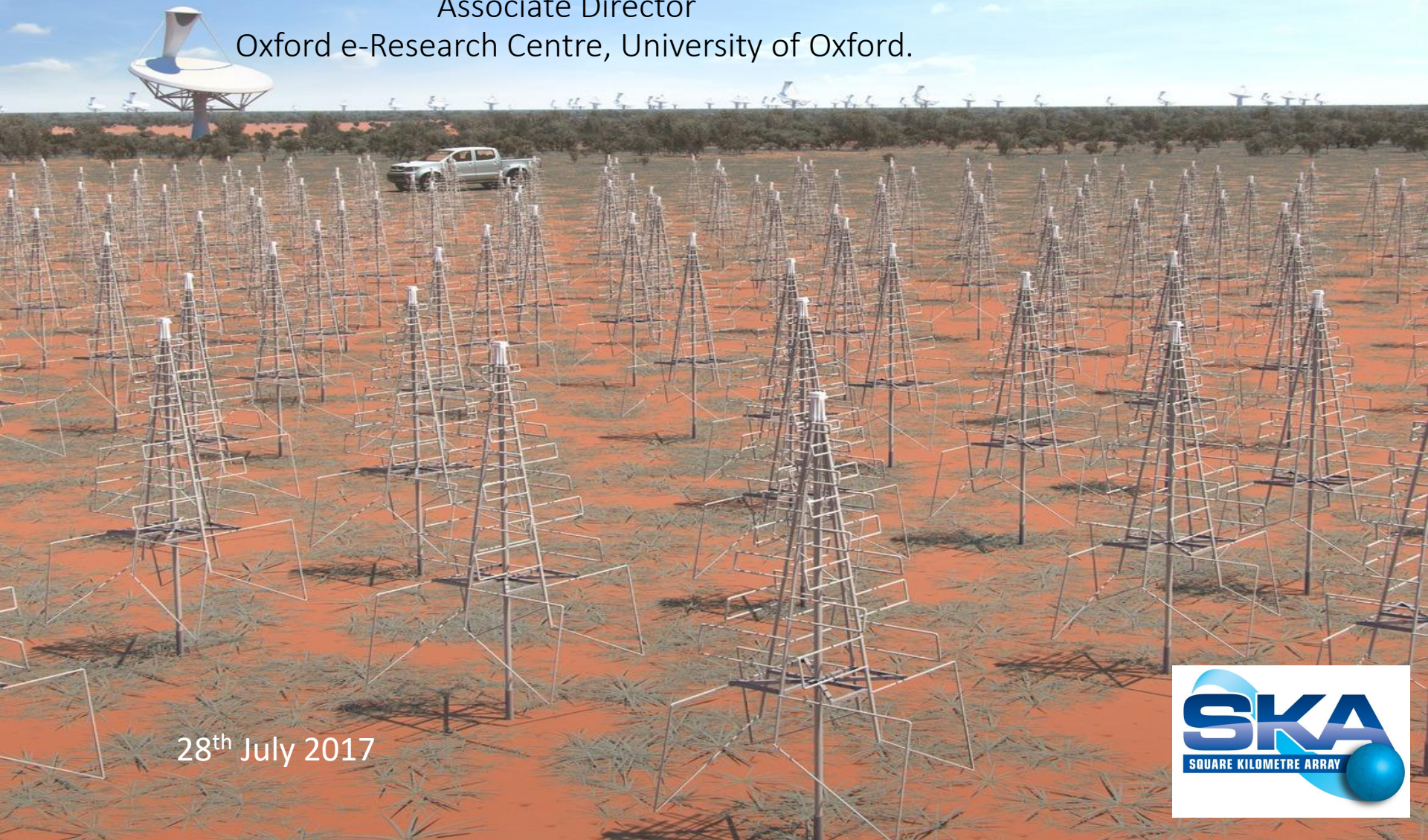
Signal processing at scale - GPUs enabling next generation radio telescopes

W. Armour
Associate Director
Oxford e-Research Centre, University of Oxford.



28th July 2017

Cutting Edge Radio Telescopes

UNIVERSITY OF OXFORD

Oxford e-Research Centre

MeetKAT -Sixty four, 13.5m dishes. A pathfinder for the Square Kilometre Array

Murchison Widefield Array (MWA) -Fixed 128 array of 16-element dual-polarisation antennas

Australian Square Kilometre Array Pathfinder (ASKAP) - 36 antennas each 12 meters in diameter

LOFAR (LOw Frequency ARray) - Low frequency array of dipole antennas

Canadian Hydrogen Intensity Mapping Experiment (CHIME) - five 100 x 20 meter cylinders

Five hundred meter Aperture Spherical Telescope (FAST) - 500m radio telescope

SKA

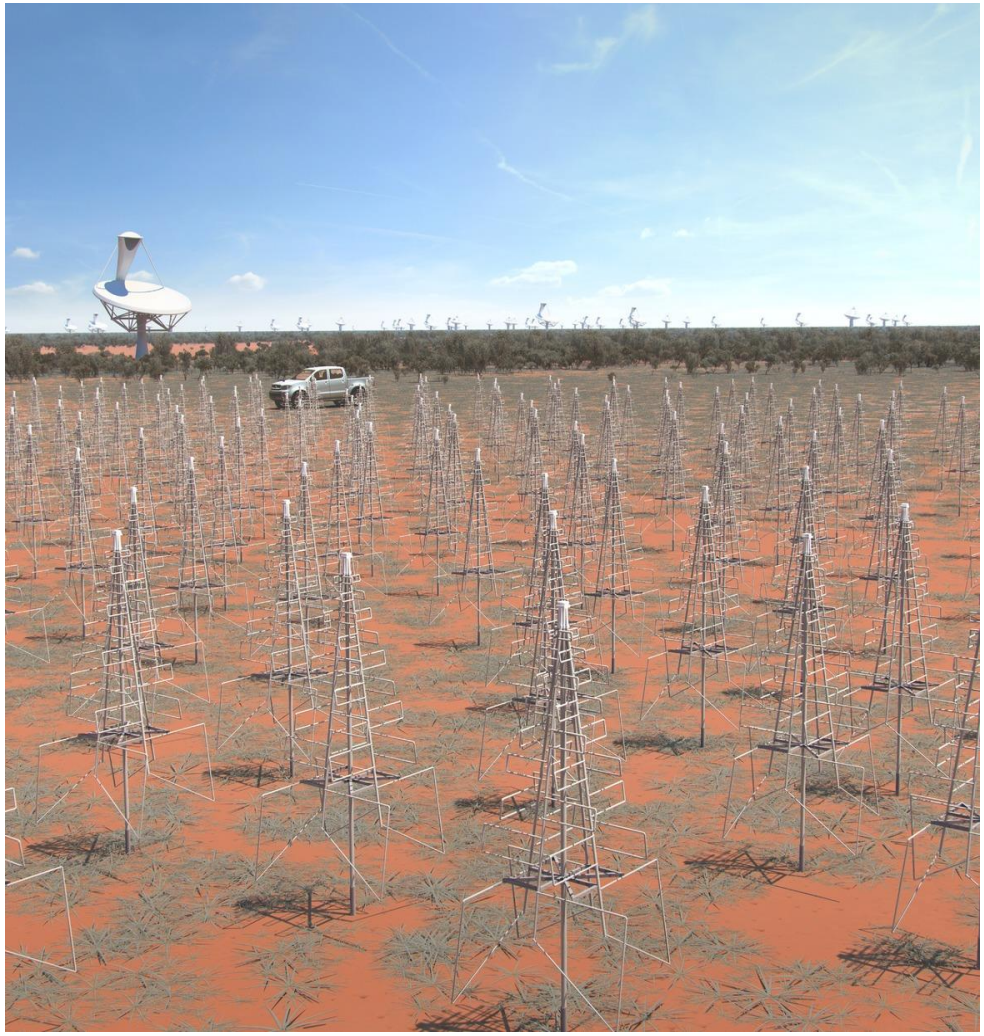
SQUARE KILOMETRE ARRAY

UNIVERSITY OF OXFORD

Oxford e-Research Centre

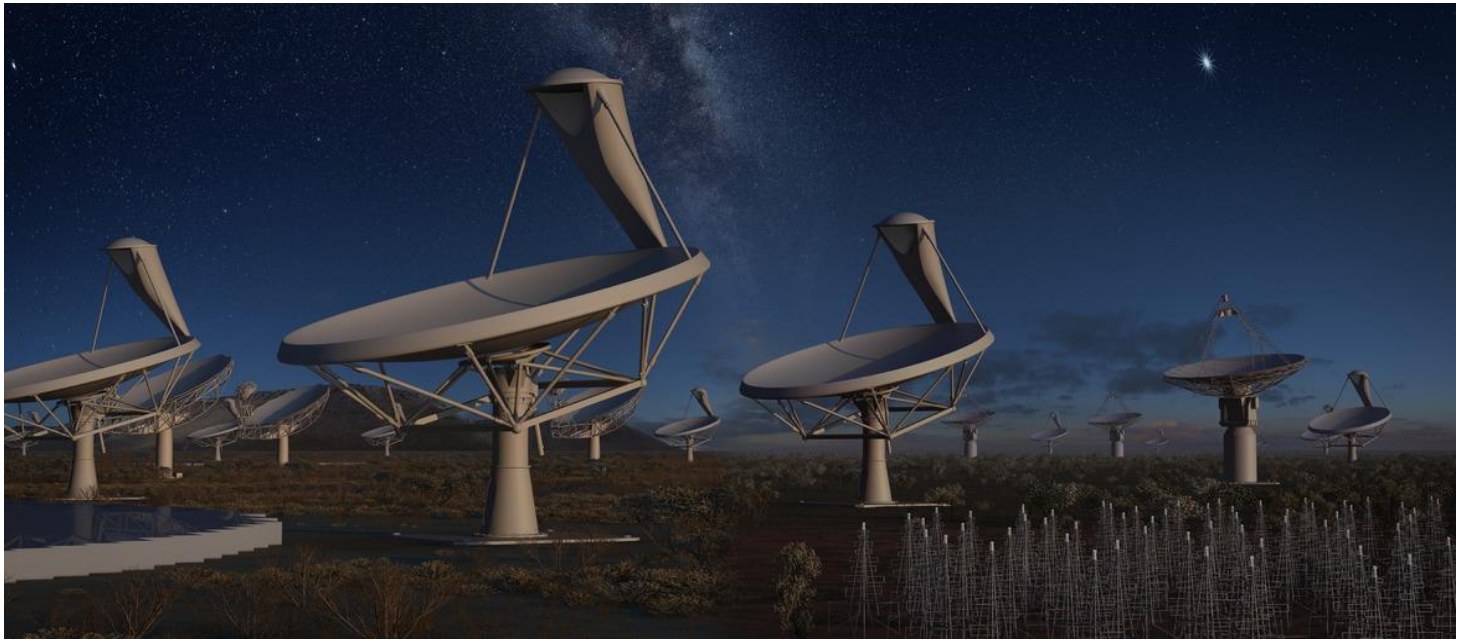
What is SKA?

What is SKA?



- SKA?
  - Square Kilometre Array
- What?
  - SKA is a radio telescope
- Where?
  - SKA will be built in South Africa and Australia

What is SKA?

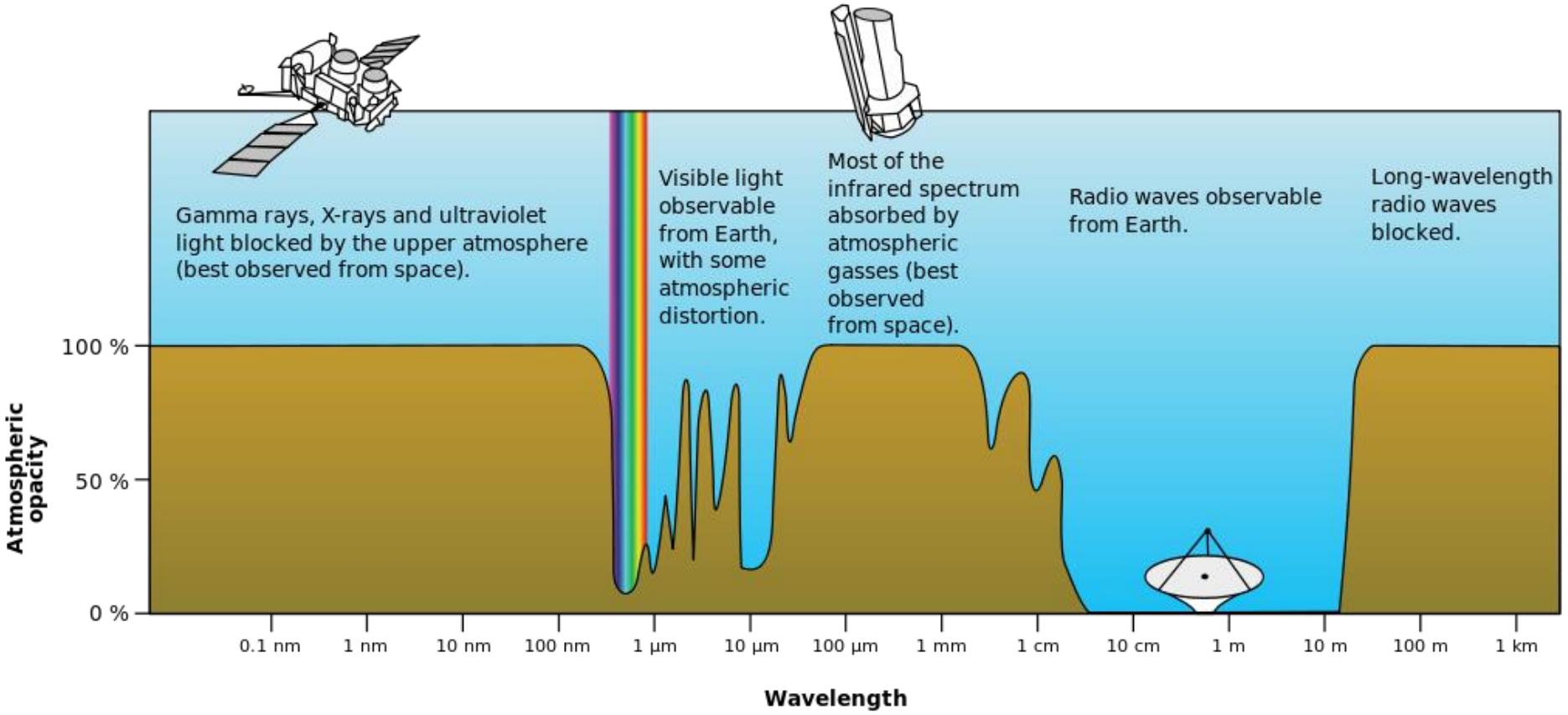


Three types of telescope:

- Dishes
- Mid frequency aperture arrays
- Low frequency aperture arrays

What is SKA?

Wide frequency range from 50MHz to 20GHz – wavelengths 15 mm to 6 m.



Great for lots of different science!

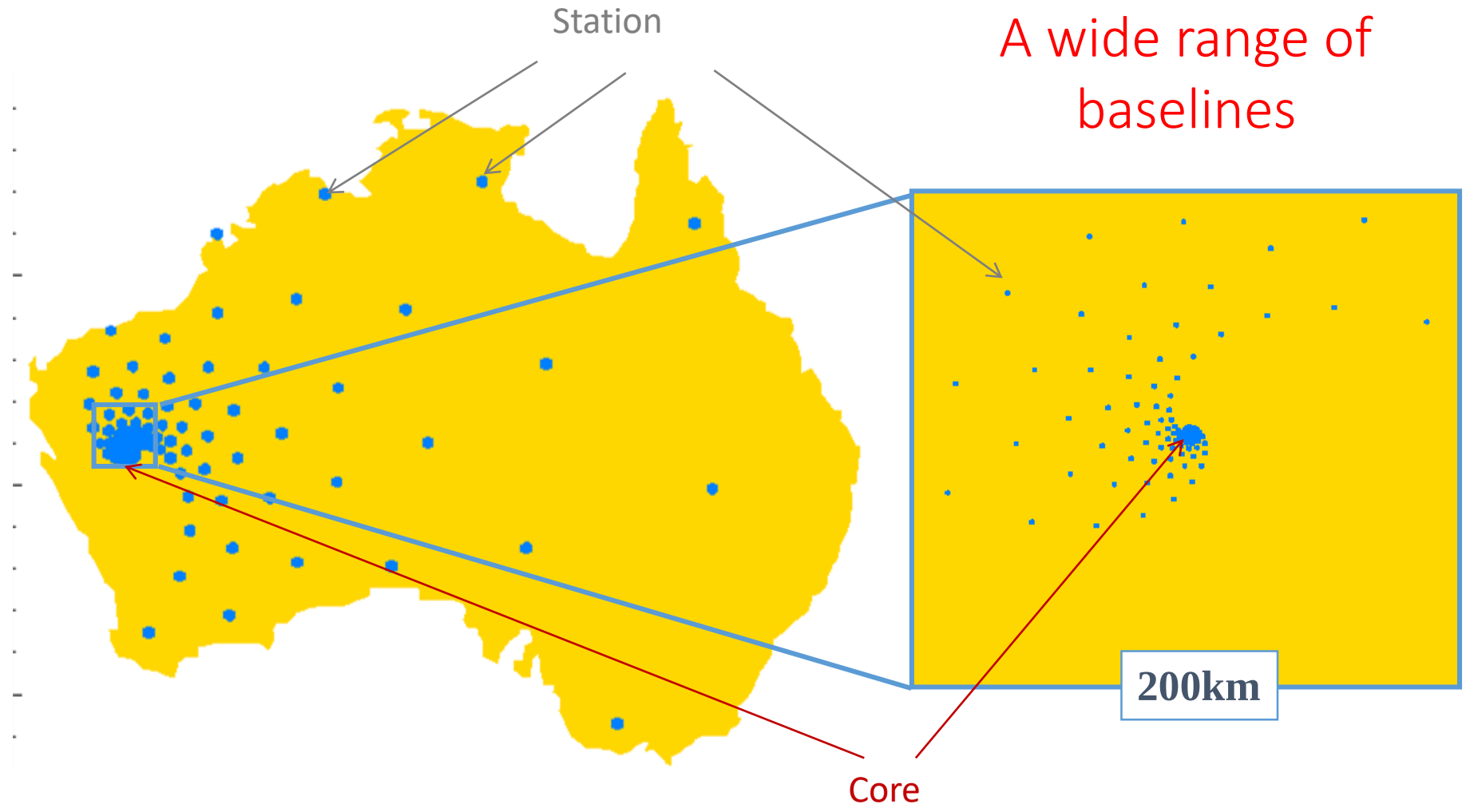
Image source Wikipedia. Authors: NASA (original); SVG by Mysid

What is SKA?

Phase delays are applied to signals from individual antennas

Allowing many different observing beams to be placed on the sky at the same time

An example of a proposed SKA configuration



Slide courtesy of Anne Trefethen

UNIVERSITY OF OXFORD

Oxford e-Research Centre

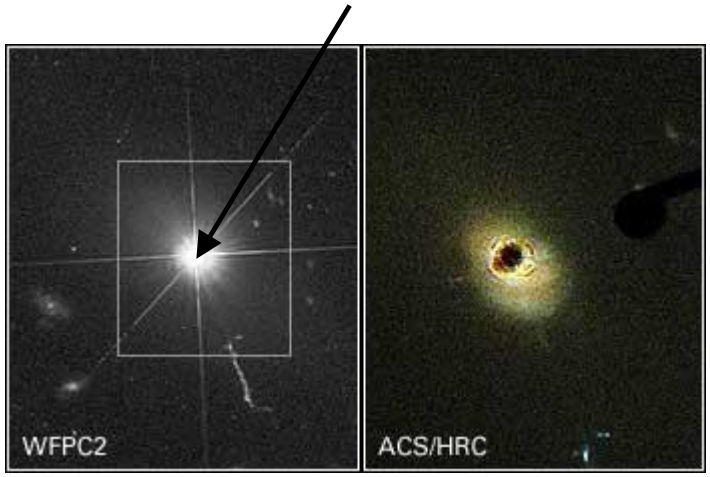
SKA Science?

SKA science

- How do galaxies evolve
  - What is dark energy?
- **Tests of General Relativity**
  - **Was Einstein correct?**
- Probing the cosmic dawn
  - How did stars form?
- The cradle of life
  - Are we alone?

