Imaging Nearby Habitable Planets with the Largest Astronomical Telescopes and GPU-powered Adaptive Optics Algorithms

Damien Gratadour

Australian National University (Australia) & Observatoire de Paris (France)

Olivier Guyon

Subaru Telescope, National Astronomical Observatory of Japan, (Japan)

University of Arizona (USA)

Astrobiology Center, National Institutes for Natural Sciences (Japan)

Breakthrough Initiatives (USA)

Hatem Ltaief

King Abdullah University of Science and Technology (Saudi Arabia)

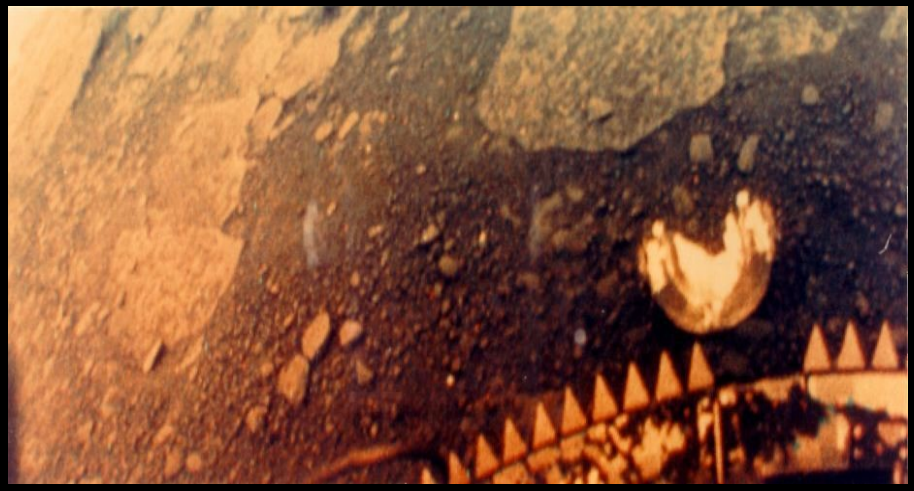
GTC 2019

Olivier @ 3am this morning - live connection with Maunakea @ 4200m elevation
… and Australia

What makes planets habitable ?

(Habitable = could harbor liquid water / carbon-based life as we know it)

The planet must be in the **habitable zone** of its star: not be too close or too far



Venera 13 lander, survived 127mn at 457 C, 89 atm

Venus: too close, too hot

Image credit: NASA/JPL-Caltech/MSSS

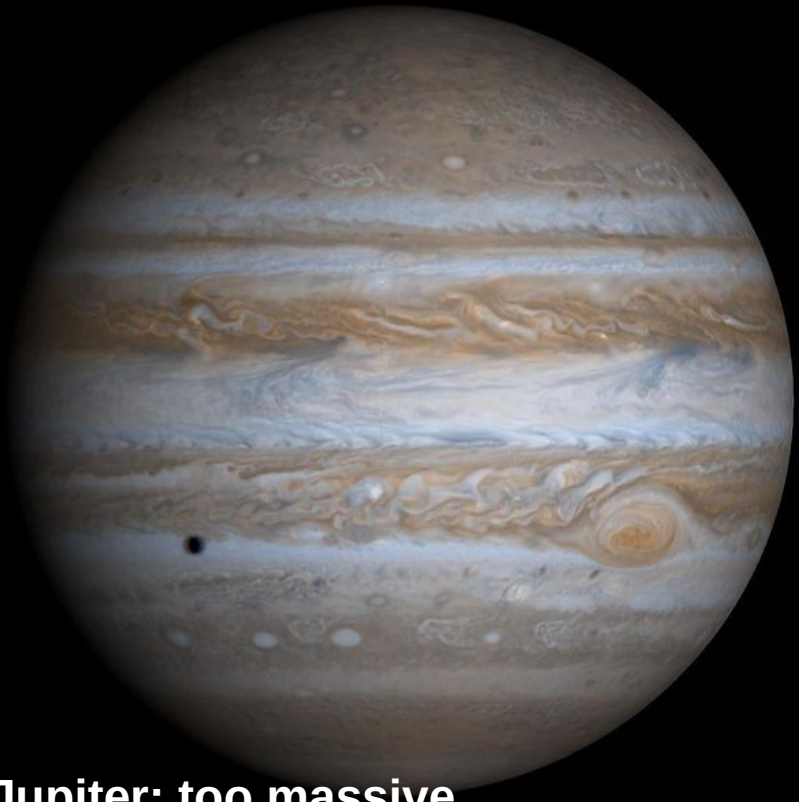
Mars: too far, too cold

What makes planets habitable ?

