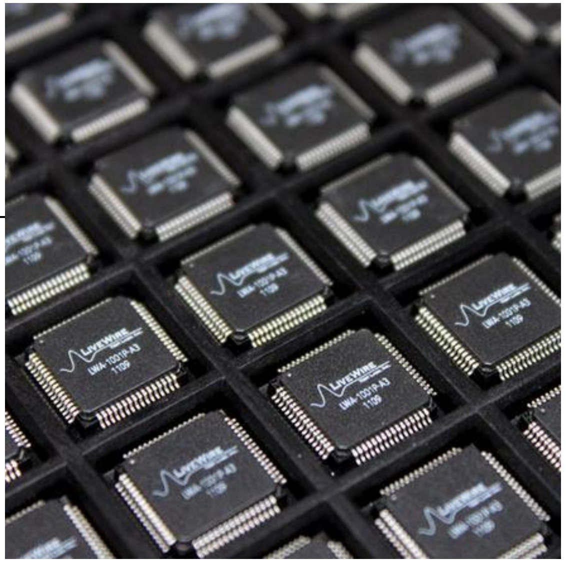
MeerKAT, 64 dishes

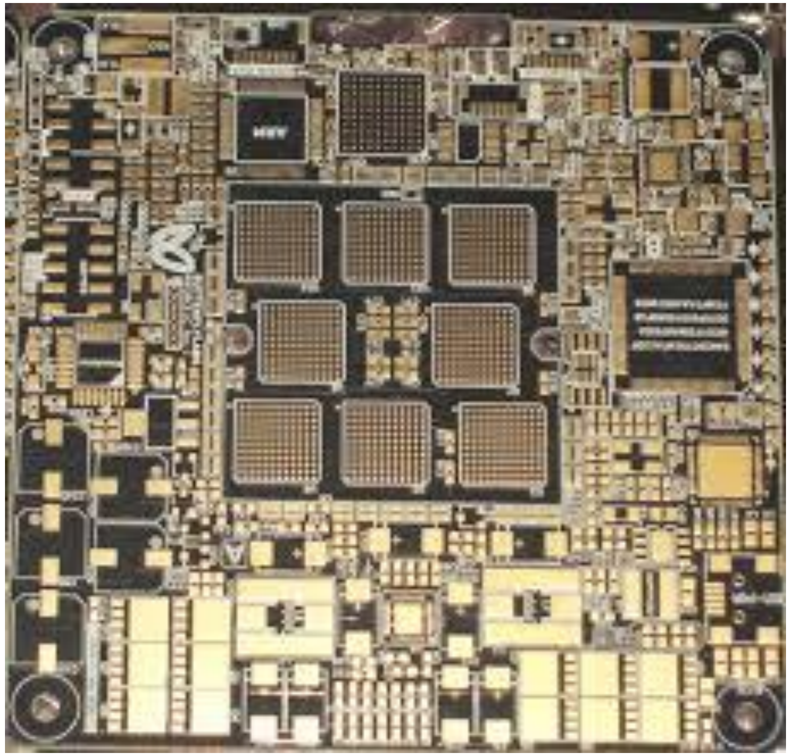
SWINBURNE UNIVERSITY OF TECHNOLOGY



Correlators on ASICs

- Advantages:
  - Highest possible efficiency, low per-unit cost
- Disadvantages:
  - Highest development cost (time and manufacturing setup)
  - Specialized knowledge required
  - Can't be changed / very difficult to upgrade during lifetime