SKA time-domain science

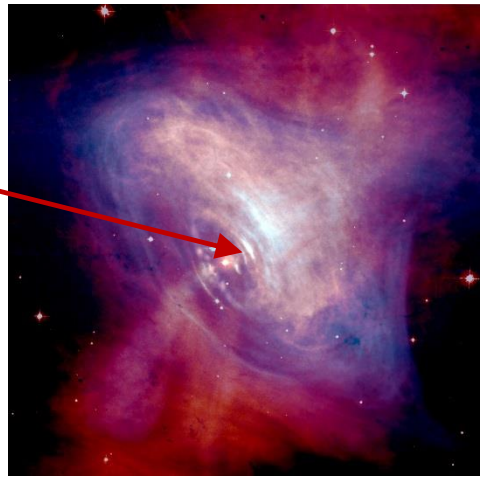
Quasars – Energetic region of a distant galactic core, surrounding a supermassive black hole



NASA and J. Bahcall (IAS)

RRATS – Rotating Radio Transients. Short, bright irregular radio pulses. Discovered 2006

Pulsars – Magnetized, rotating neutron stars. Emit synchrotron radiation from the poles, e.g. Crab Nebula

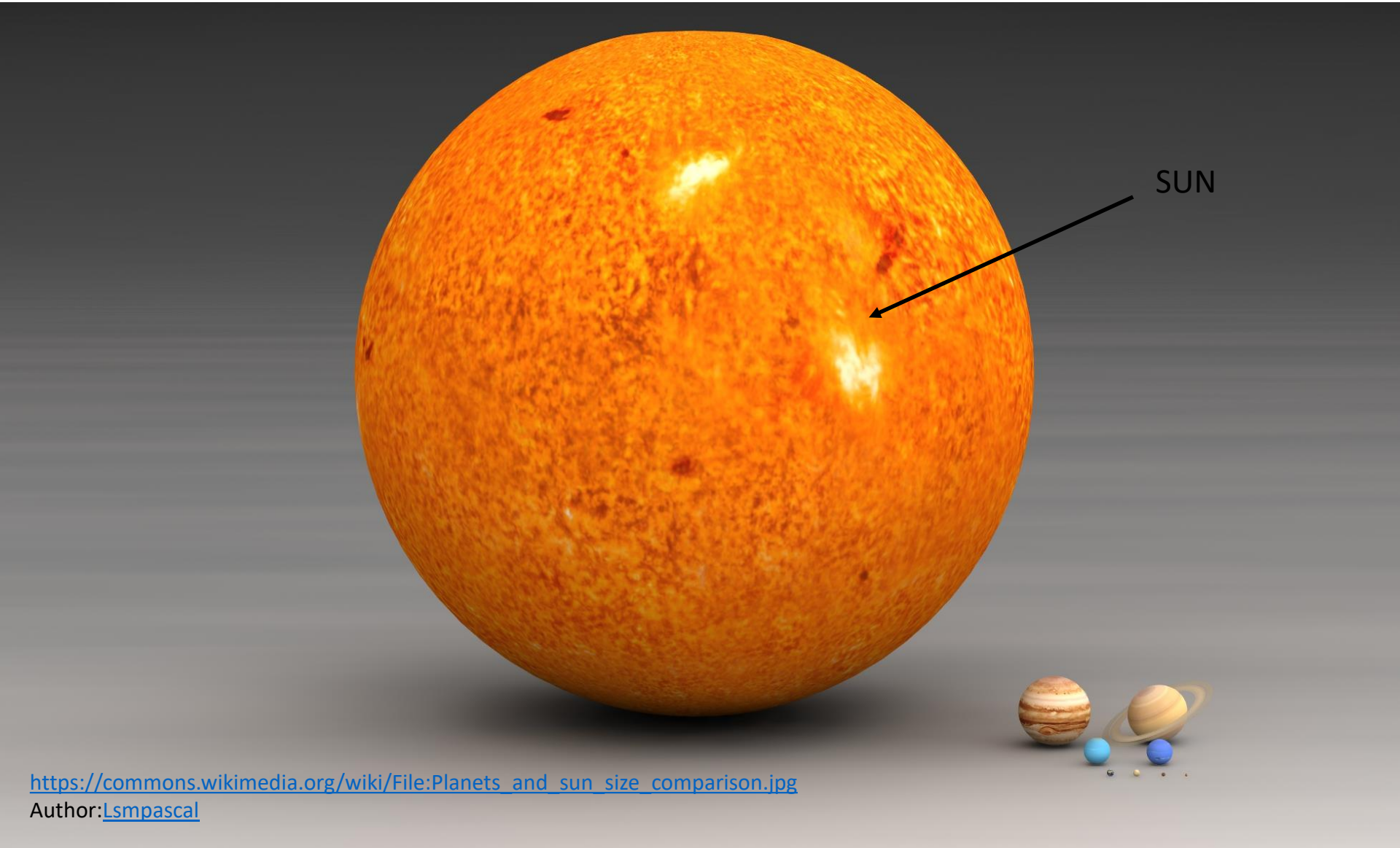


Hester et al.

FRB → ?

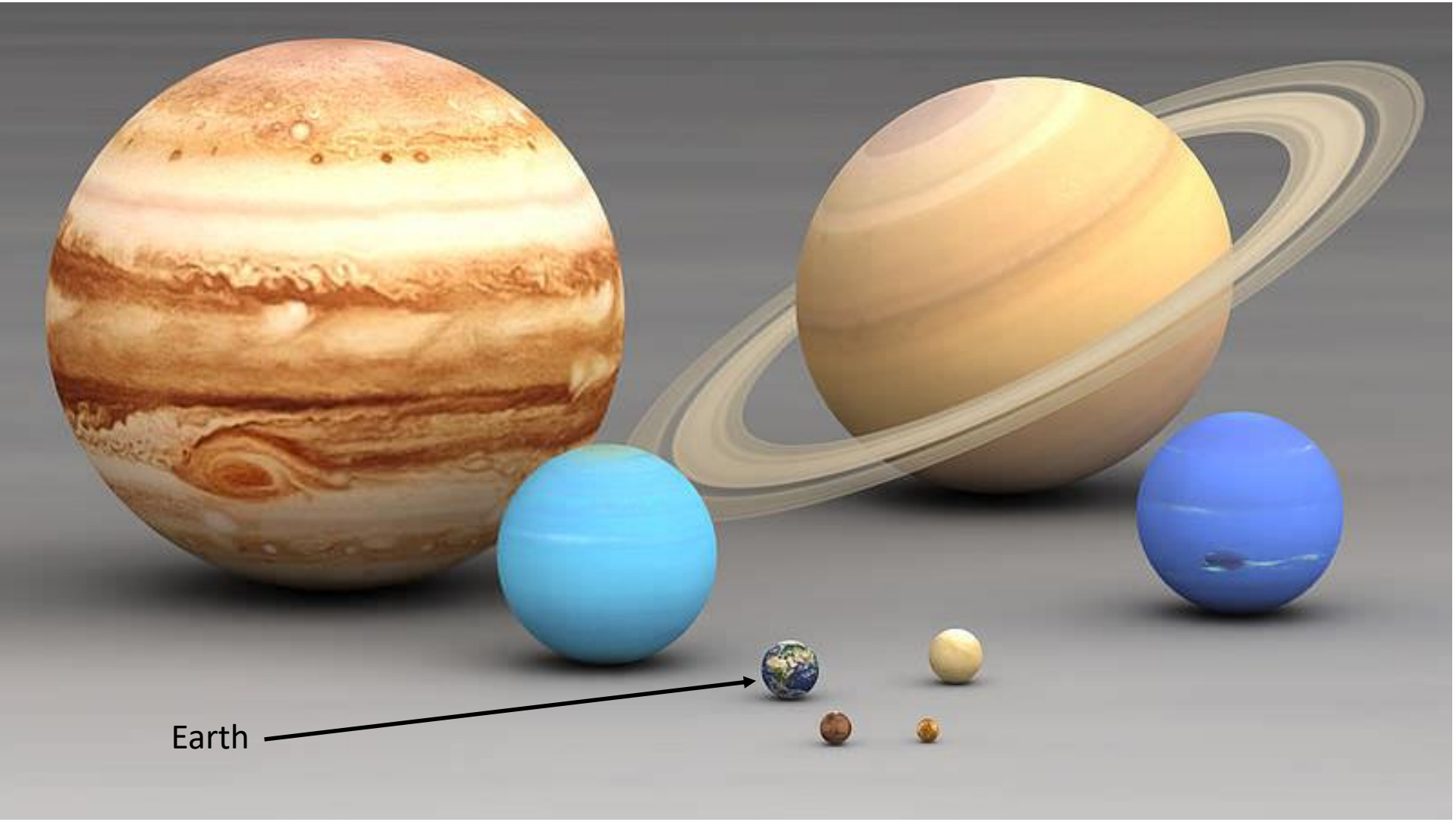
Of order of 10 found in survey data. Very high DM implies extra-galactic(?) Unknown origin.

Size and Scale



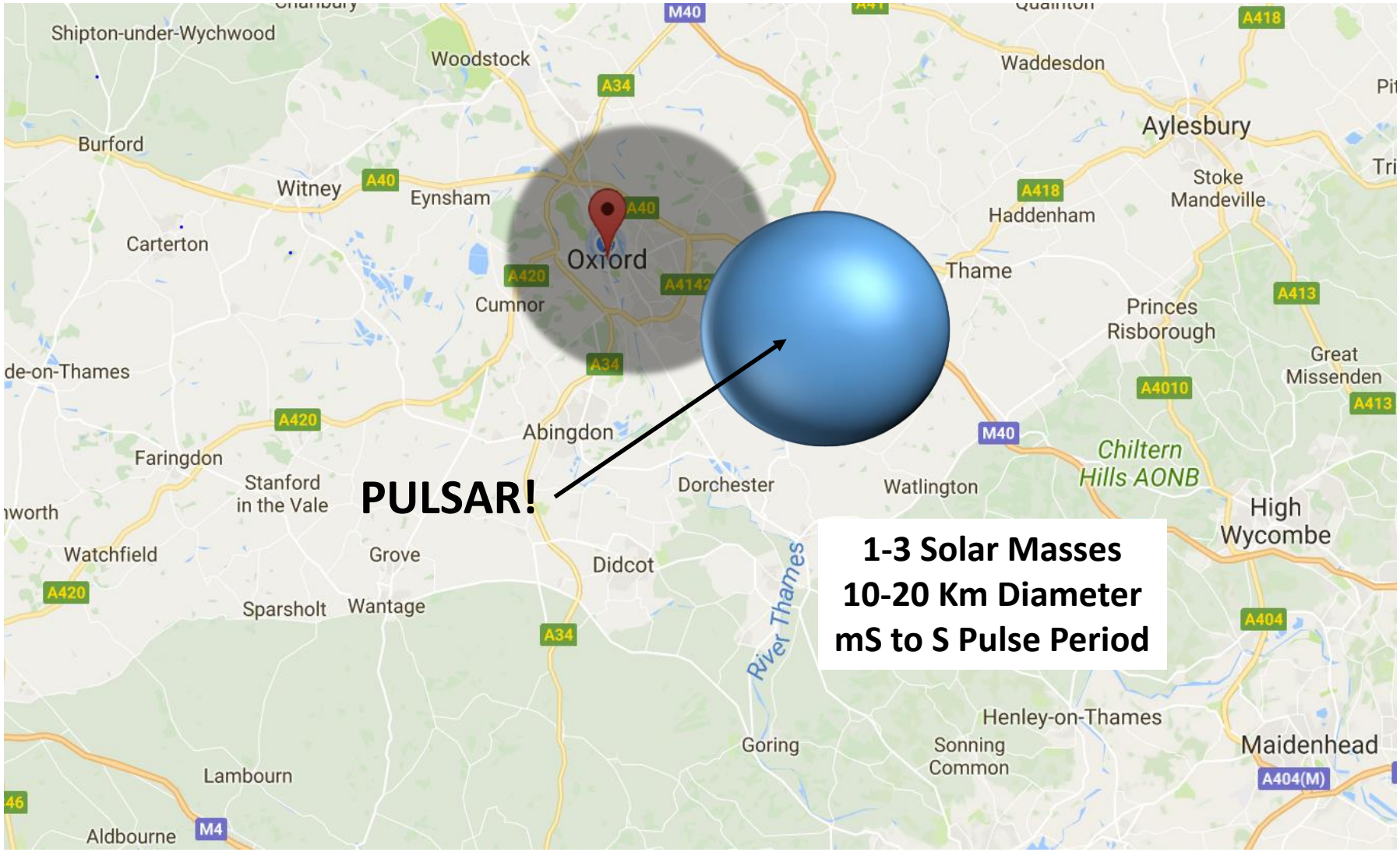
https://commons.wikimedia.org/wiki/File:Planets_and_sun_size_comparison.jpg
Author:Lsmpascal

Size and Scale



https://commons.wikimedia.org/wiki/File:Planets_and_sun_size_comparison.jpg
Author:Lsmpascal

Pulsars


Shipton-under-Wychwood
Woodstock
Waddesdon
Burford
Aylesbury
Witney
Eynsham
Stoke Mandeville
Haddenham
Carterton
Oxford
Thame
Cumnor
Princes Risborough
Great Missenden
Abingdon
Faringdon
Stanford in the Vale
PULSAR!
Dorchester
Watlington
Chiltern Hills AONB
High Wycombe
Watchfield
Grove
Didcot
Sparsholt
Wantage
River Thames
Henley-on-Thames
Goring
Sonning Common
Maidenhead
Lambourn
Aldbourne
M40
A34
A40
A420
A4142
A418
A413
A4010
A404
A404(M)
M4
1-3 Solar Masses
10-20 Km Diameter
mS to S Pulse Period

Pulsars

Magnetic field is offset from rotational axis

- Act as cosmic lighthouses
- Extremely periodic
- Make great clocks!

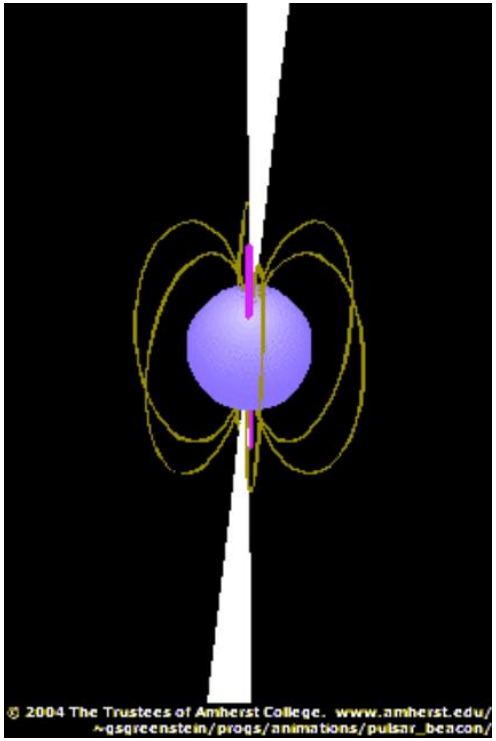
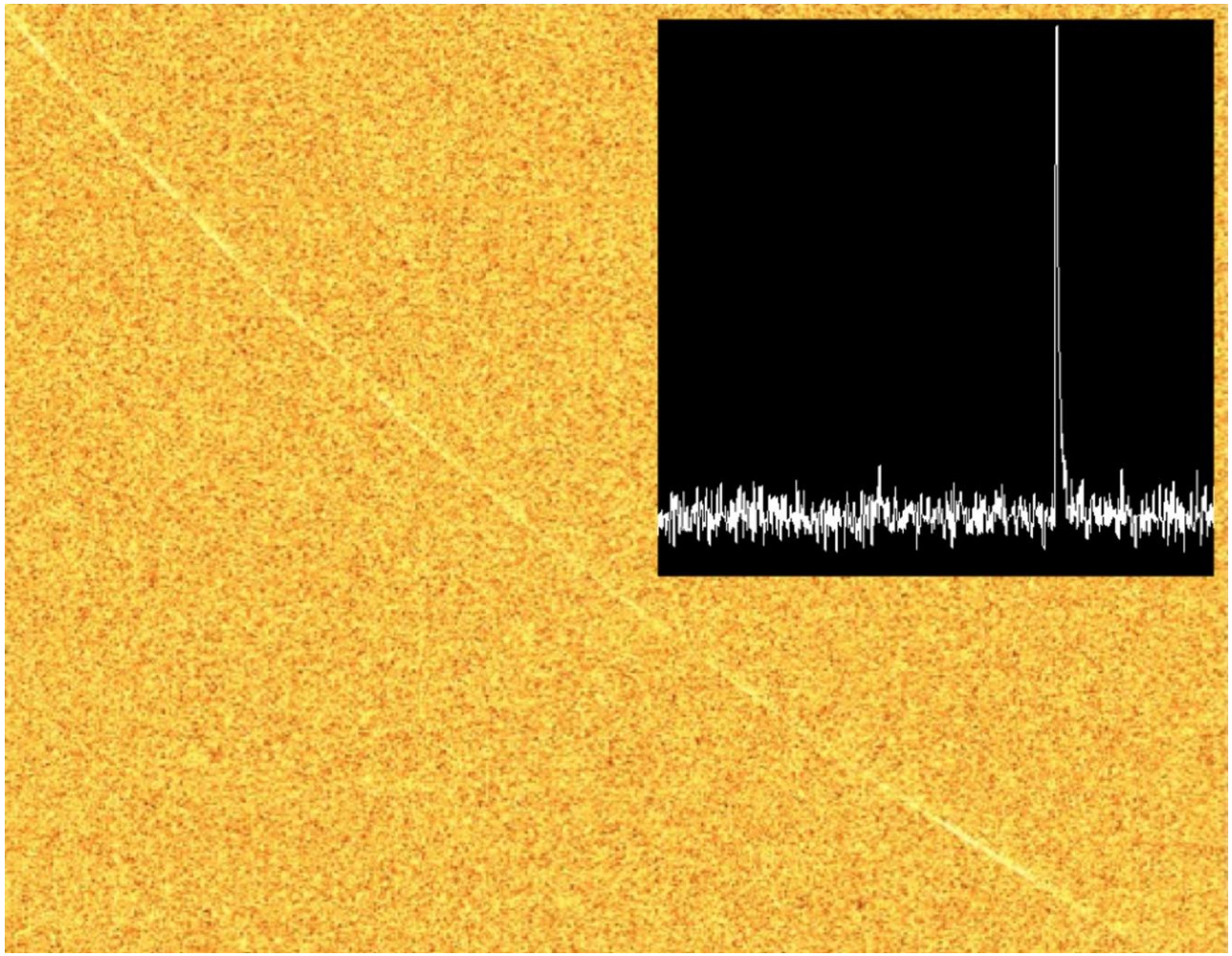


Image: Amherst College

Fast Radio Bursts



Extremely Bright

Extremely Dispersed

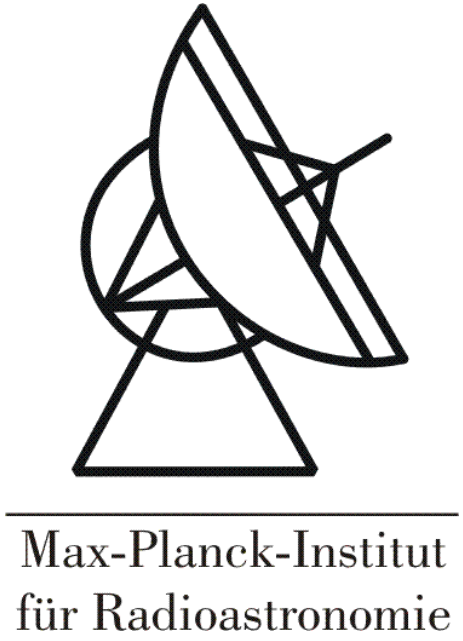
=> Extra Galactic ?

?

Credit: FRB110220 Dan Thornton (Manchester)

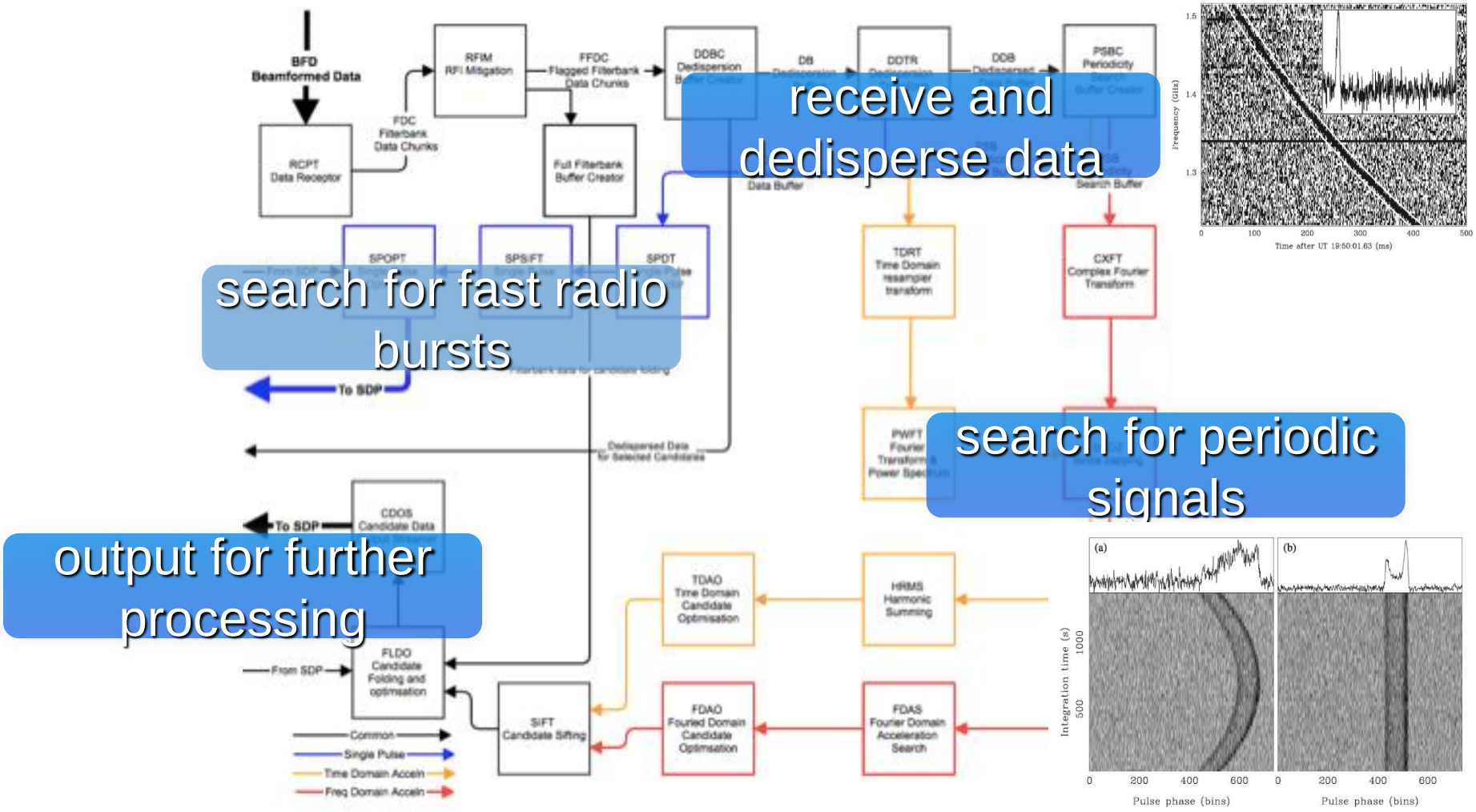
SKA Signal Processing

SKA time-domain science - TDT



International team led by Oxford and Manchester

Time domain signal processing



Slide courtesy of Aris Karastergiou

SKA time-domain data rates

- 2000 beams on the sky
- 20,000 samples per second
- 4096 frequency channels per sample
- 4x8 bits per sample

160GB/s of relevant data to analyse -
Approximately equal to analysing 50 hrs of HDTV every second.