Size also matters:
not too big, not too small

Moon: too small
Weak gravity can't hold atmosphere
No atmosphere → no life

Earth

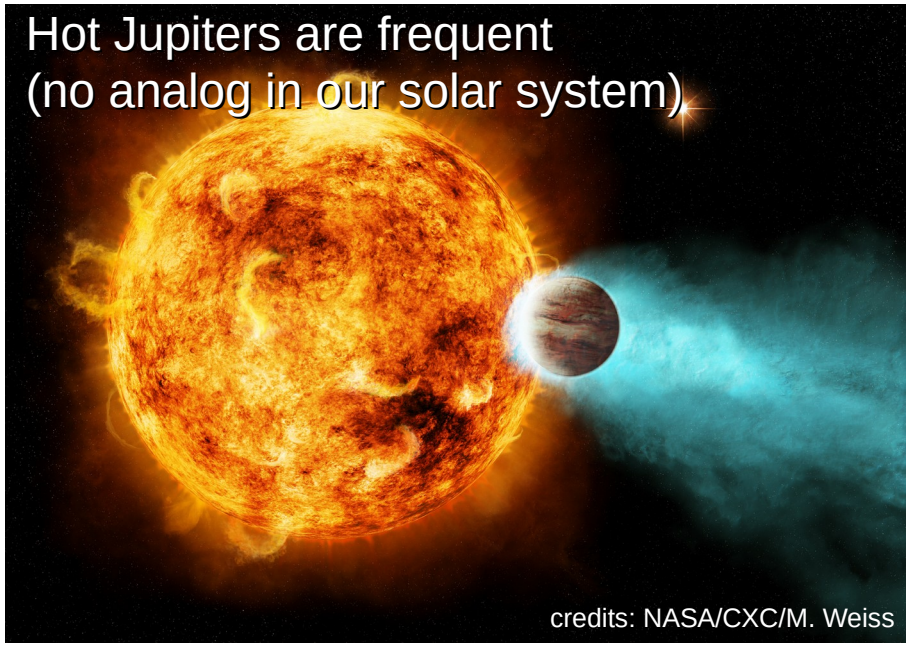


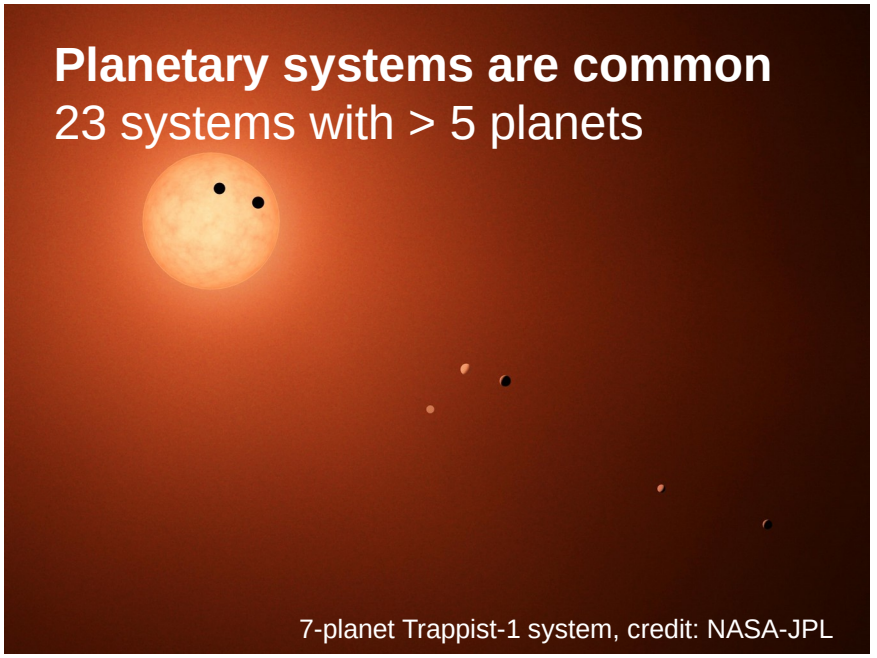
Jupiter: too massive

Gravity holds thick atmosphere of Hydrogen and Helium

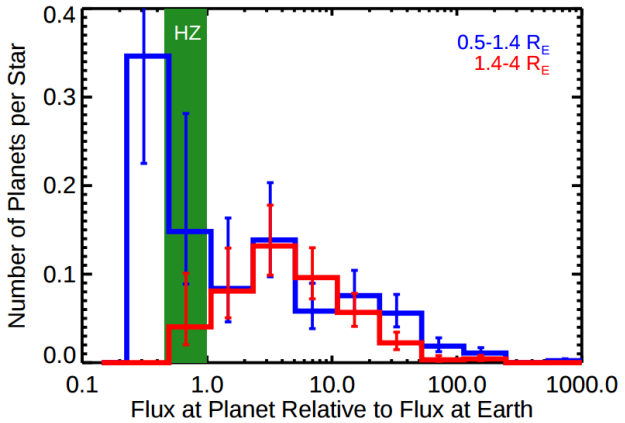
Mostly gas
→ no habitable surface for life to take hold
→ wrong atmosphere composition