As with FPGAs, ASICs are mounted on boards

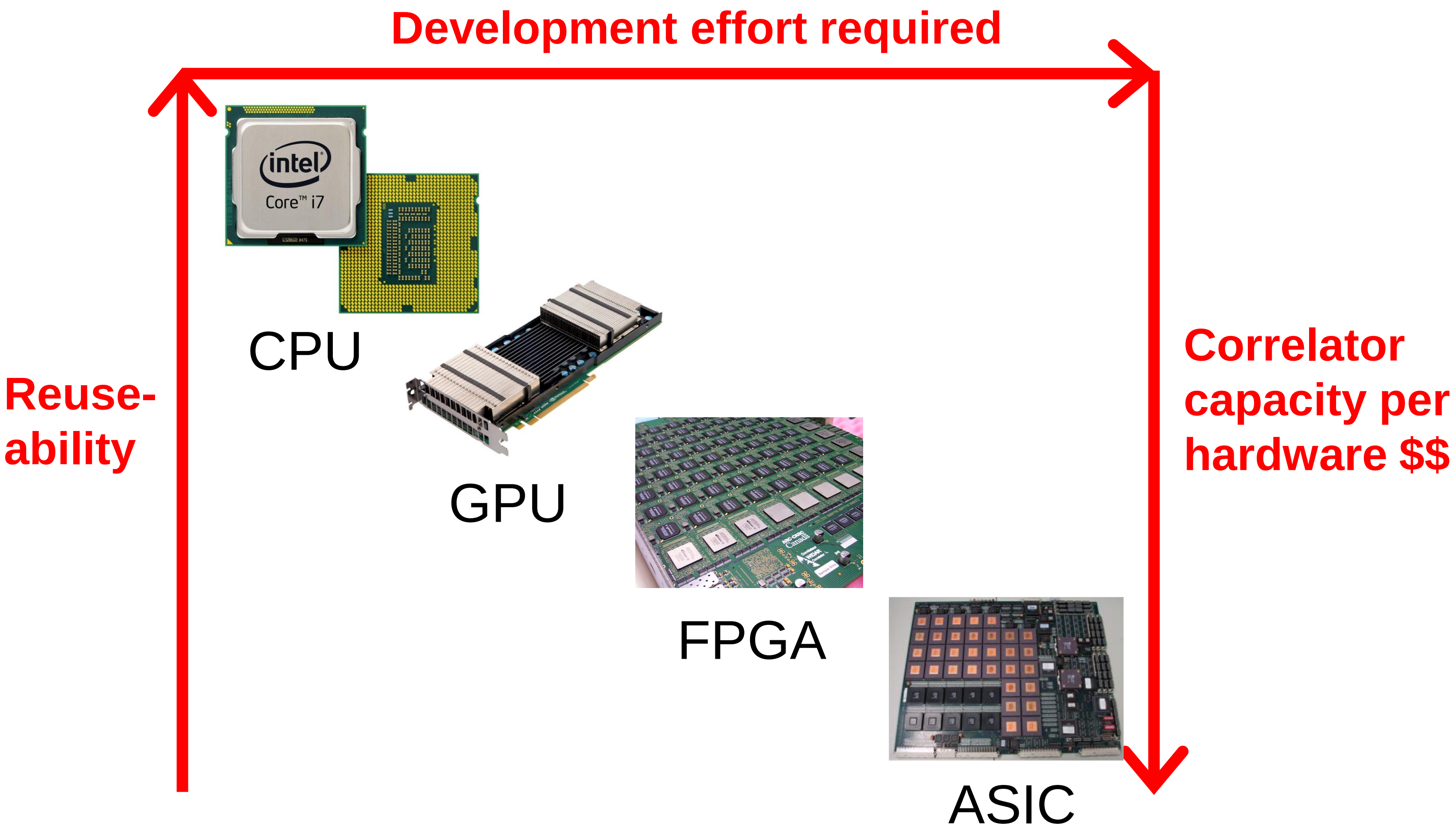
SWINBURNE UNIVERSITY OF TECHNOLOGY

Correlators on ASICs

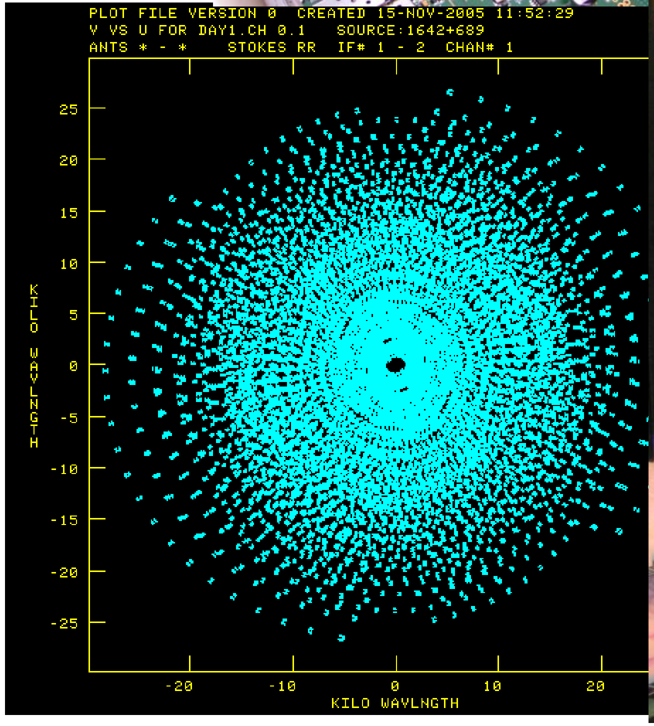
The Atacama Large Millimetre Array, Chile

The Very Large Array, New Mexico

Correlator platform overview



The end


PLOT FILE VERSION 0 CREATED 15-NOV-2005 11:52:29
V VS U FOR DAY1.CH 0.1   SOURCE:1642+689
ANTS * - *   STOKES RR   IF# 1 - 2   CHAN# 1
KILO WAVLNGTH
KILO WAVLNGTH
25
20
15
10
5
0
-5
-10
-15
-20
-25
-20
-10
0
10
20