Most Costly computational operations in data processing pipeline…

De-dispersion ~ $O(n_{dms} * n_{beams} * n_{samps} * n_{chans})$
Acceleration search ~ $O(n_{dms} * n_{beams} * n_{samps} * n_{acc} * \log(n_{samps}) * 1/t_{obs})$

SKA Compute Requirements

Acceleration search

~ $O(n_{dms} * n_{beams} * n_{samps} * n_{acc} * \log(n_{samps}) * 1/t_{obs})$

= (6120 x 2000 x 8,388,608 x 96 x 23) / 534

=424,550,278,500,674

=> ½ PetaFlop

http://www.olcf.ornl.gov/titan/

De-dispersion

~ $O(n_{dms} * n_{beams} * n_{samps} * n_{chans})$

= 6120 x 2000 x 20,000 x 4096

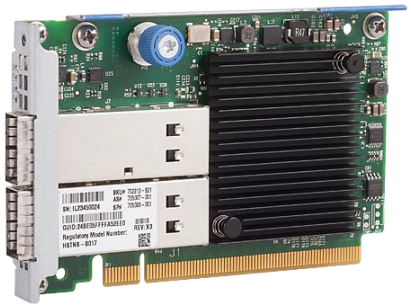
= 1,002,700,800,000,000

=> PetaFlop

SKA Time Domain Challenges

Data is too vast to store on site

- Should we transport it of site?



Data is too vast to transport

- Processing must happen close to the telescope

We don't want to loose data

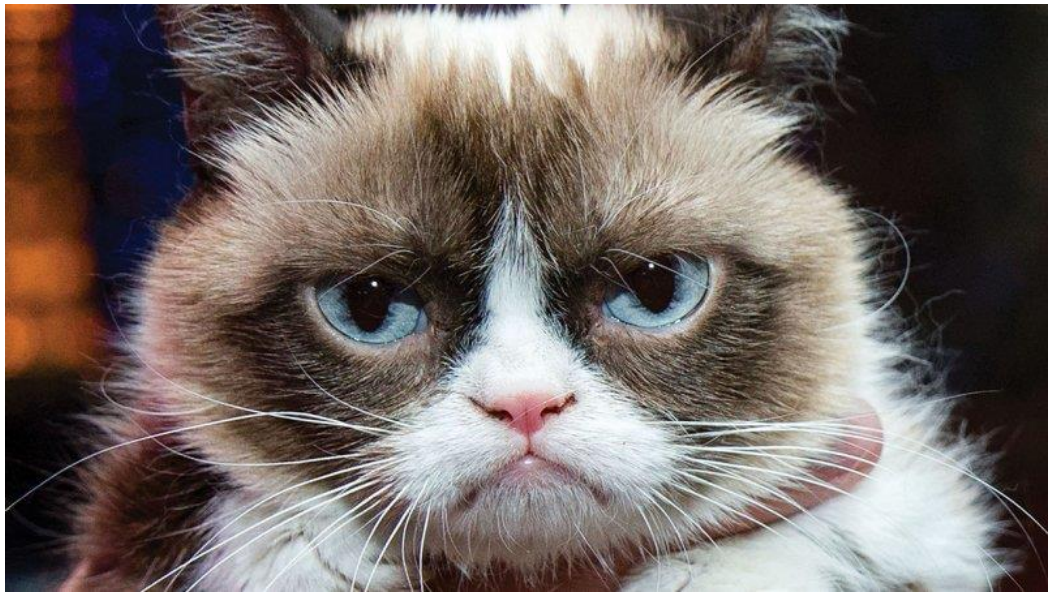
- Processing must happen in real-time

How do we put a computer capable of processing big-data streams in real-time in the desert

Connectivity, Power, Operation???

Real-time considerations for the SKA

We need advances in **both** computational and mathematical algorithms to achieve our goal of real-time processing.



“There’s more than one way to skin a cat…”

Advances in technology and Algorithms are needed

UNIVERSITY OF OXFORD

Oxford e-Research Centre

Technological

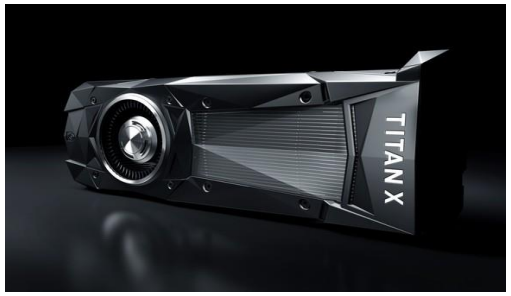
GPU programming using CUDA.

Low level CPU/Phi with Vector Intrinsics.

Multi/Many-core using OpenMP.

OpenCL for FPGAs.

MPI.



Numerical analysis.

Developing parallel algorithms.

Mathematics and Statistics expertise.

Modelling and Simulation.

$$A_{r_0} \simeq \sum_{k=[r_0]-m/2}^{[r_0]+m/2} A_k A^*_{r_0-k}$$

Foundational

UNIVERSITY OF OXFORD

Oxford e-Research Centre

Technological

Work with leading HPC Companies…

Intel Xeon Phi

GPUs – NVIDIA P100 Pascal GPUs

FPGAs – Stratix 5 and 10

ALTERA

Our first prototype hardware for SKA

Hybrid GPU/FPGA compute node – Lenovo x3650 Server

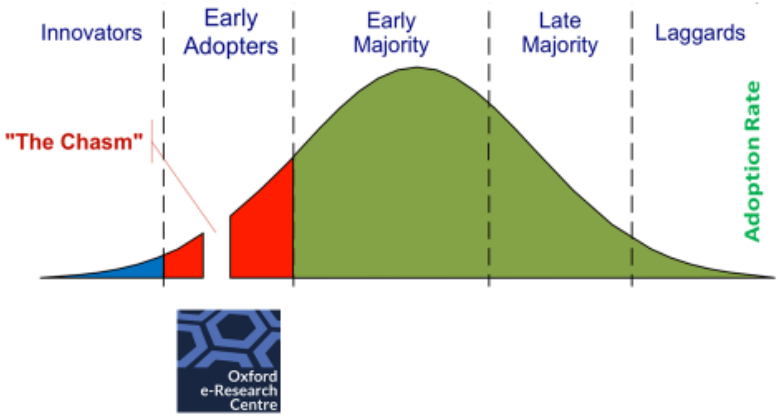
De-risked by using COTs technology

Uses low power CPUs, 1.2V DDR4 RAM and SSDs to try to reduce power consumption

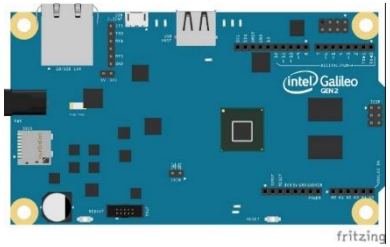
512 GB DDR4 RAM
5x SATA SSD drives
2x Intel Xeon E5-2650L
1x NVIDIA P100 GPU
1x Arria 10 FPGA

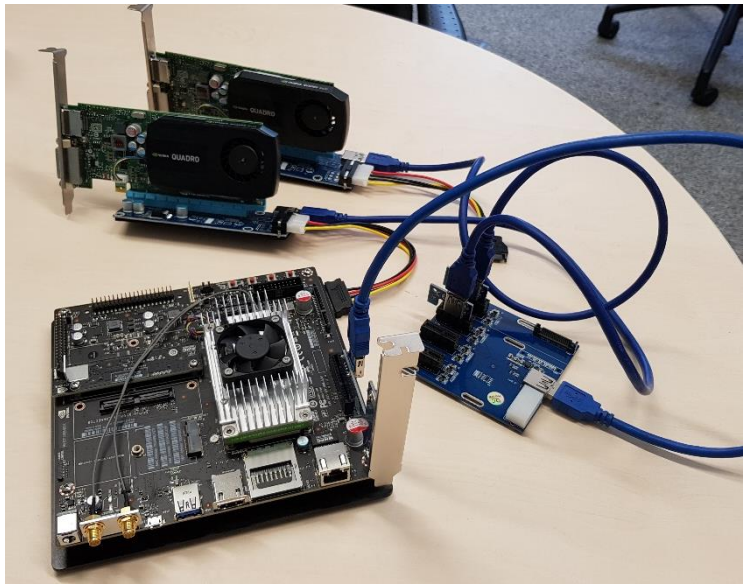
Kate Steele, Jim Roache, Noam Rosen, Guy, Luigi… (Lenovo)
Caroline Bradley, Georgina Ellis (OCF) Jeremy Purches, Kate Clark (NVIDIA)

Also consider up-and-coming Technologies…



NVIDIA - TK1 / TX1 / TX2
Intel – Edison/Galileo
Arduino enabled: **“open source”** Hardware





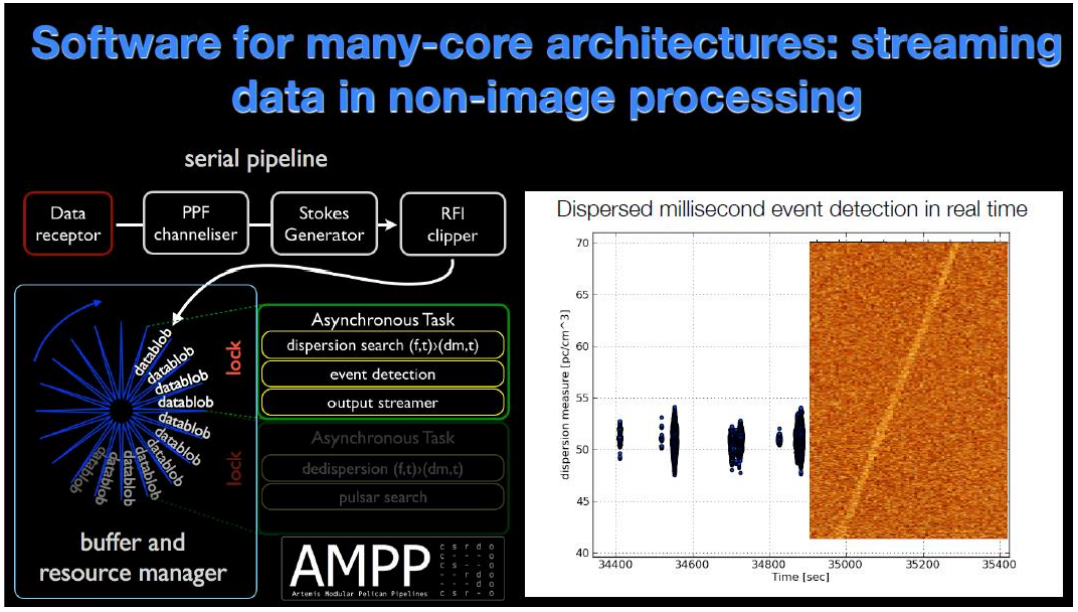
TX2 + PCIe 3.0 switch with
2x GPUs attached

Oxford Projects: ARTEMIS and Astro-Accelerate

Many-core accelerated modules to enable real-time time-domain data processing.

Support multiple architectures such as GPUs, FPGA, CPUs and Xeon Phi.

AstroAccelerate



End-to-end signal processing pipeline for FRBs.

Real-time data management and movement.

Real-time discovery of events as they happen.

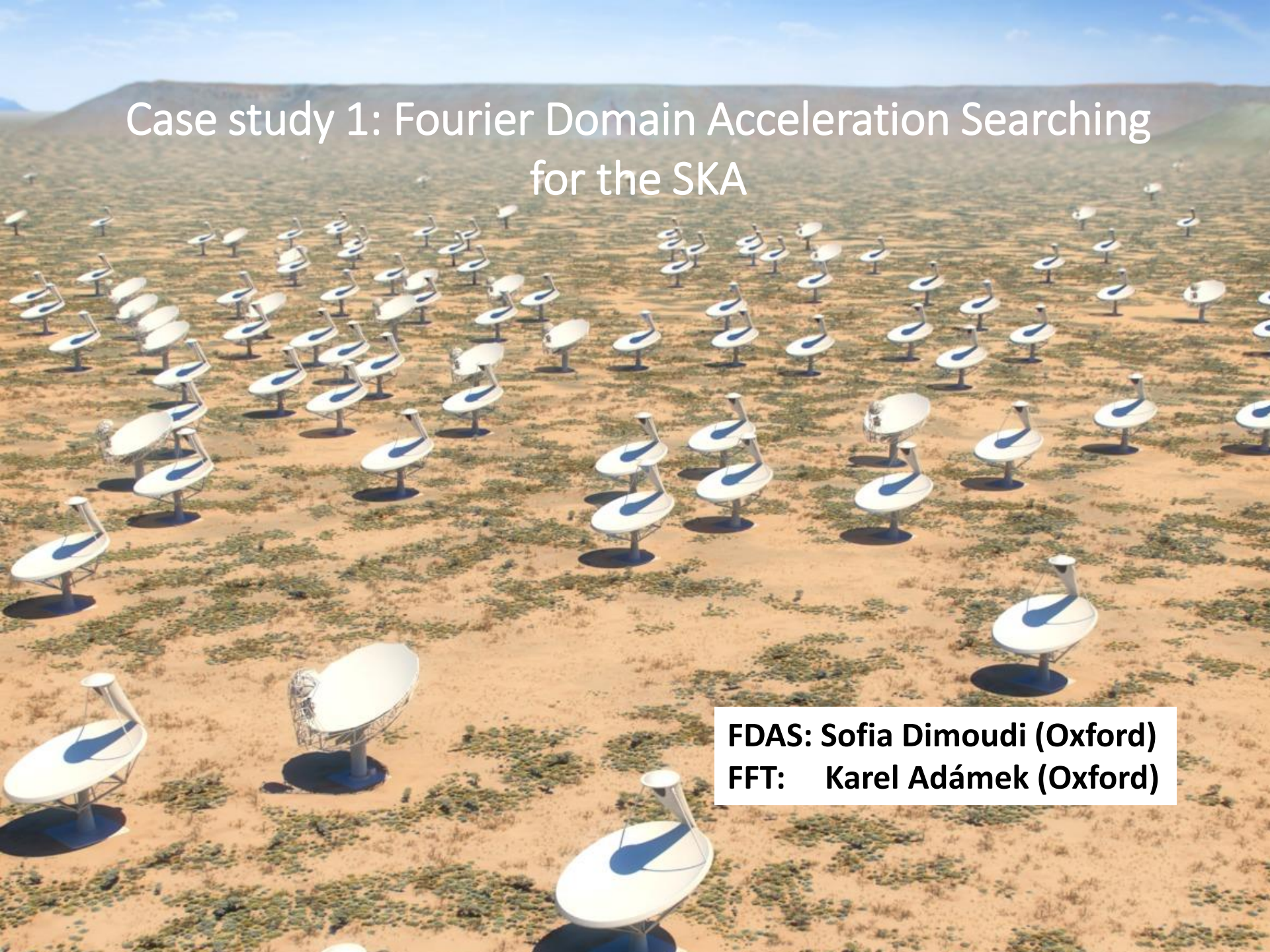
ARTEMIS: aris.karastergiou@astro.ox.ac.uk
Astro-Accelerate: wes.armour@oerc.ox.ac.uk

UNIVERSITY OF OXFORD

Oxford e-Research Centre

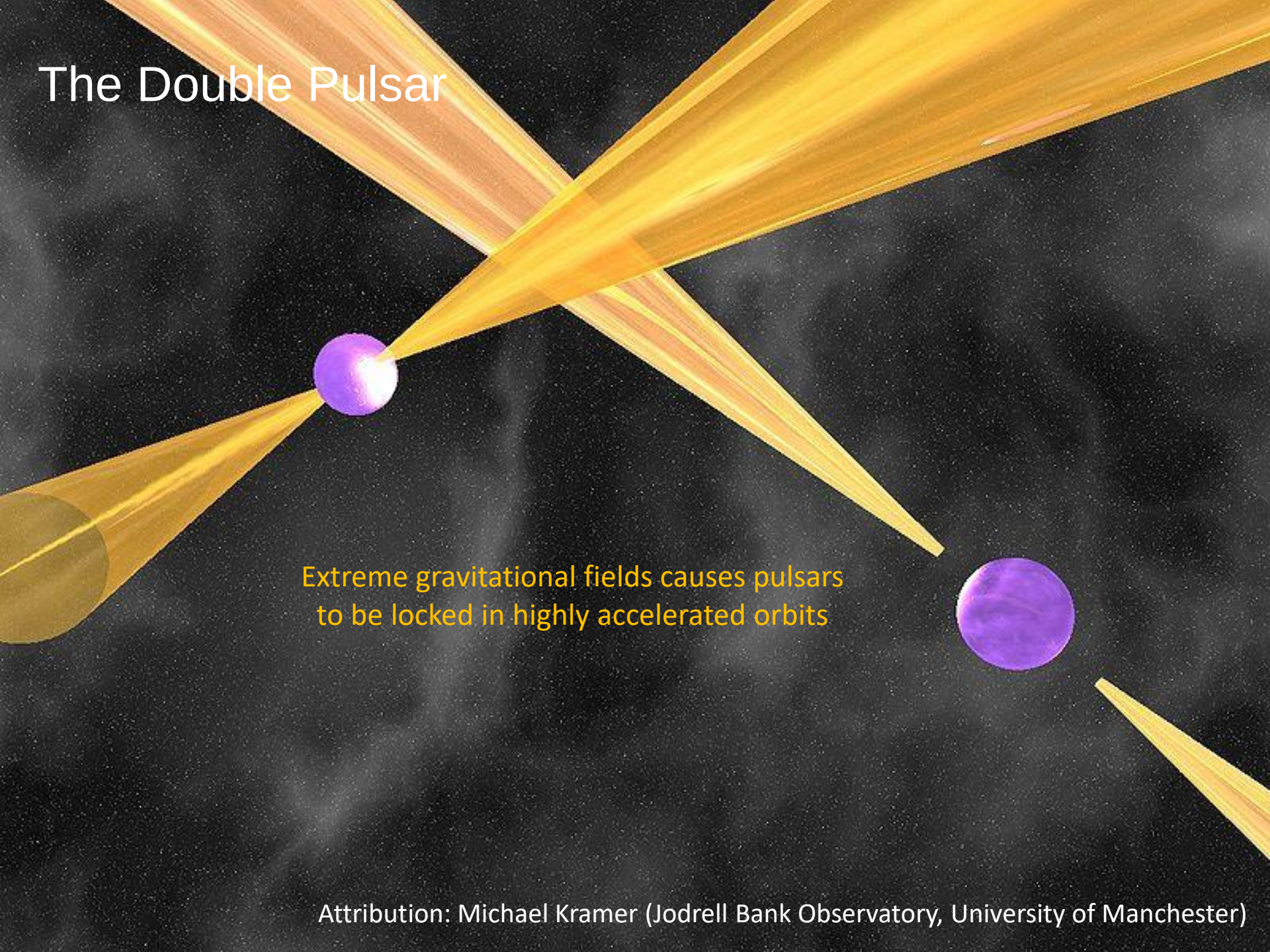
Foundational

Case study 1: Fourier Domain Acceleration Searching for the SKA



FDAS: Sofia Dimoudi (Oxford)
FFT: Karel Adámek (Oxford)

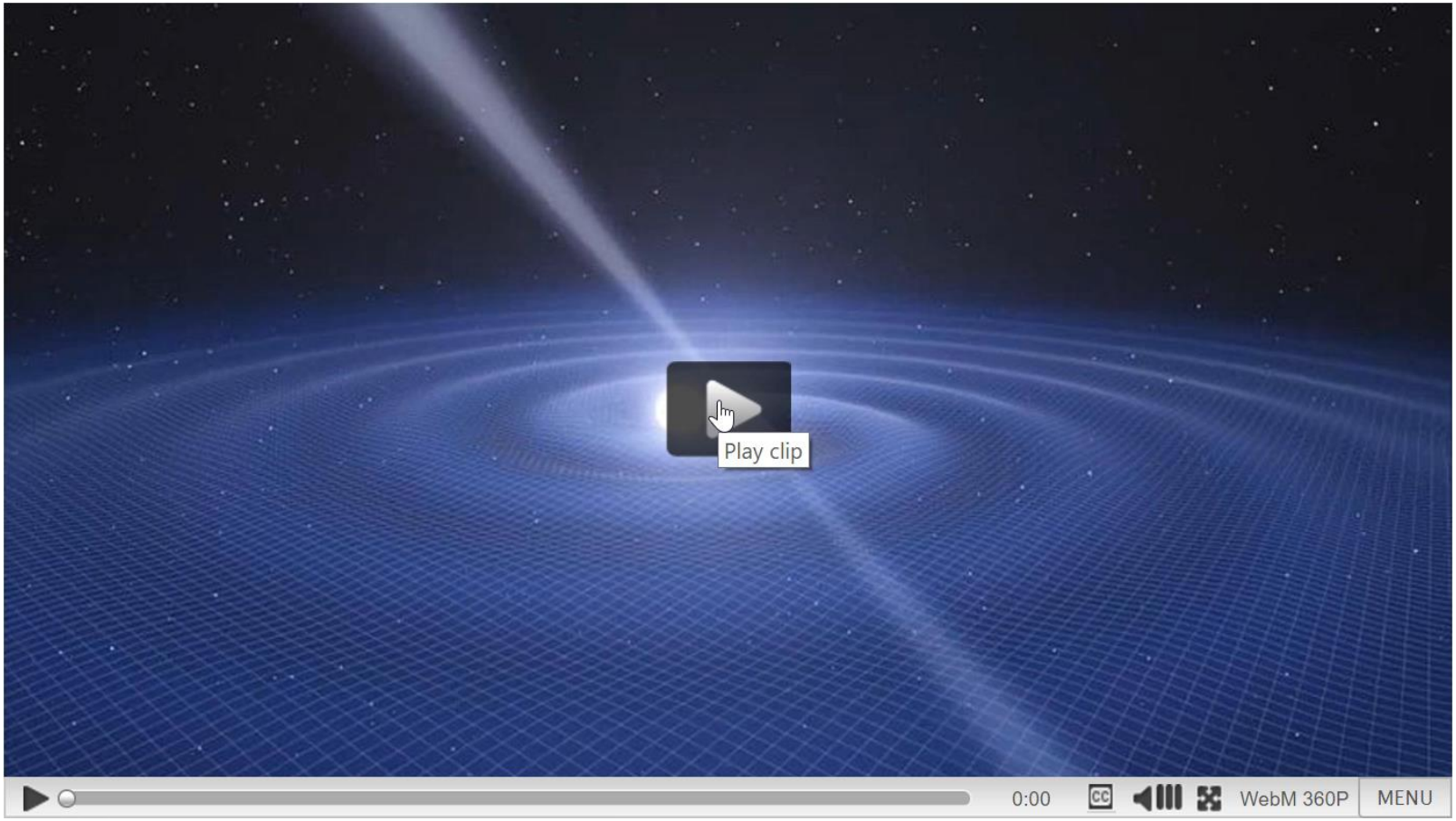
The Double Pulsar



Extreme gravitational fields causes pulsars to be locked in highly accelerated orbits

Attribution: Michael Kramer (Jodrell Bank Observatory, University of Manchester)

Gravitational Waves



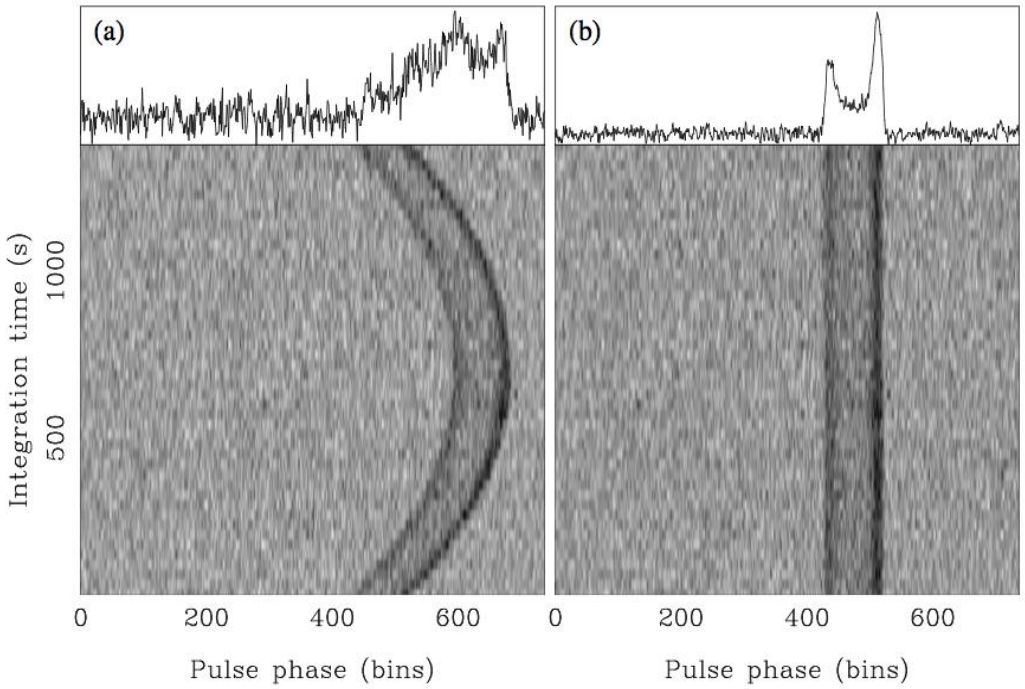
http://www.eso.org/public/videos/eso1319a/ Author: ESO/L. Calçada

Fourier Domain Acceleration Search

Signals from binary systems can undergo a Doppler shift due to accelerated motion experienced over the orbital period.