Key statistical findings





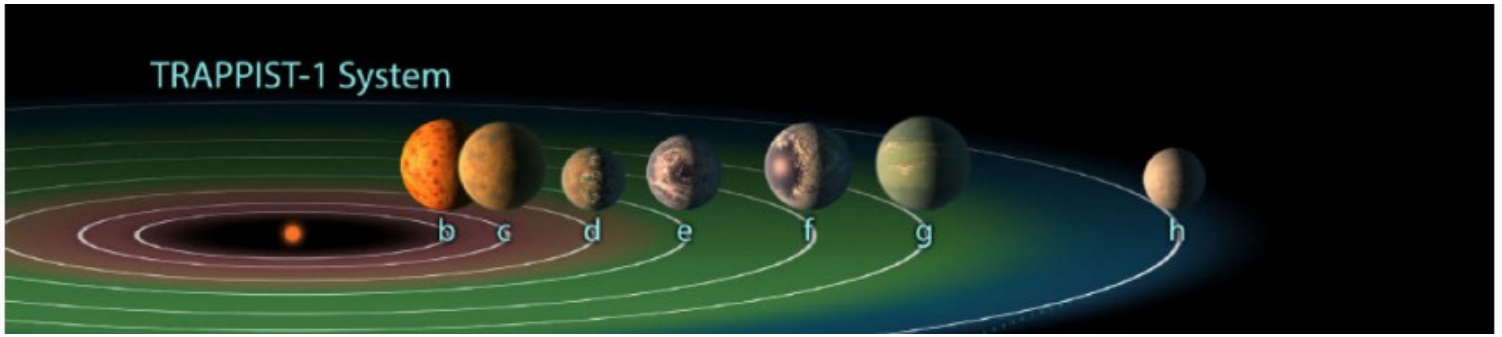
Earth-size rocky planets are abundant



Dressing & Charbonneau 2013

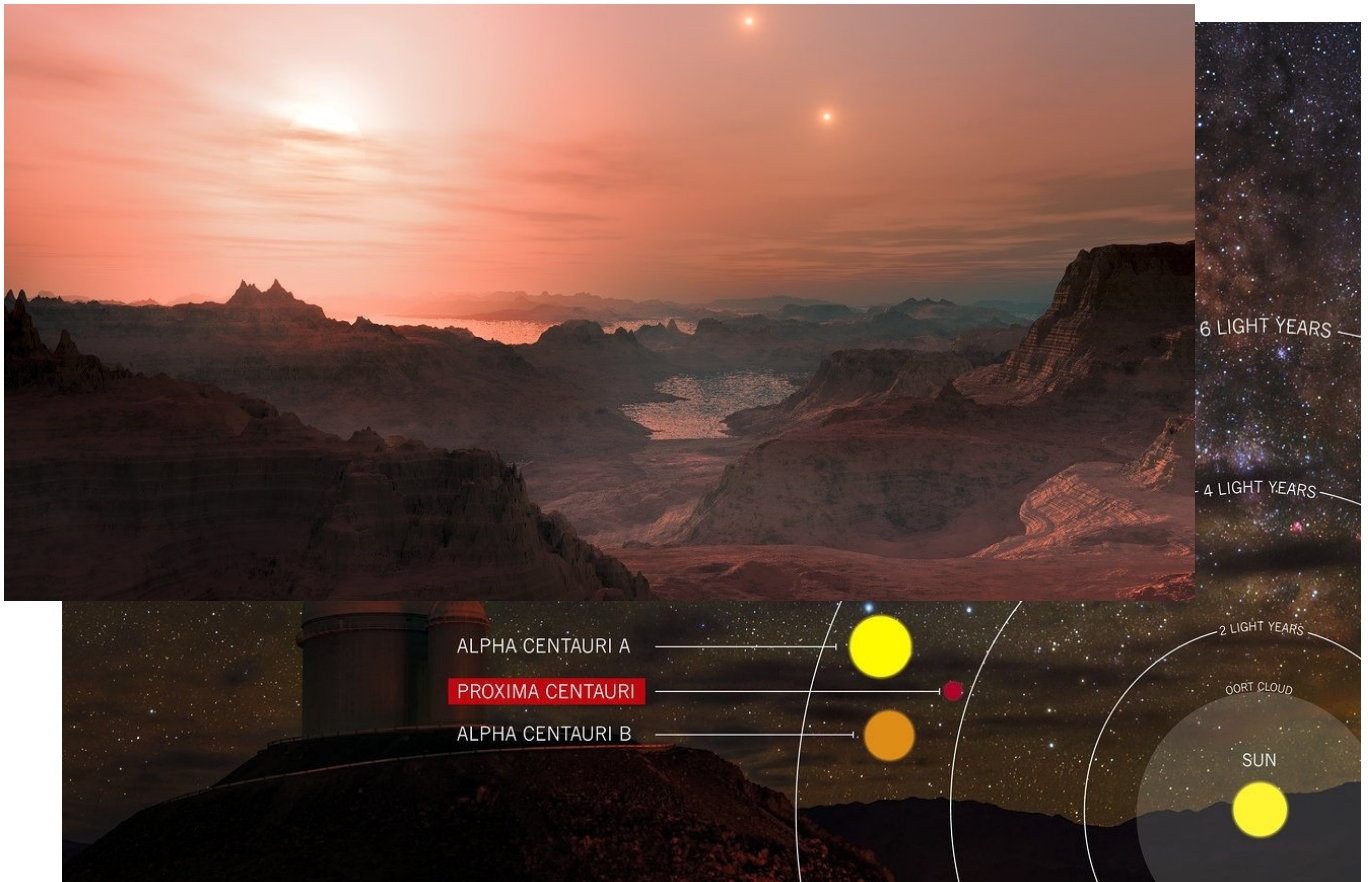
Spectacular recent discoveries around nearby stars

Trappist-1 system
7 planets
~3 in hab zone
likely rocky
40 ly away



Proxima Cen b
planet
Possibly habitable

Closest star to our solar system (only 4.2 light years away)



300 billion stars in our galaxy

→ 30 billion habitable planets

If 100 explorers were sent to visit each habitable for 10 seconds (only 300 million planets/explorer)...

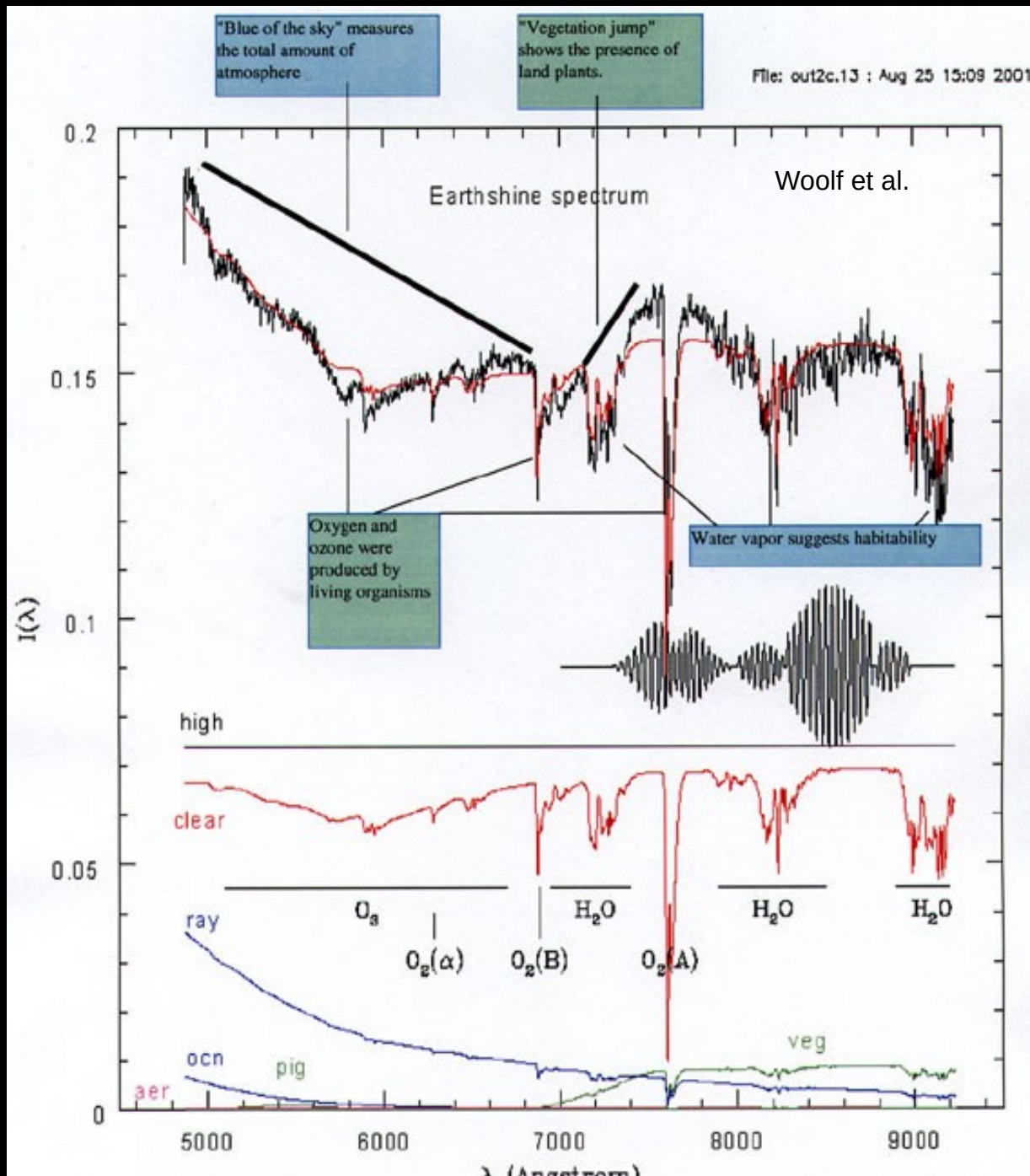
… it would take 95 yrs to complete the habitable exoplanets tour … in our galaxy alone