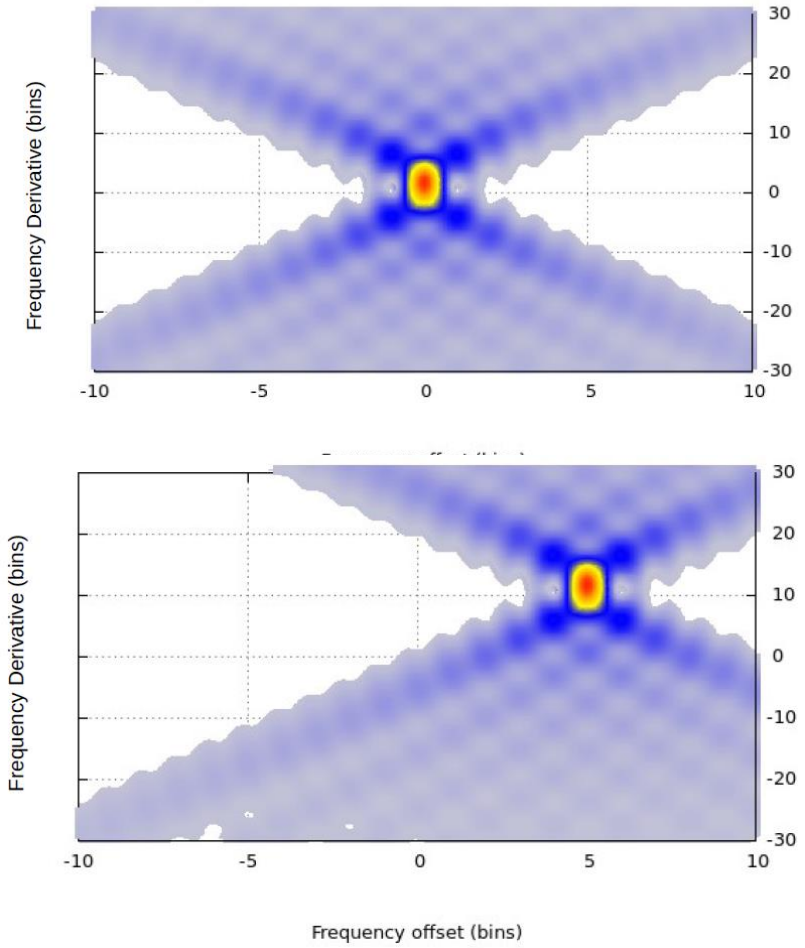
Much like the sound of a siren approaching you and then speeding away.

This can be corrected by using a matched filter approach.



Ransom, Eikenberry, Middleditch: AJ, Vol 24, Issue 3, pp. 1788-1809

FDAS example



The two plots illustrate the effect of orbital acceleration.

The first plot shows a signal without acceleration, the signal is centred on its frequency and lies on the f-dot template corresponding to zero acceleration.

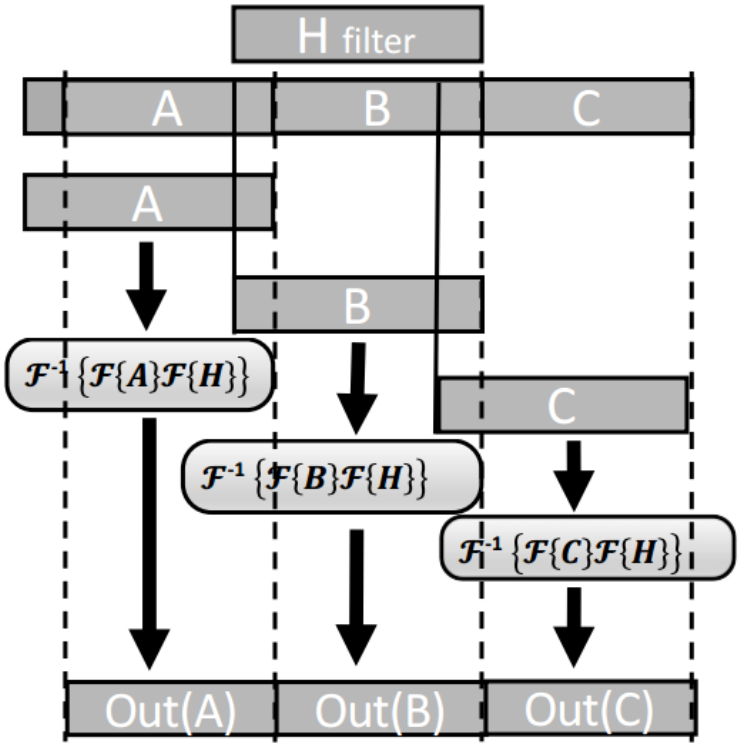
The second plot shows a signal with a frequency derivative, and has drifted from the original frequency by a number of bins.

S. Dimoudi et.al. *Submitted to ApJS.*

FDAS on GPUs

Use overlap-save algorithm to compute cyclic N-point convolution of template with signal segment.

Avoids the need for synchronisation because contaminated ends of convolved data are discarded.



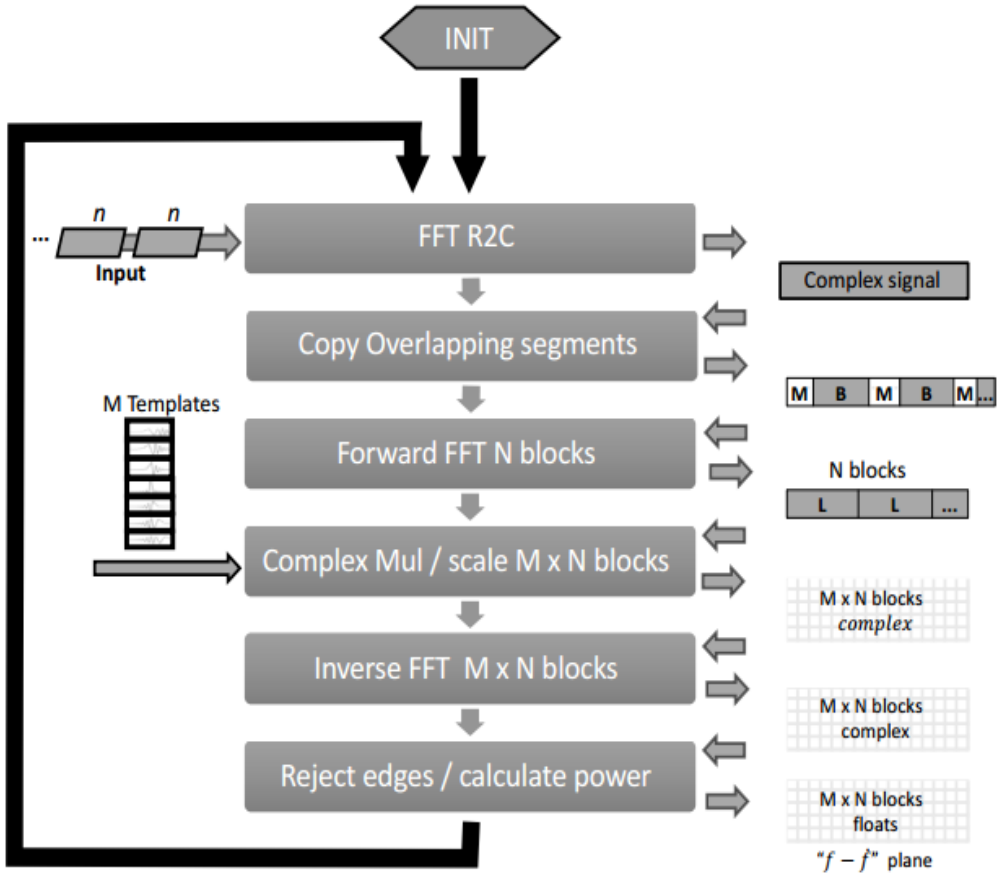
S. Dimoudi et.al. *Submitted to ApJS.*

FDAS on GPUs using cuFFT

Using cuFFT means many transactions to device memory on the GPU (represented by grey arrows on the right of the diagram).

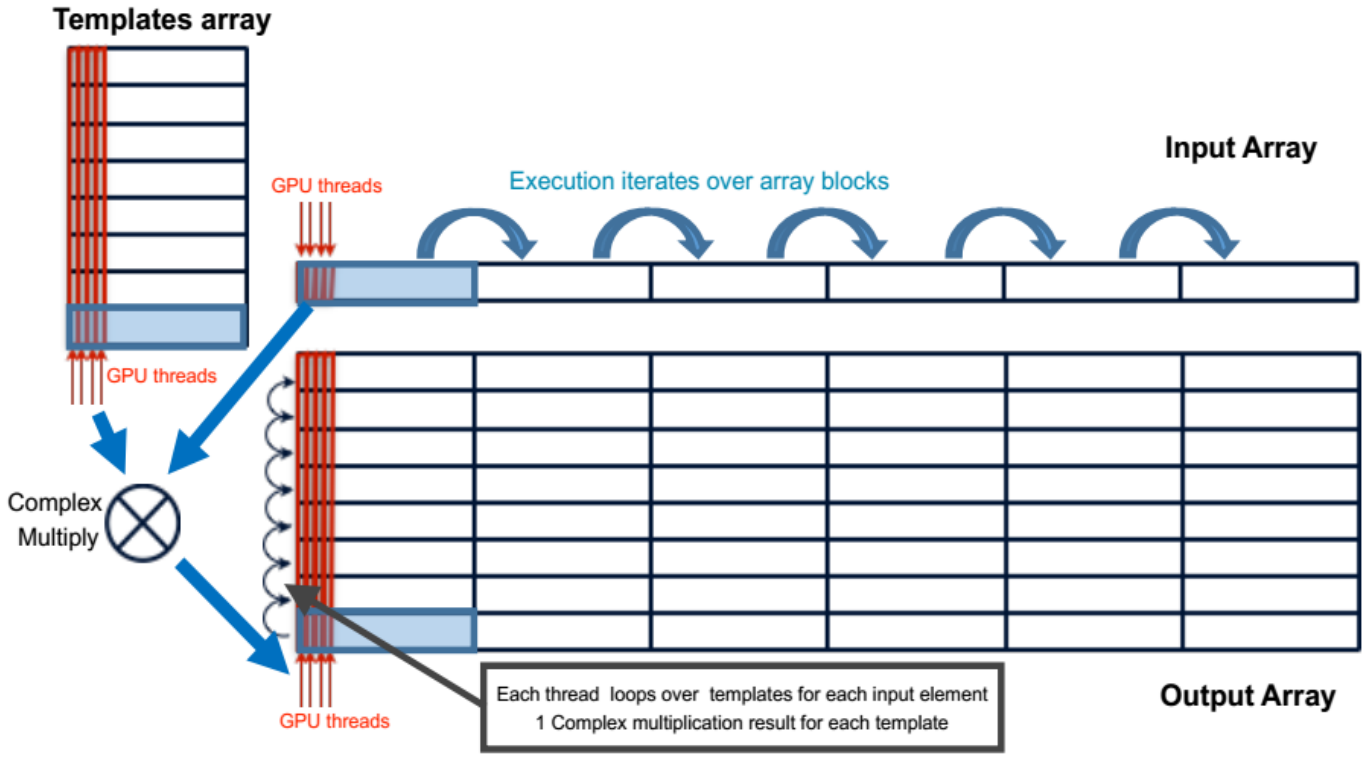
This causes the computation to be limited by global memory bandwidth.

So is slow.



S. Dimoudi et.al. *Submitted to ApJS.*

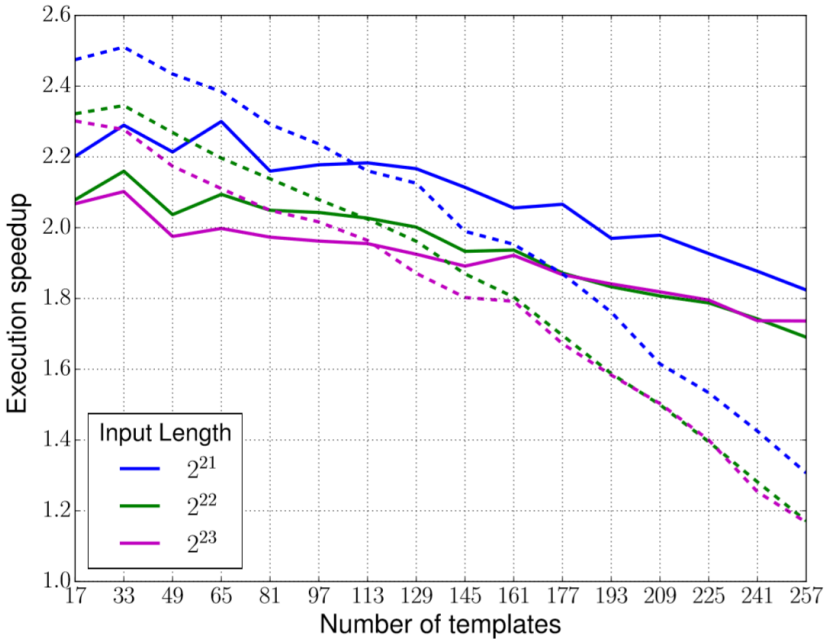
Eliminating the bandwidth bottleneck

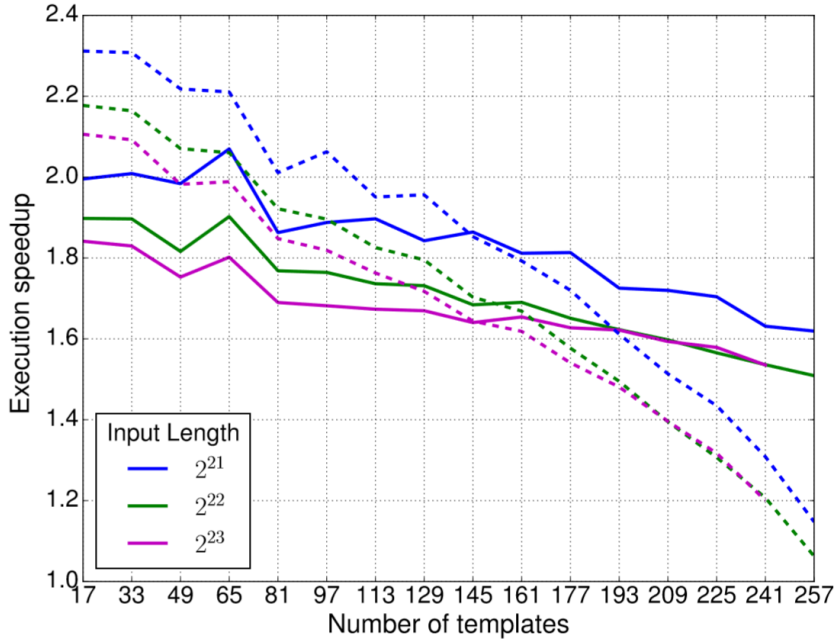


By writing our own custom I/FFT codes to work on shared memory we can perform the FFT, pointwise multiply and scale, IFFT and edge rejection all in one kernel.

S. Dimoudi et.al. *Submitted to ApJS.*

Performance gains on a P100





Results from our tests on a Tesla P100. In the SKA region of interest – signal length 2^{23}, template size of 512 (solid line) and no interbinning (left graph)

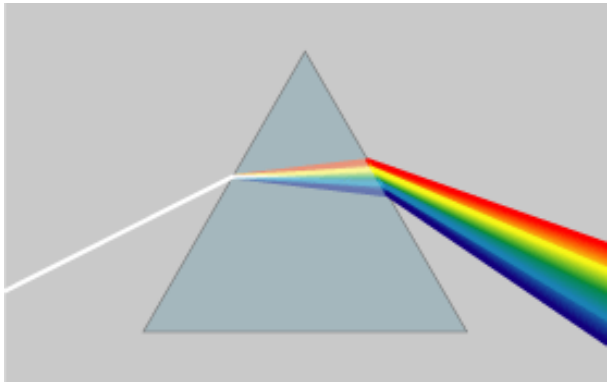
We achieve approximately a 2x speed increase (3x on K80).

S. Dimoudi et.al. *Submitted to ApJS.*

Case study 2: Real-time de-dispersion for the SKA

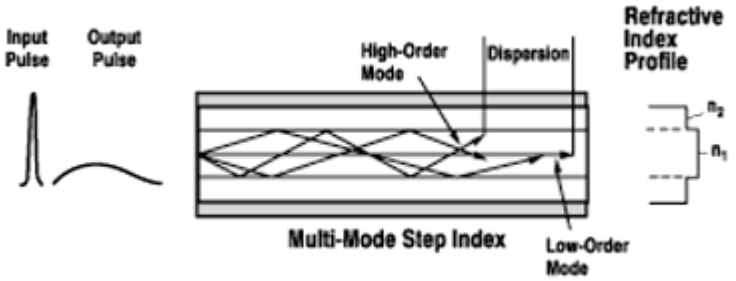
M. Giles (Oxford)
Karel Adámek (Oxford)
Jan Novotný (Opava)
Byron Sinclair (Altera)
Andrew Ling (Altera)
Kate Clark (NVIDIA)
Tim Lanfear (NVIDIA)

What is dispersion



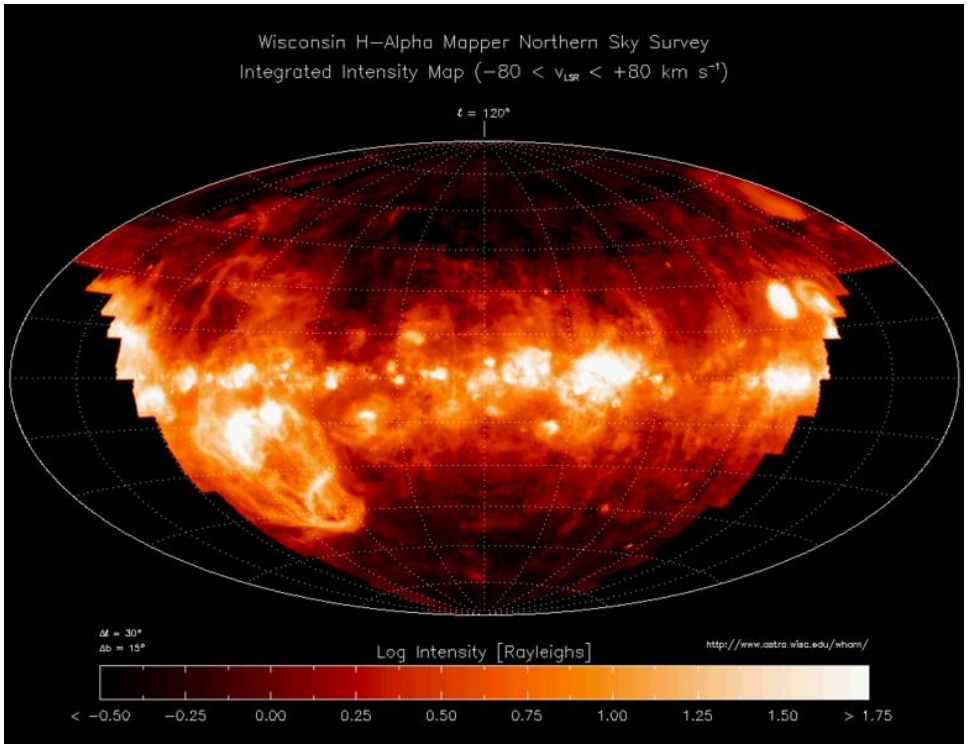
Chromatic dispersion is something we are all familiar with. A good example of this is when white light passes through a prism.

Group velocity dispersion occurs when pulse of light is spread in time due to its different frequency components travelling at different velocities. An example of this is when a pulse of light travels along an optical fibre.



Dispersion by the ISM

The interstellar medium (ISM) is the matter that exists between stars in a galaxy.



Haffner et al. 2003

In warm regions of the ISM (~8000K) electrons are free and so can interact with and effect radio waves that pass through it.

The Dispersion Measure – DM

The time delay, $\Delta\tau$, between the detection of frequency f_{high} and f_{low} is given by:

$$\Delta\tau = C_{DM} \times \mathrm{DM} \times \left(\frac{1}{f_{low}^2} - \frac{1}{f_{high}^2} \right)$$

Where C_{DM} is the dispersion constant. DM is the dispersion measure:

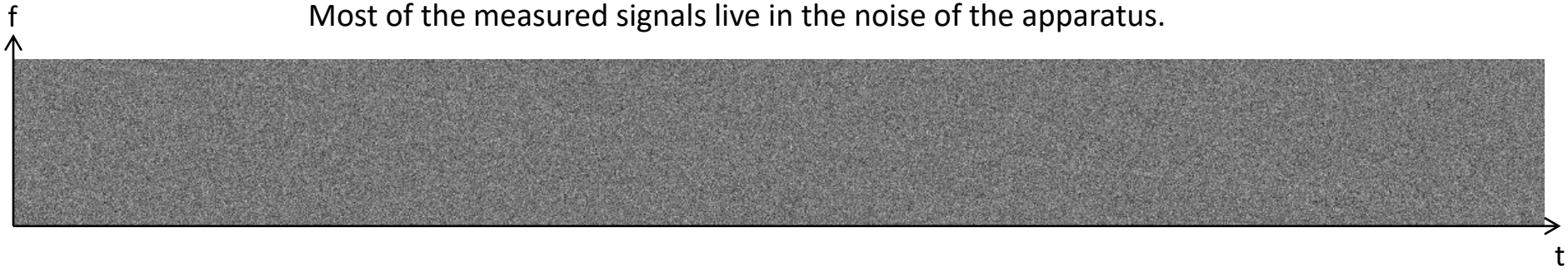
$$DM = \int_0^d n_e dl$$

This is the free electron column density between the radio source and observer.

We can measure $\Delta\tau$ and f and so can study DM

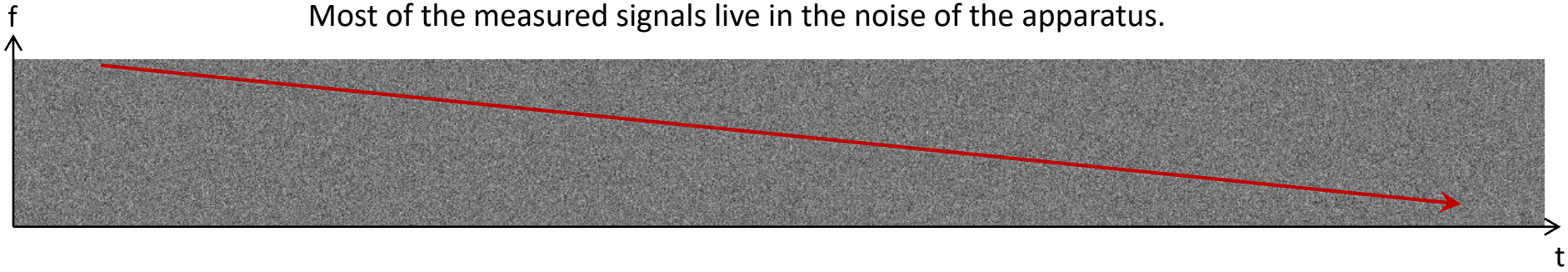
Experimental data…

Most of the measured signals live in the noise of the apparatus.

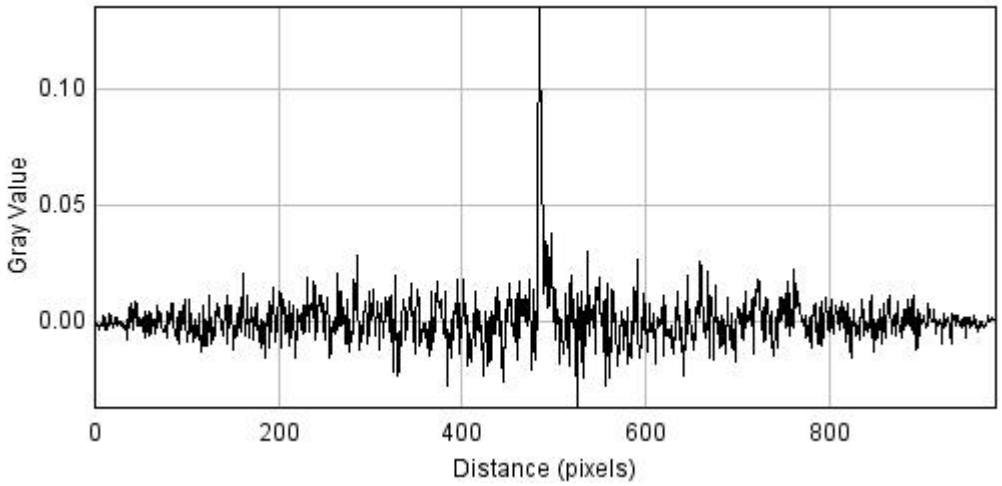


Experimental data…

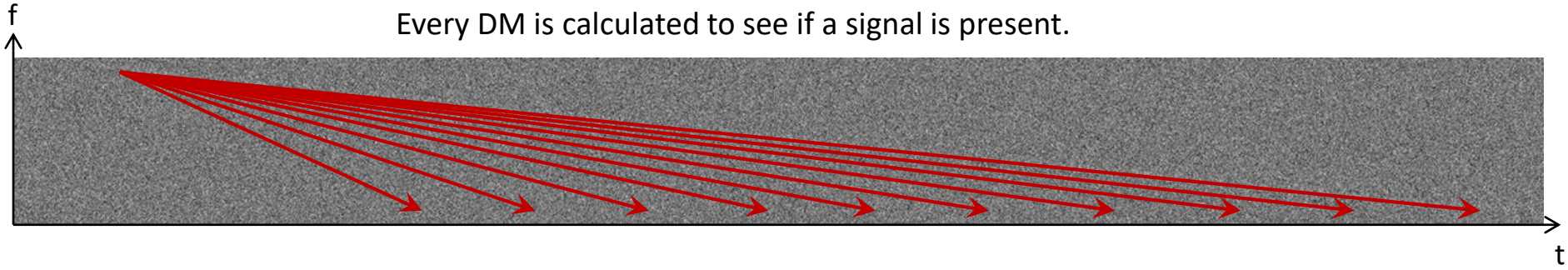
Most of the measured signals live in the noise of the apparatus.



Hence frequency channels have to be “folded”



De-dispersion…



- In a blind search for a signal many different dispersion measures are calculated.
- This results in many data points in the (f,t) domain being used multiple times for different dispersion searches.
- This allows for data reuse in a GPU algorithm.

All of this must happen in real-time i.e. The time taken to process all of our data must not exceed the time taken to collect it

ALTERA OpenCL

Byron Sinclair and Andrew Ling – ALTERA
Jayantha Roy, Prabu Thiagaraj and Ben Stappers (Manchester).

- Worked on a SKA test case using OpenCL on Altera FPGAs.
- OpenCL code is a new implementation based on brute force pseudo-code.
- Code is portable between generations and families of FPGAs.

De-dispersion on NVIDIA GPUs

UNIVERSITY OF OXFORD

Oxford e-Research Centre

Produced algorithms for three generations of GPU

- Fermi generation: L1 Cache and Shared memory (with Mike Giles UOx)
- Kepler generation: Texture Cache, Shared memory (with Kate Clark and Tim Lanfear - NVIDIA).
- Maxwell generation: SIMD in word (with Kate Clark - NVIDIA).
- Pascal generation: Mixed precision instructions & Energy Efficiency (with Kate Clark - NVIDIA).

Intel algorithms

- Use OpenMP to spread work across cores.
- Use vector intrinsics to make use of the AVX units.
- Threads and vectors are arranged so that we gain maximum data re-use in L1 cache.
- Each thread processes 4 dispersion measures (data blocks in L1 cache) and each AVX vector processes 8 or 16 time samples depending on whether we have Ivy Bridge (AVX256) or Xeon Phi (AVX512).

With Karel Adámek and Jan Novotný (Opava).

DDTR on CPUs and Phi

CPU code snippet…

- Process vectors of time, holding DM constant (no blocking).

```
// Declare a local array AVX vectors
__m256 xmm[16]

// Loop over half of the 16 avx registers
for(i = 0; i < 8; i++) {

        // Unaligned load of 8 floats into AVX register i
        xmm[i] = _mm256_loadu_ps(input_buffer+shift+(i*SIMDWIDTH));

        // Add the loaded (f,t) values = xmm[i], to the accumulator register xmm[i+8]
        xmm[i + 8] = _mm256_add_ps(xmm[i], xmm[i + 8]);
}
```

Xeon Phi modifications

Xeon Phi Optimisation

No unaligned load so two cache lines must be loaded and unpacked

```
zmm[i] = _mm256_loadu_ps(input_buffer + (shift+(i*SIMDWIDTH)));
```

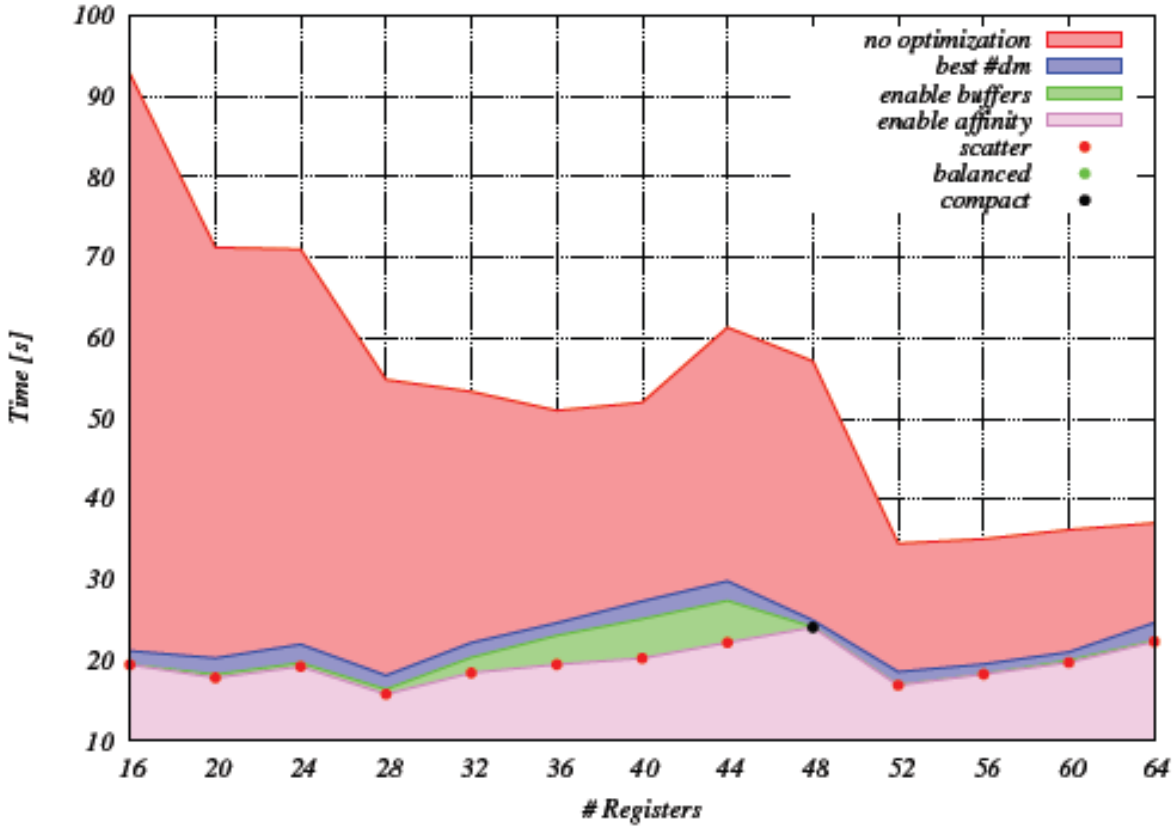
In CPU code must be changed for…

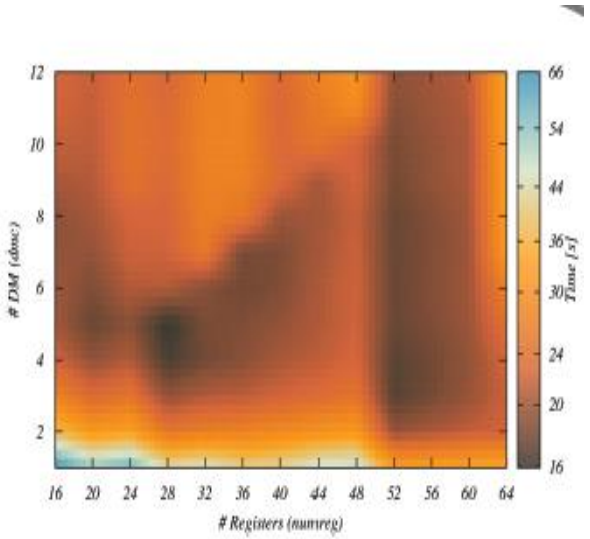
```
zmm[i] = _mm512_loadunpacklo_ps(xmm[i], input_buffer + (shift+(i*SIMDWIDTH));
zmm[i] = _mm512_loadunpackhi_ps(xmm[i], input_buffer + shift+(i*SIMDWIDTH))+16);
```

Intel results

De-dispersion for the SKA using Phi

Wes Armour, Mike Giles, Karel Adámek and Jan Novotný.





Xeon Phi Optimisation

Intel results

Code/Hardware	Fraction of real-time
2x E5-2680 Serial (one core)	0.007
2x E5-2680 OpenMP	0.050
2x E5-2680 OpenMP+Intrinsics	0.300
Xeon Phi 5110P OpenMP+Intrinsics	0.388

Here we see Phi is about 1.3x faster than two high end Xeons.
Because we wrote parameterised code with instrinsic instructions the port from the CPU to Phi was relatively painless.

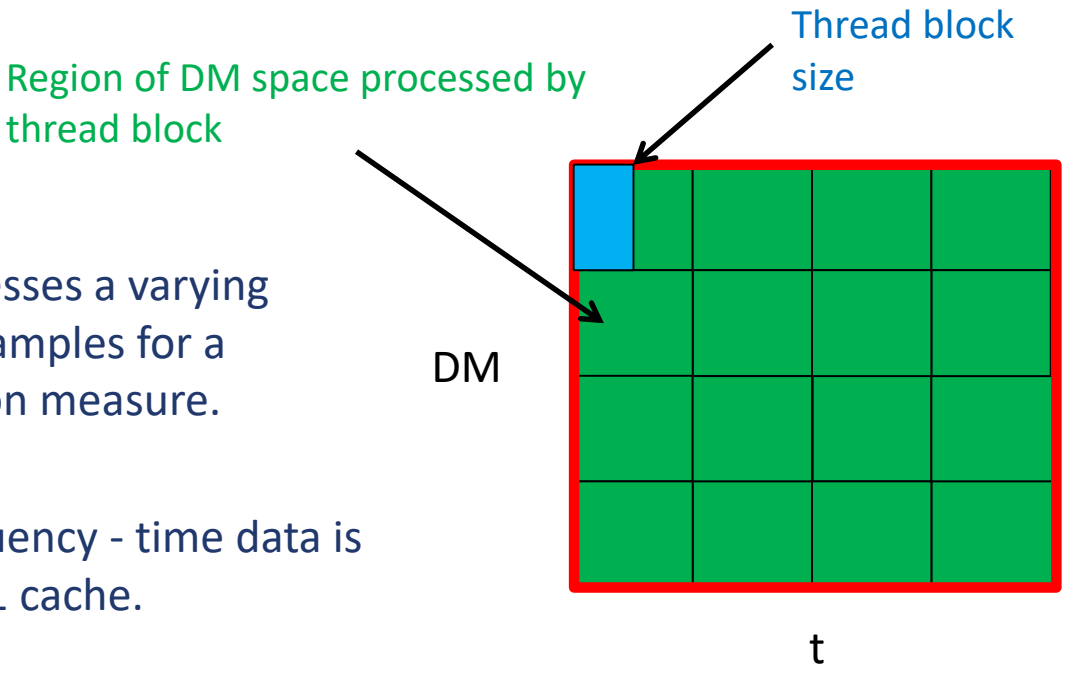
UNIVERSITY OF OXFORD

Oxford e-Research Centre

DDTR on NVIDIA GPUs

Fermi L1 cache algorithm

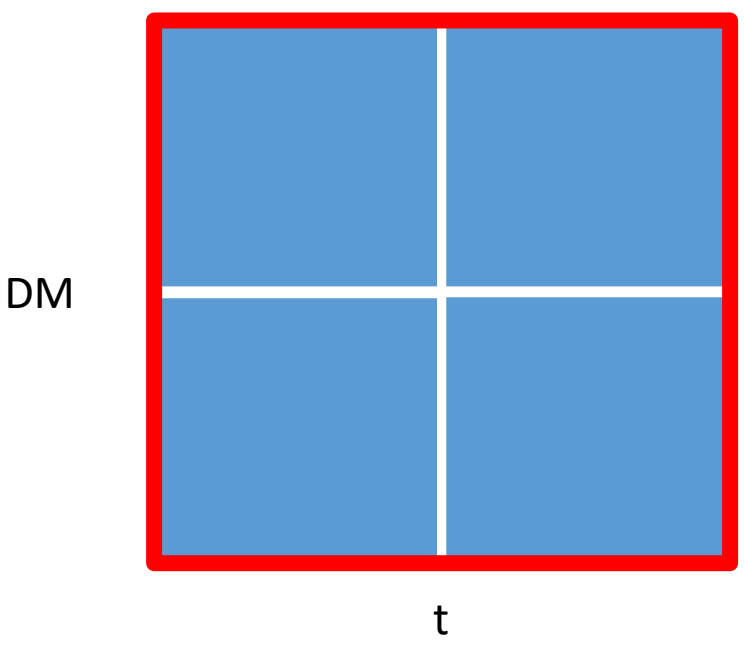
Processing several DM's per thread…

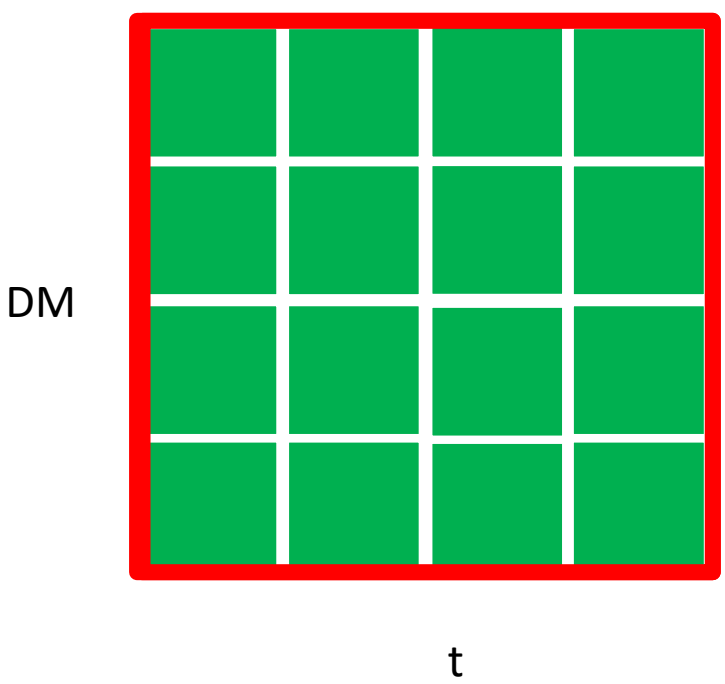


- Each thread processes a varying number of time samples for a constant dispersion measure.
- This ensures frequency - time data is loaded into fast L1 cache.
- Using registers ensures very quick memory access.

Optimising the parameterisation

The GPU block size of the new algorithm can take on any size that is integer multiples of the size of a “data chunk”…

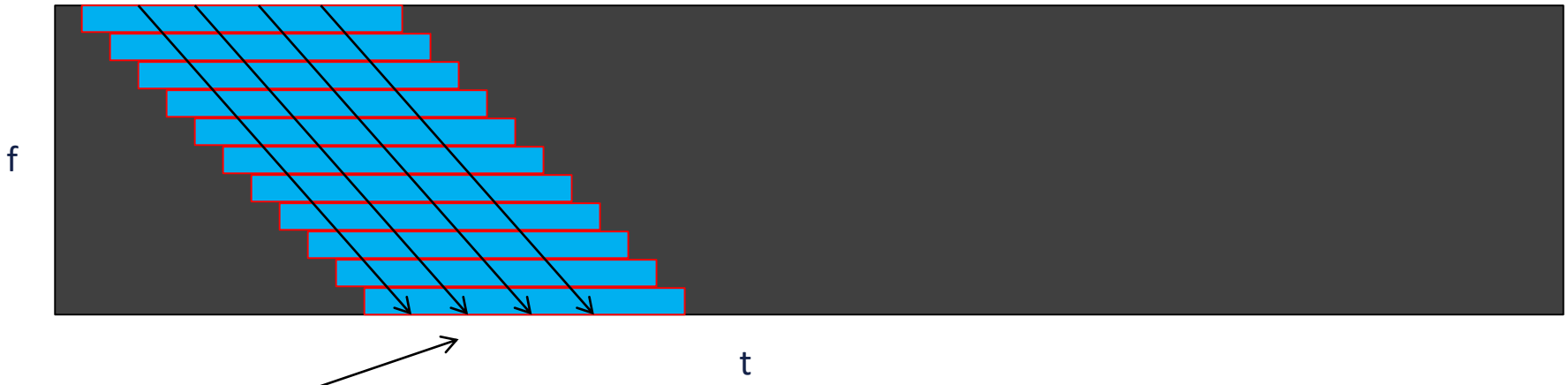




Fermi shared memory algorithm

Exploiting fast shared memory…

Each dispersion measure for a given frequency channel needs a shifted time value.

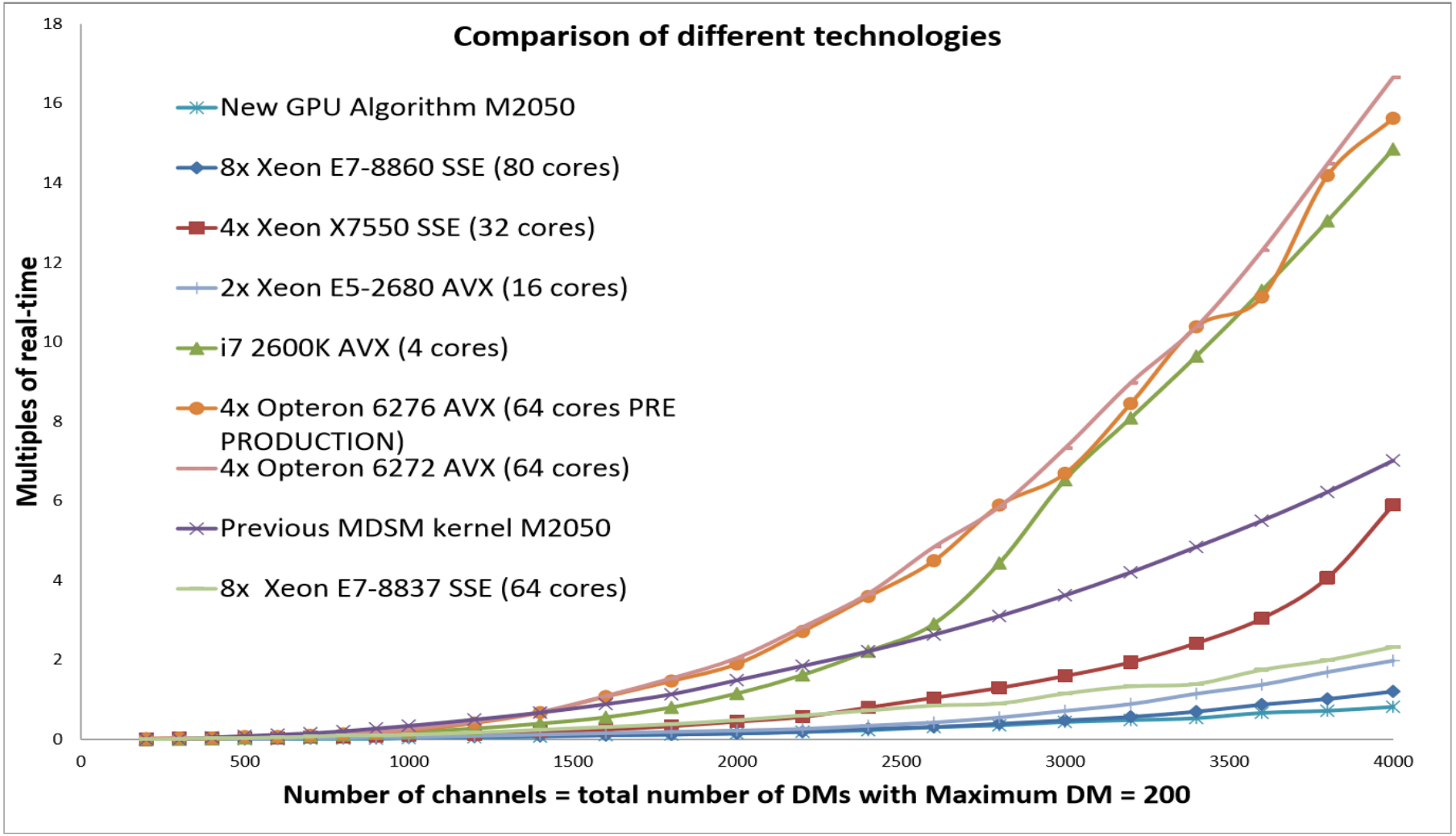


Constant DM's with varying time.

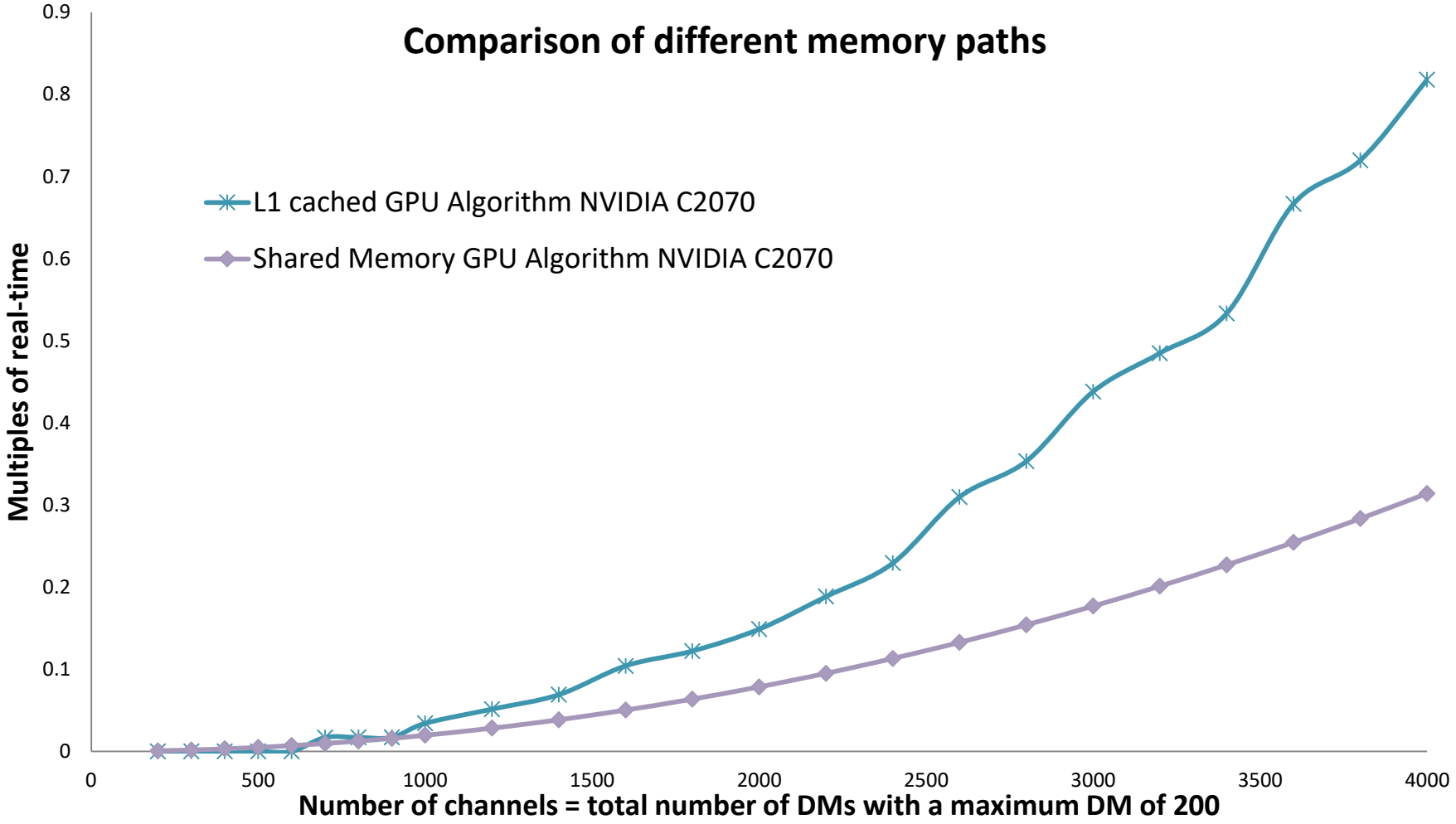
In practice a thread will process multiple time samples and a threadblock will also process neighboring DM trials to increase data reuse.

Incrementing all of the registers at every frequency step ensures a high data reuse of the stored frequency time data in the L1 cache or shared memory.

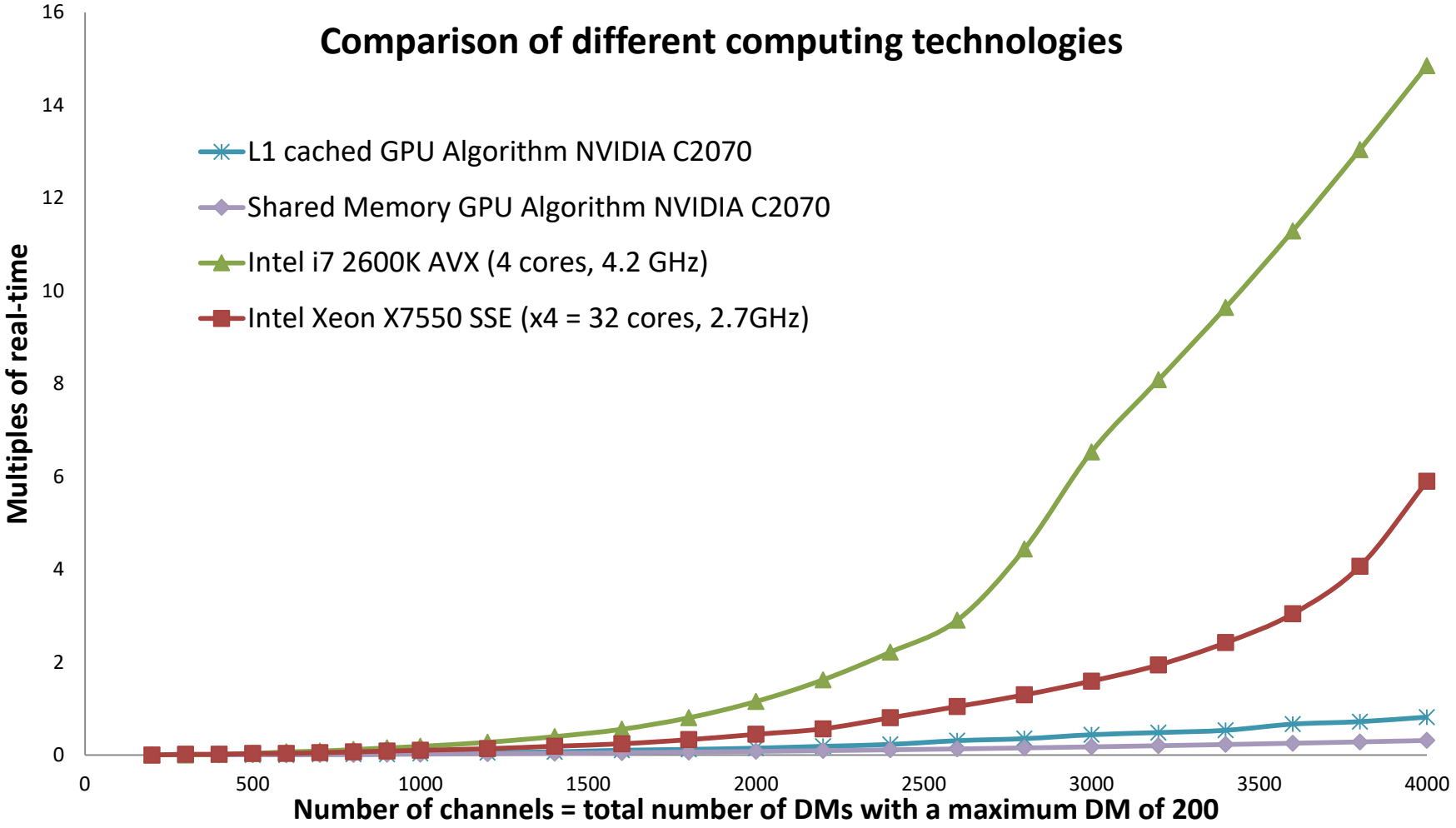
Initial Fermi results compared to CPU code



Initial Fermi results



Initial Fermi results compared to CPU code



Exploiting Shared Memory

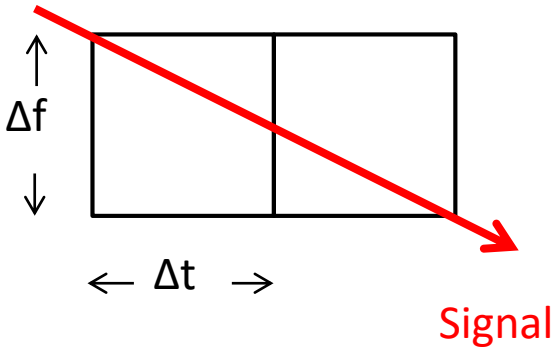
Problem with shared memory algorithm

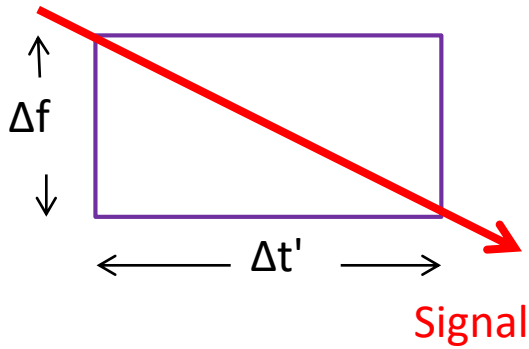
For realistic telescope frequencies/bandwidths and interesting values for the DM trials we need long shared memory lines or need to use a reduced number of accumulators

This causes the performance to drop towards our cache algorithm

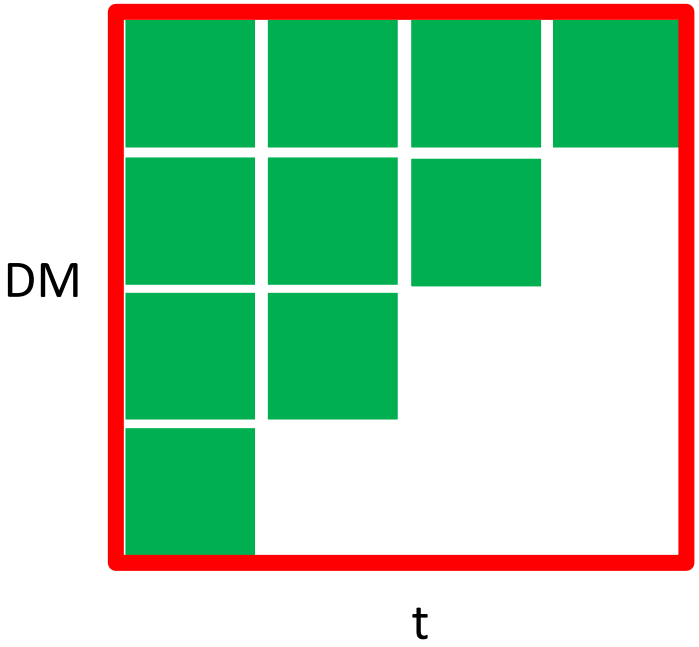
However we can work around this: Time Binning…

Time binning…





Has the effect of reducing the amount threads that are needed to process a region of (DM,t) space.



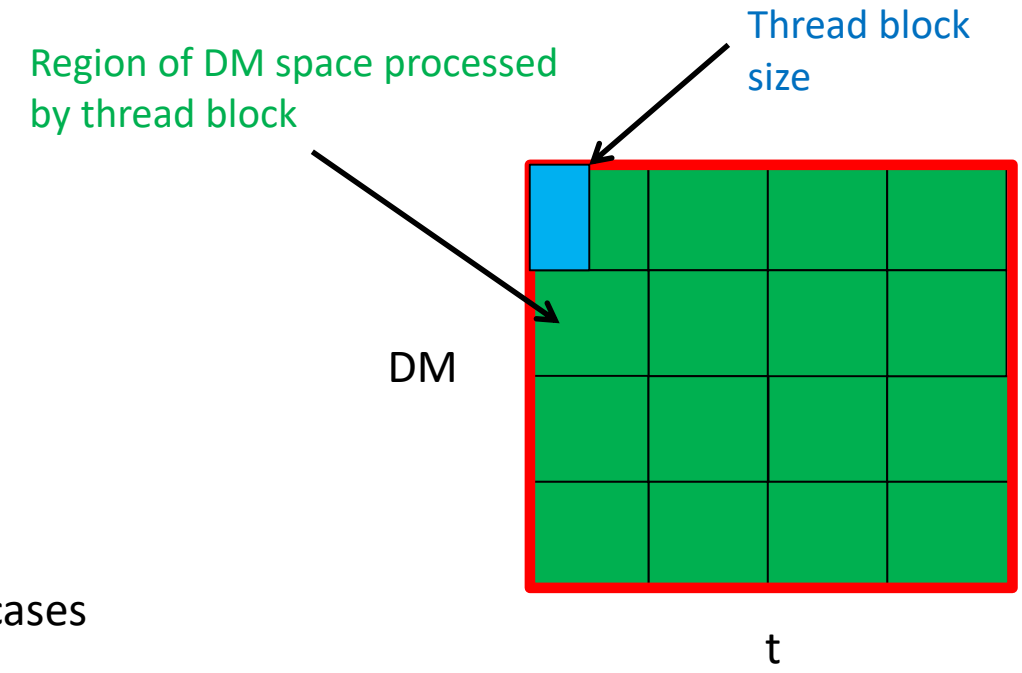
Utilizes the CPU and GPU at the same time (analyze previous de-dispersed data or bin on CPU).

Kepler texture cache algorithm

The Kepler architecture (GK110) introduced a read-only data cache. The requirement on data being the data must be guaranteed read-only for the duration of the kernel (since the read-only data cache is incoherent with respect to writes).

- The read-only data cache is simple to use with the provided __ldg(); intrinsic.
- This allowed for simple re-use of the L1 Fermi algorithm with minimal code alterations.

Produced a 25-30% speedup in some cases



Kepler shared memory algorithm

Kepler introduced shared memory that has 8 byte banks. If your code makes 64 bit transactions to shared memory, has the correct access pattern and correct data alignment then it is possible to get 2x shared memory bandwidth….

For the shared memory code this meant packing data as float2. However the shift that each thread calculates and uses to increment its accumulator isn't known when the data is packed.

Solution: Make each thread calculate two time values: t_i, t_{i+1} and then pack the data in an interleaved (even/odd),(odd/even) format…

Kepler shared memory algorithm

float2[] = $[t_0,t_1][t_1,t_2][t_2,t_3][t_3,t_4][t_4,t_5][t_5,t_6]$ …
float2[] = [0][1][2]

Data is now correctly aligned for 64 bit access

For thread with an even shift (lets say 2)…

(t_0,t_1) (t_1,t_2) (t_2,t_3) (t_3,t_4) (t_4,t_5) (t_5,t_6)

$t_i = t_2$

$t_{i+1} = t_3$

For thread with an odd shift (lets say 3)…

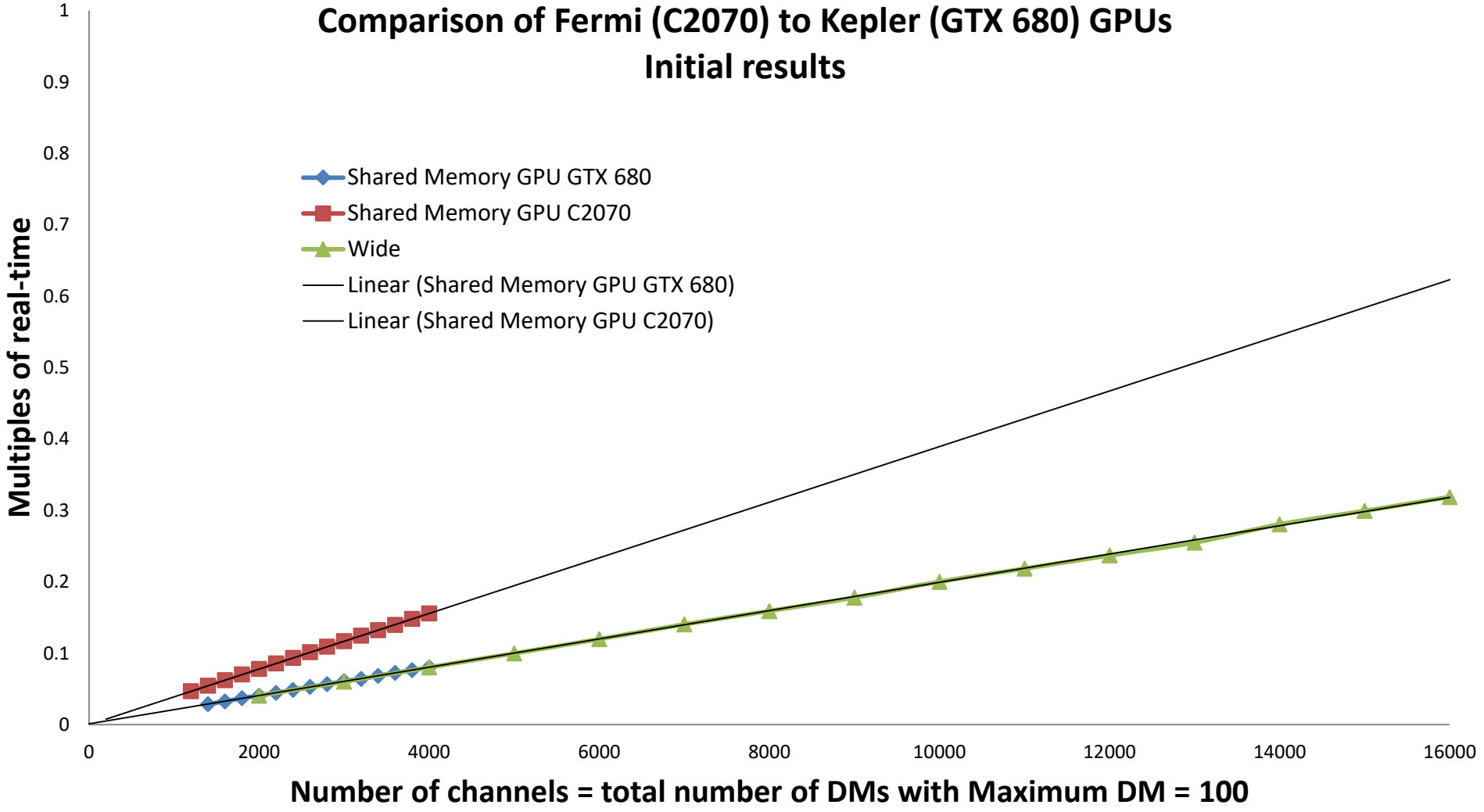
(t_0,t_1) (t_1,t_2) (t_2,t_3) (t_3,t_4) (t_4,t_5) (t_5,t_6)

$t_i = t_3$

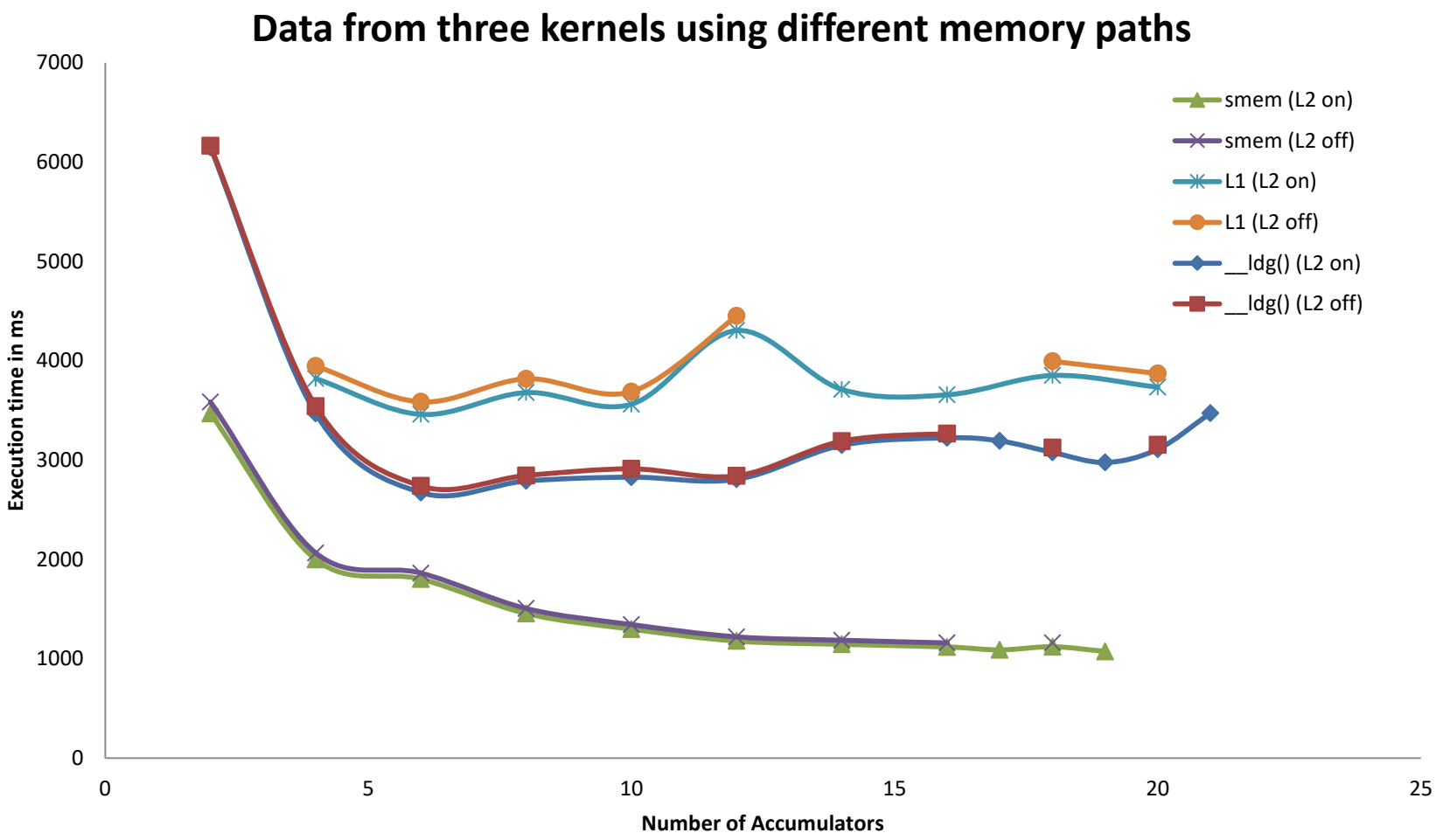
$t_{i+1} = t_4$

Now each thread computes the correct two time values and at double data rate

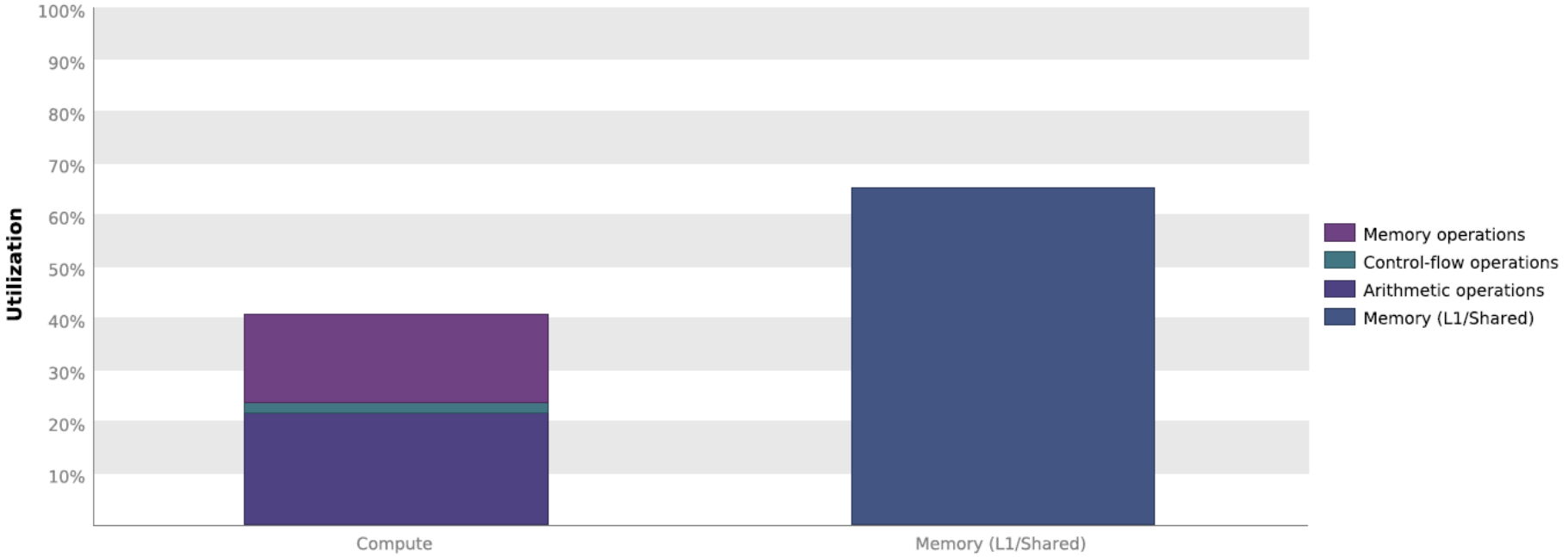
Comparison of Fermi to Kepler



Comparison of Kepler data paths



Profiling the Kepler GTX 780Ti



L1/Shared Memory

Local Loads	0	0 B/s	
Local Stores	0	0 B/s	
Shared Loads	7549747200	2,570.461 GB/s	
Shared Stores	629145600	214.205 GB/s	
Global Loads	0	0 B/s	
Global Stores	19833600	1.533 GB/s	
Atomic	0	0 B/s	
L1/Shared Total	8198726400	2,786.199 GB/s	Idle Low Medium High Max

Maxwell shared memory algorithm

- To simplify algorithm design and programming Maxwell returned to using 4 byte banks. This means an effective reduction in the shared memory bandwidth.
- The Maxwell architecture is more energy efficient – very important when trying to put HPC in a very inhospitable environment (as is the case for SKA).
- However the reduction in shared memory bandwidth isn't good for a code that is limited by shared memory bandwidth.

Solution? At the start of the talk I mentioned that SKA data will be 8 bits per sample. Try to exploit this to increase the shared memory bandwidth. Use the same data access scheme as Kepler with ushort2 or add either a pairing function or combine values…

Maxwell shared memory algorithm

Store data in a zero/odd/zero/even interleaved fashion

In a 32 bit word we split the information into 4 lots of 8 bits.

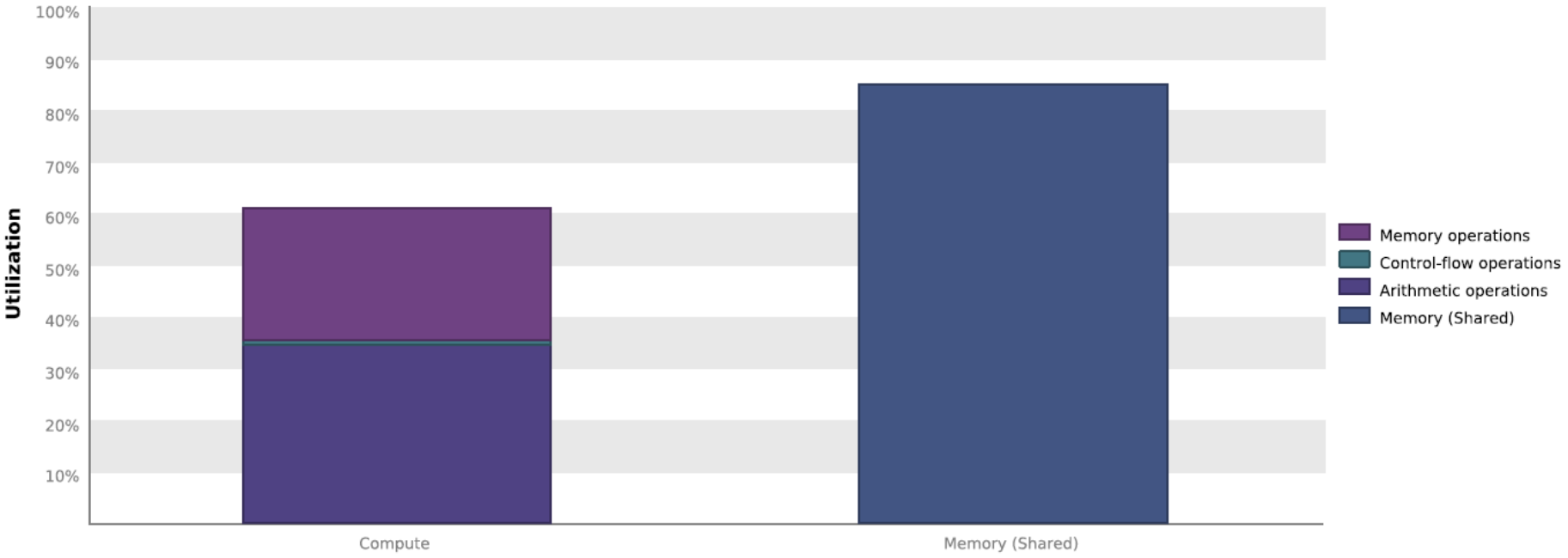
| 0 | data odd | 0 | data even | ….

So we have two data samples per 32 bit word.

By summing the 32 bit integers we achieve two additions for the price of one.

We can accumulate sums up to 16 bits of information before the upper and lower half of the integer have to be “unloaded” to a 32 bit float

Profiling the Maxwell GTX 980

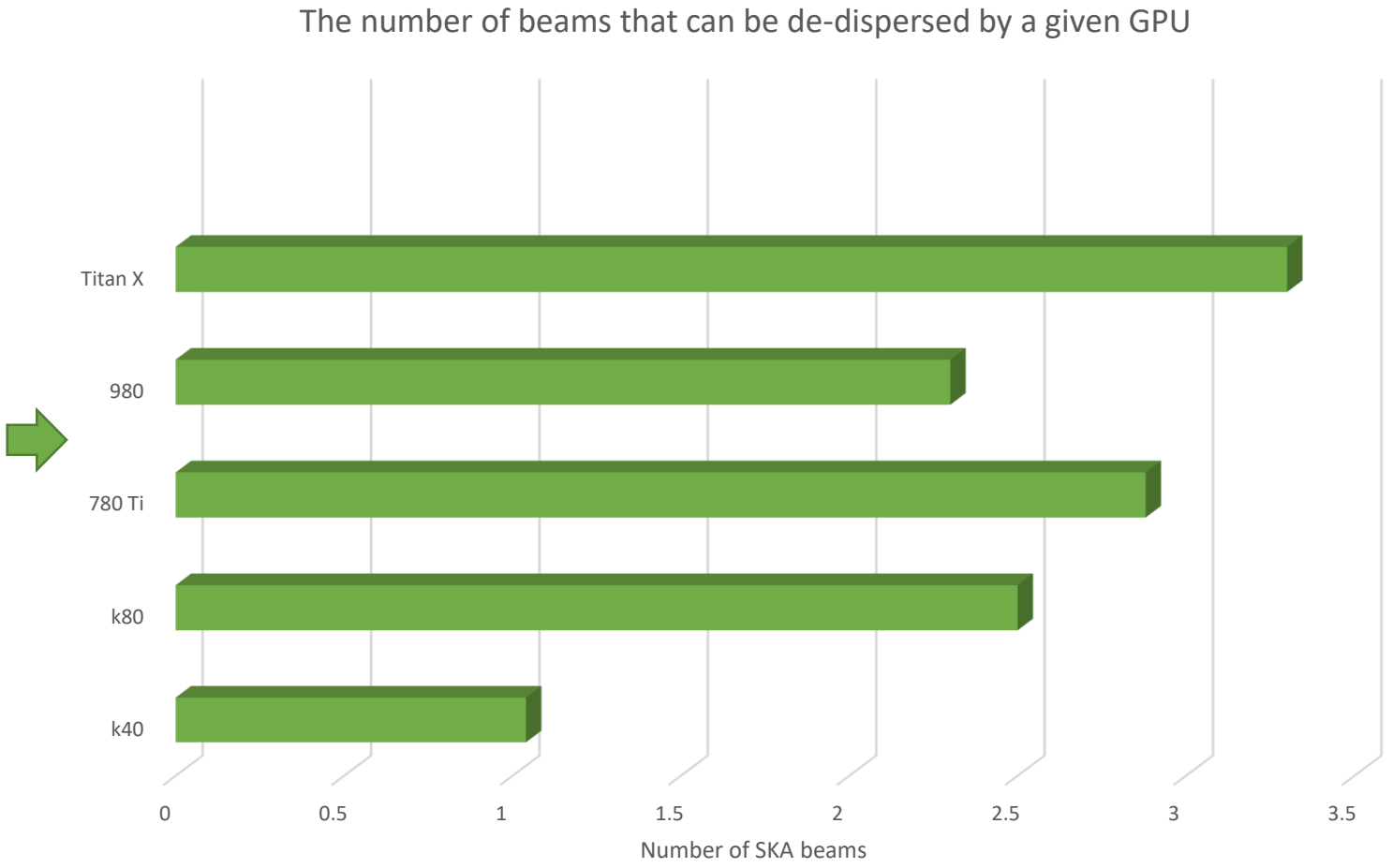


Note: Although Maxwell has less effective shared memory bandwidth we achieve a greater percentage of peak - 85% compared to Kepler's 65%

De-dispersion on NVIDIA GPUs

Changing from the Kepler to Maxwell generation of GPUs gives a **3x** performance increase.

The Titan X can process over 3 SKA beams at once



Kate Clark (NVIDIA), J. Novotny (Opava), W. Armour (OeRC, UOx)

UNIVERSITY OF OXFORD

Oxford e-Research Centre

GPUs vs FPGAs ?

Altera results

Technology	Stratix V	Arria 10	NVIDIA Titan X
Fraction of real-time	1.38	2.17	4.45
Watts per beam (Average)	21.7 W	~10 W	21.1 W
Cost per beam (capital, accelerator only)	~£5K	~£5K	~£180
Cost per beam (2 year survey, GPU only, based on 1KWh costing £0.2)	~£76	~£38	~£74

This comparison is done using a reduced de-dispersion search. 2500 dm trials with no decimation in time. This has been done to make for a clean and easy comparison.

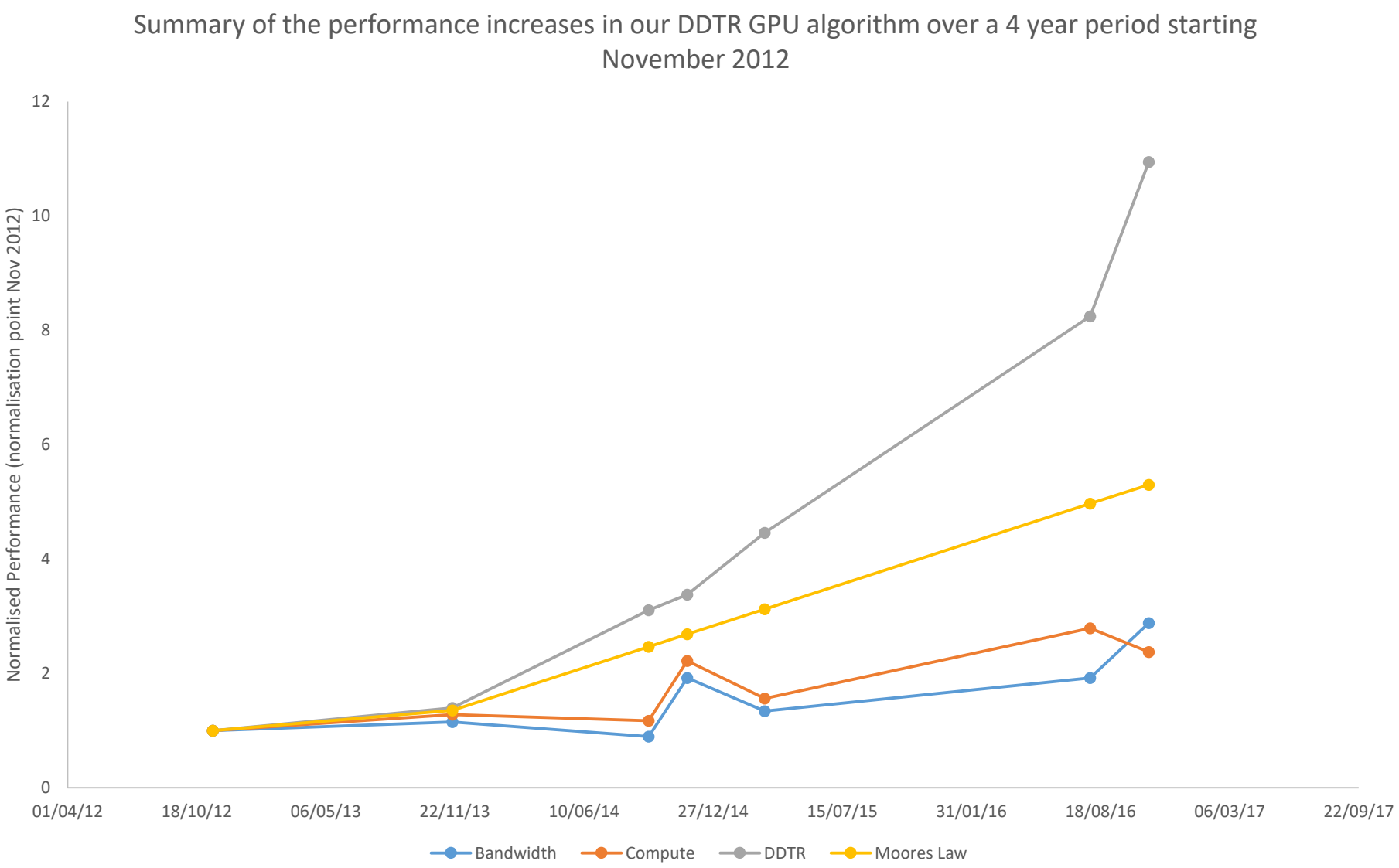
UNIVERSITY OF OXFORD

Oxford e-Research Centre

Where are we now?

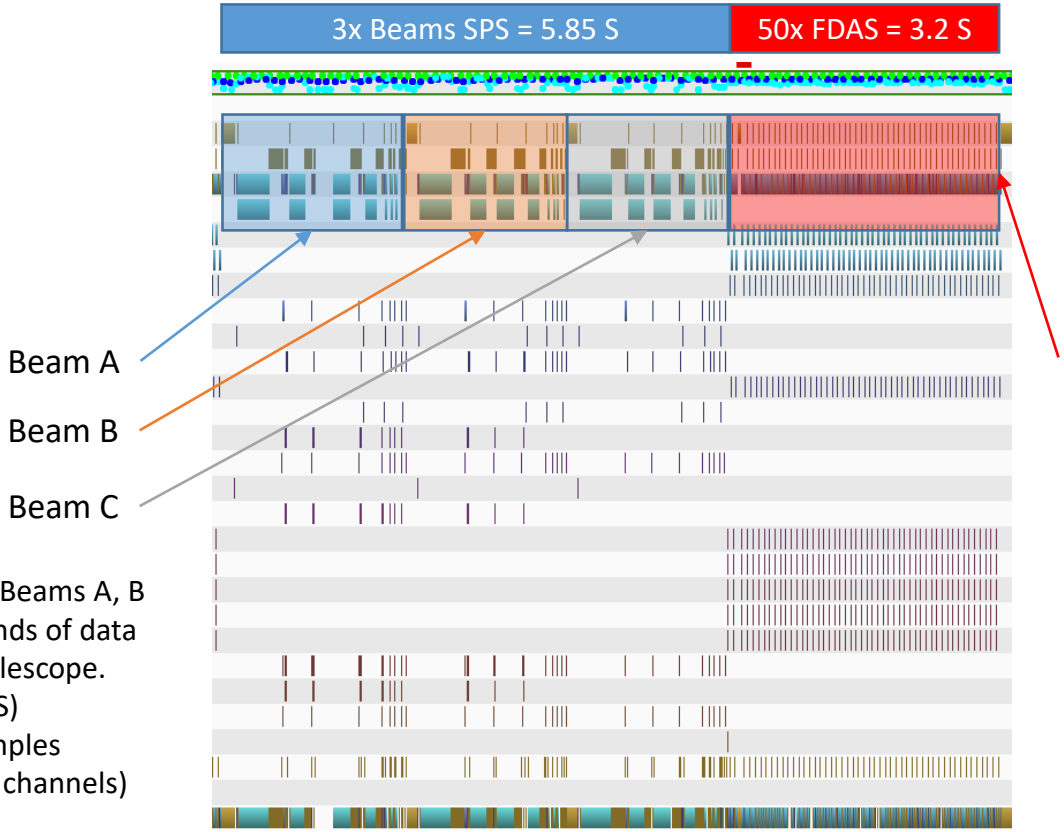
Pascal

The last four generations…



Where we are…

3x Beams of SPS using 9.63S input chuncks and 50 FDAS trials.
Total Processing taking about 9 seconds -> Faster than real-time.



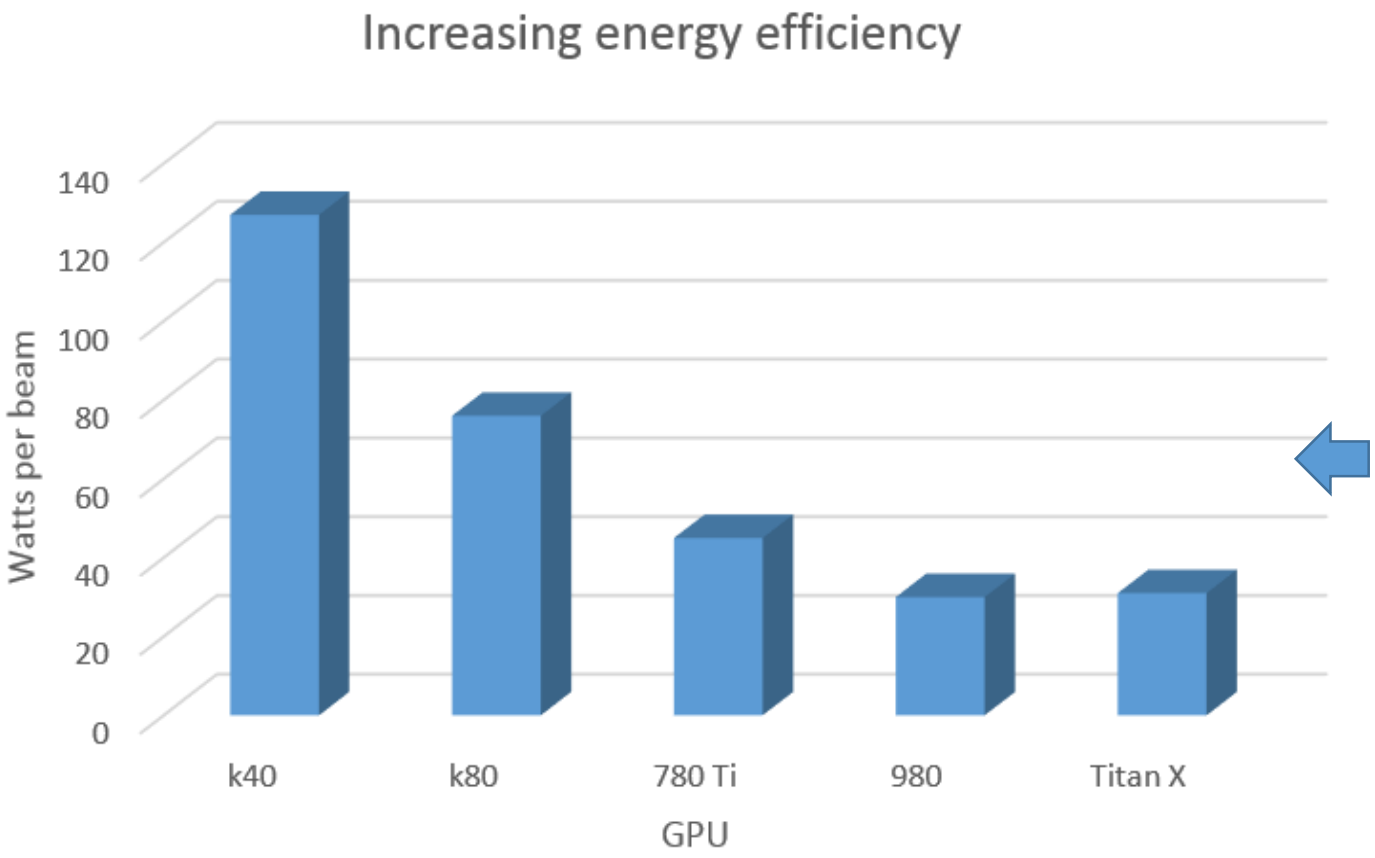
NVIDIA Profiler output from a run on a Tesla P100 GPU

This shows 50 full resolution FDAS trials being performed from a pervious pointing:
2^23 samples using 96 templates.
NO HARMONIC SUM

The input data for Beams A, B and C is 9.63 Seconds of data collected by the telescope.
9.63(S)/0.000064(S)
= 150500 time samples
(each having 4096 channels)

Given that each observation is about 536 seconds long this means that it is possible to perform 536/9.7 x 50 = 2750 full resolution FDAS trials while performing SPS on 3x beams. Current work indicates that the harmonic sum will (at most) half this -> **450 FDAS trials per beam**

What about the Power?



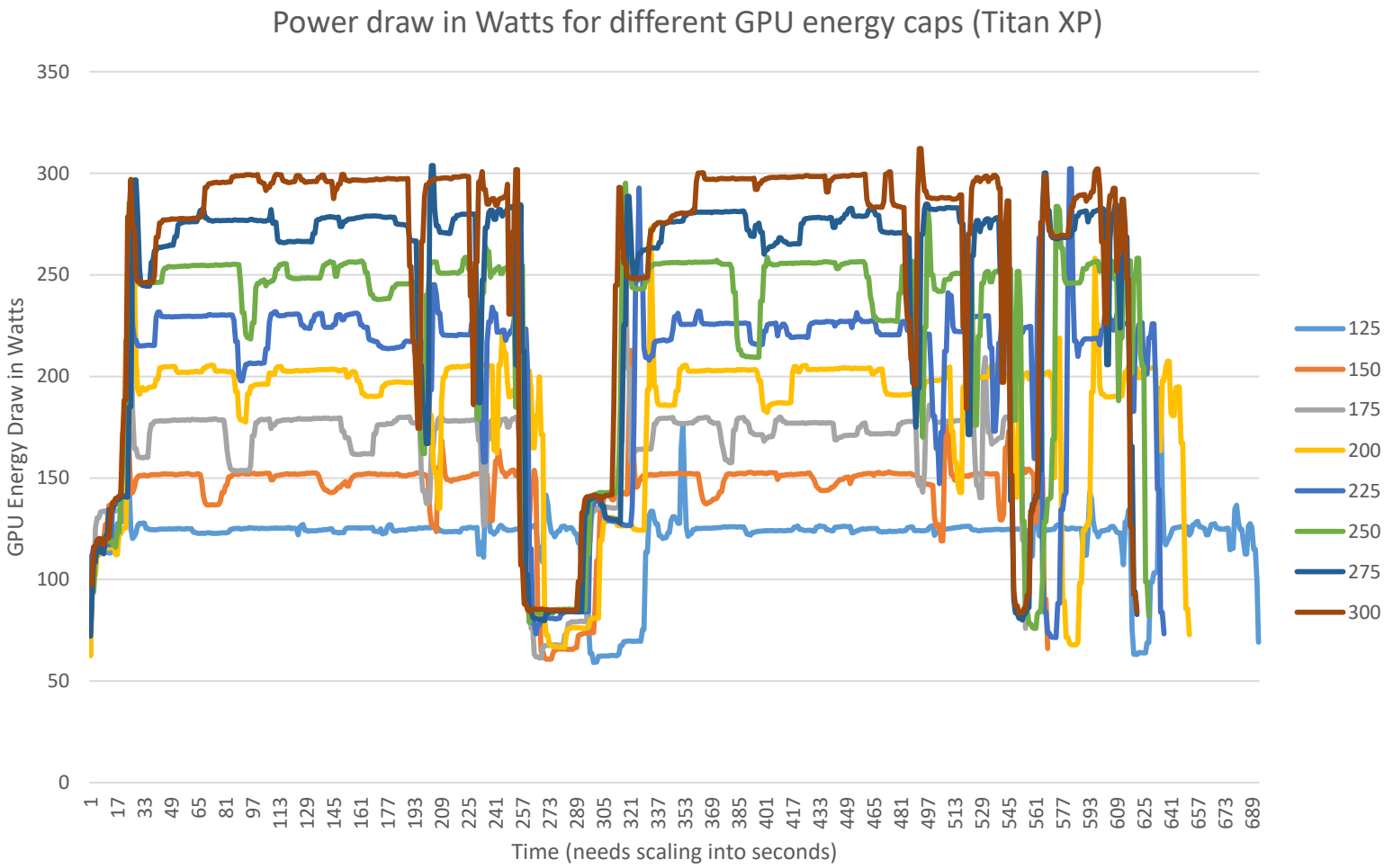
As GPUs evolve so does their energy efficiency

This plot shows that the NVIDIA Titan X is **4x** more energy efficient than the NVIDIA K40 when performing DDTR

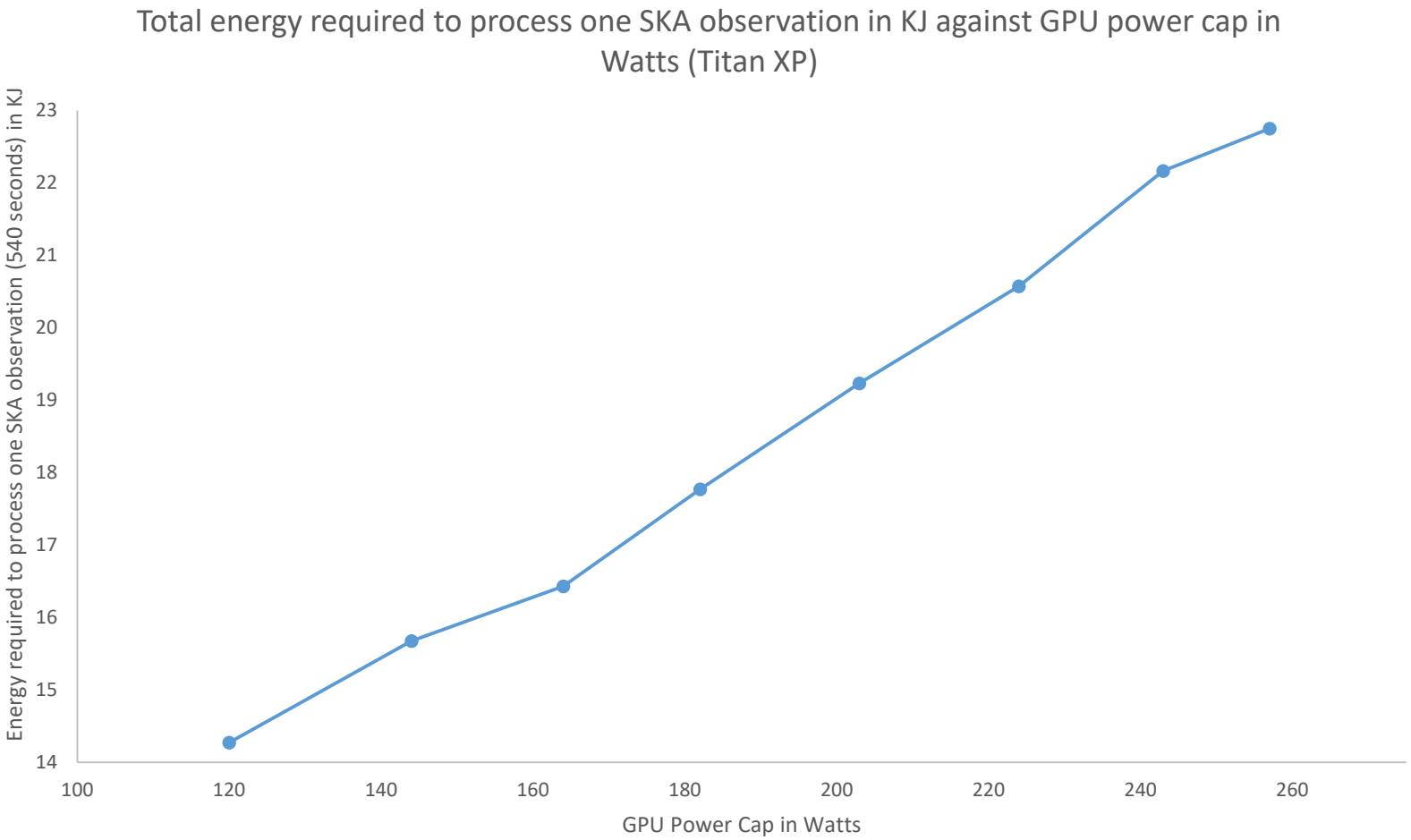
Improvement between generations comes from a combination of advances in both the hardware and algorithm

Kate Clark (NVIDIA), J. Novotny (Opava), M. Giles, W. Armour (OeRC, UOx)

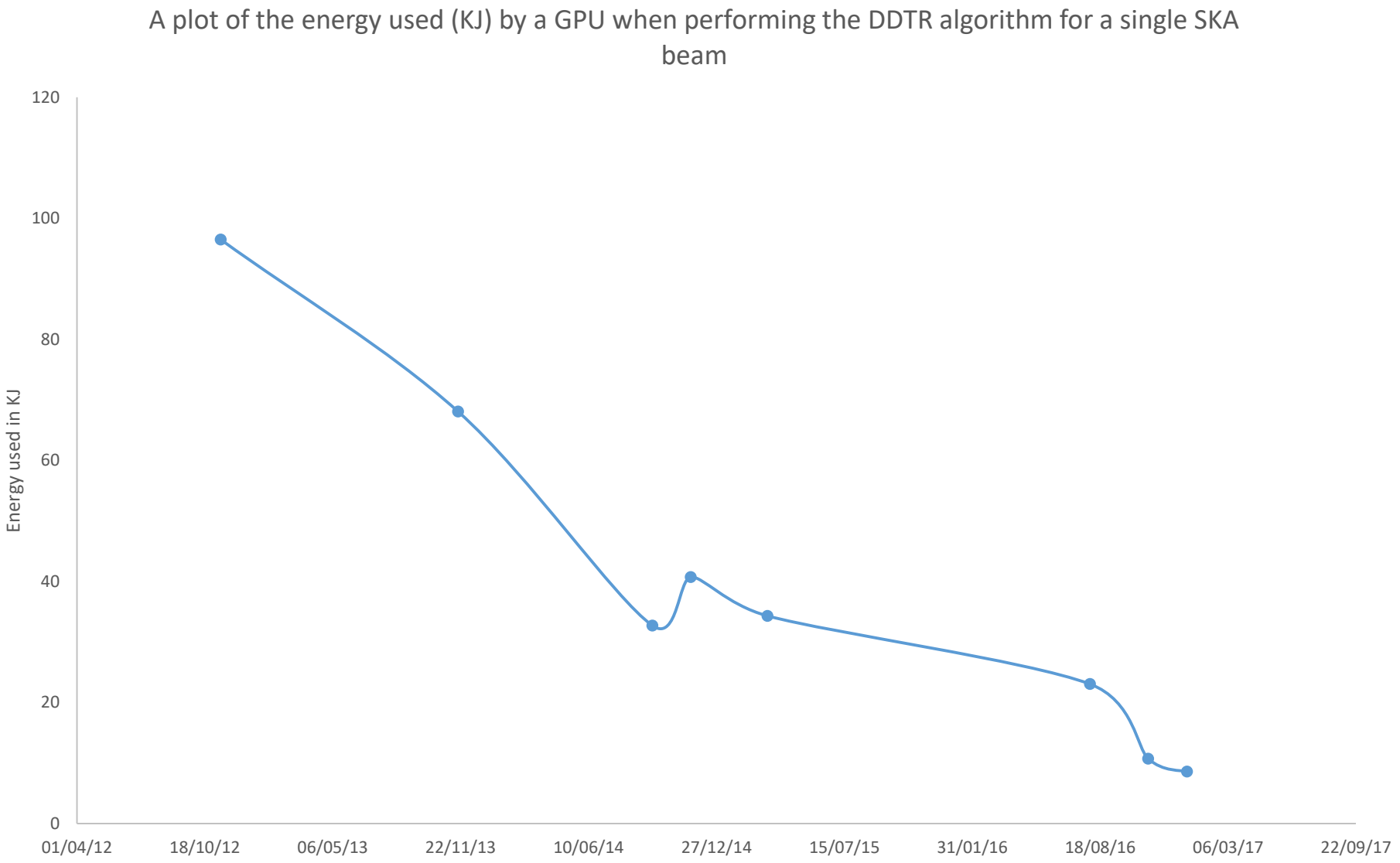
What about the Power?



Energy needed to process one observation



What about the Power?



Conclusion: comparison of GPUs

Technology	Kepler (K40)	Kepler (K80)	Kepler (780Ti)	Maxwell (980)	Maxwell (Titan X)	Pascal (Titan XP)	Pascal P100
Fraction of real-time	1.035	2.5	2.88	2.3	3.3	6.1	8.1
Watts per beam (Average)	127W	76 W	~70W	~61W	~64W	~43W	~24W
Cost per beam (capital, accelerator only)	£3K?	£4K?	£250	£200	£240	~£200	£500?
Cost per beam (2 year survey, GPU only, based on 1KWh costing £0.2)	~£430	~£265	~£245	~£213	~£224	~£151	~£85

Improvement between generations comes from a combination of advances in both the hardware and algorithm

Conclusions

GPU technologies enable us to process multiple SKA beams in real-time.

Equivalent to searching **50 hours of HDTV data every second**.

This will allow us to search our universe for undiscovered exotic objects like FRBs and test Einstein's Theory of General Relativity.

Acknowledgments and Collaborators

University of Oxford

Mike Giles (Maths)

Aris Karastergiou (Physics)

Chris Williams (Physics)

Steve Roberts (Engineering)

Sofia Dimoudi (OeRC)

Nassim Ouannoughi (OeRC)

Karel Adamek (OeRC)

Jayanth Chennamangalam (Physics)

University of Manchester

Ben Stappers

Mike Keith

Prabu Thiagaraj

Jayanta Roy

Mitch Mickaliger

University of Bristol

Dan Curran (Electrical Engineering)

Simon McIntosh Smith (Electrical Engineering)

ASTRON

Cees Bassa

Jason Hessels

University of Opava

Jan Novotny

NVIDIA

Kate Clark

Tim Lanfear

Tom Bradley

ALTERA

Byron Sinclair

Andrew Ling

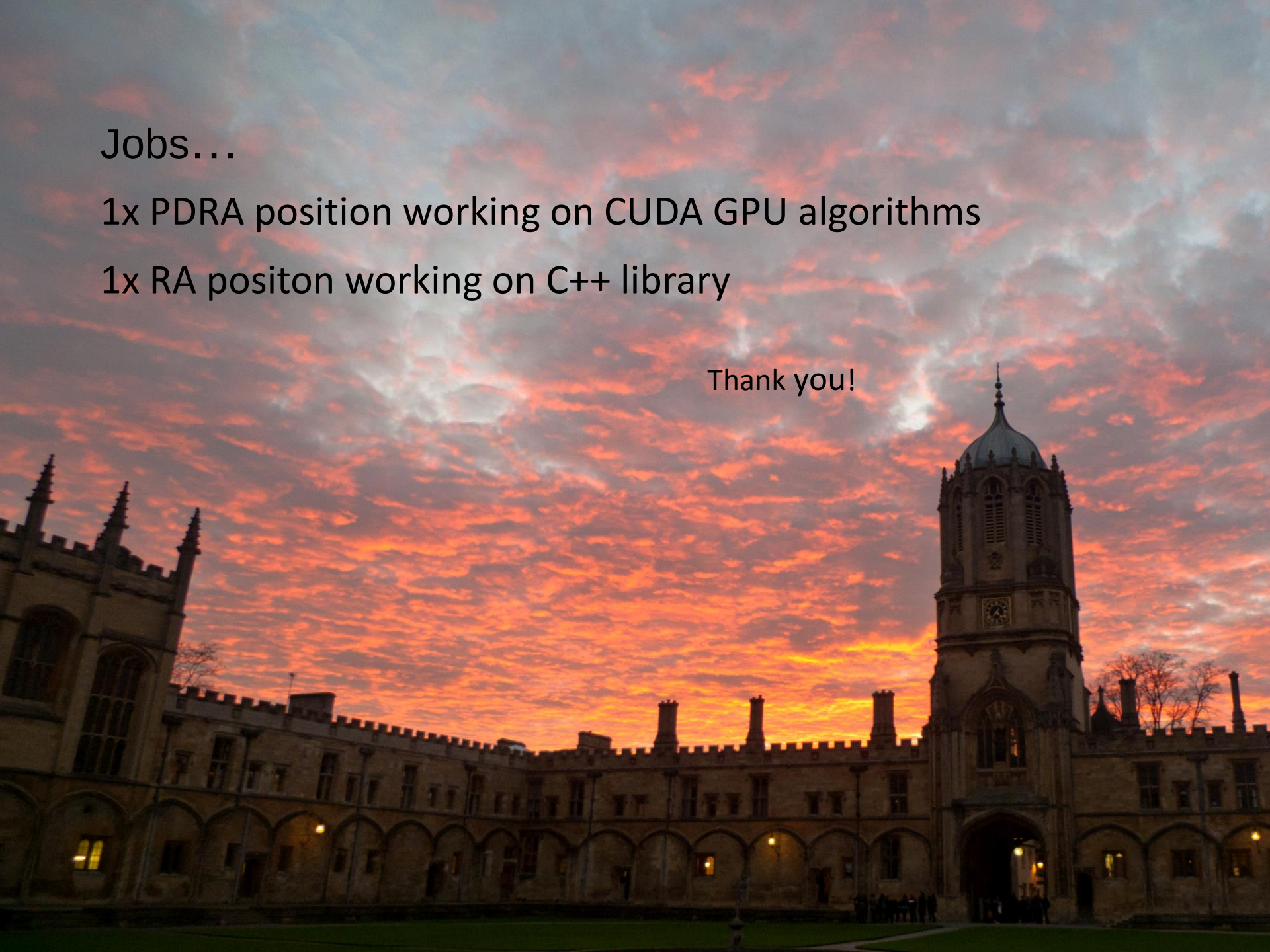
Steve Casselman

Max Plank

Ewan Barr

Astro-Accelerate : http://www.oerc.ox.ac.uk/projects/astroaccelerate

ARTEMIS : http://www.oerc.ox.ac.uk/projects/artemis



Jobs…

1x PDRA position working on CUDA GPU algorithms

1x RA positon working on C++ library

Thank you!