.. and there are 200 billion galaxies in the observable universe

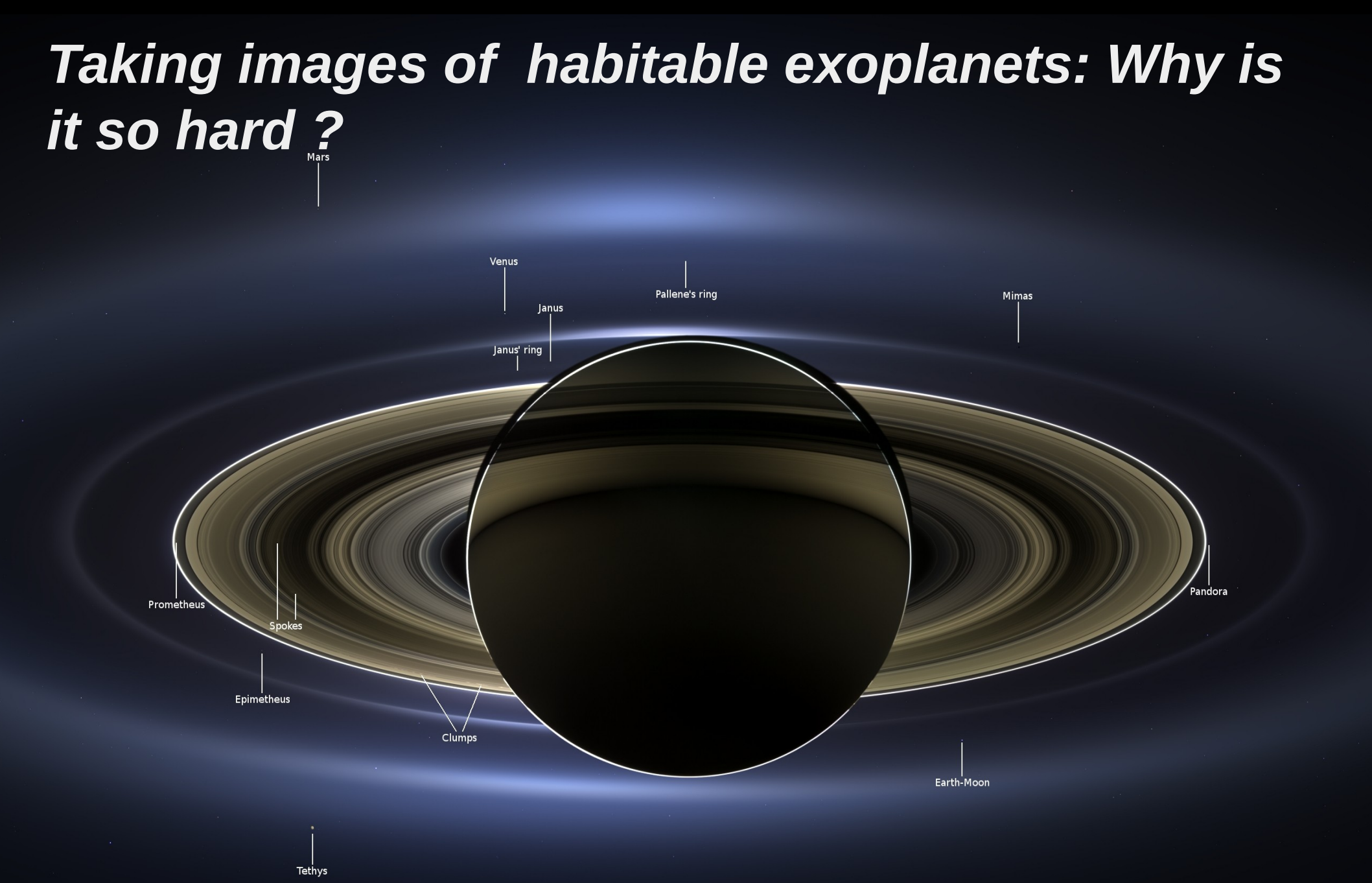
Why should we image planets ?

Imaging allows spectroscopy to measure atmosphere composition

Spectrum of Earth (taken by looking at Earthshine) shows evidence for life and plants



Taking images of habitable exoplanets: Why is it so hard ?



This image was taken by the Cassini spacecraft when it was in Saturn’s shadow… looking back at the inner solar system. Can you spot Earth ?


Saturn
Earth
Earth is 1,000,000,000 x fainter than sun !

Atmospheric Turbulence

Atmosphere Turbulence: Earth's atmosphere introduces strong and fast optical aberrations that blur images

The sun observed with a compact camera

Light rays are bent by atmosphere

→ distortions
→ blurring

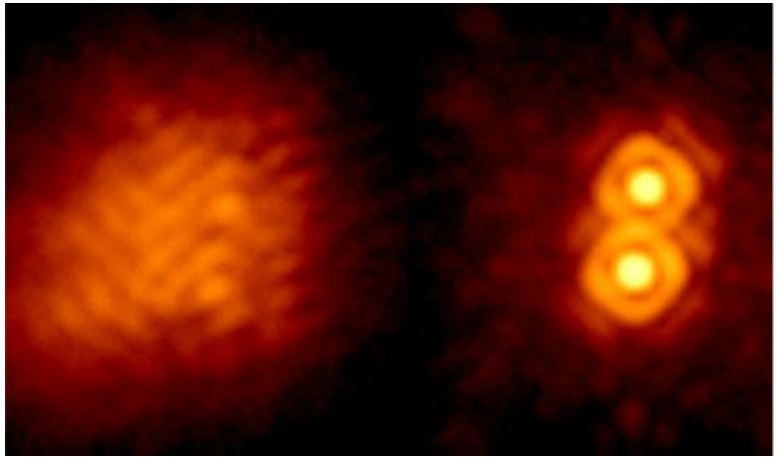
Adaptive Optics (AO)

Atmosphere Turbulence: Earth's atmosphere introduces strong and fast optical aberrations that blur images

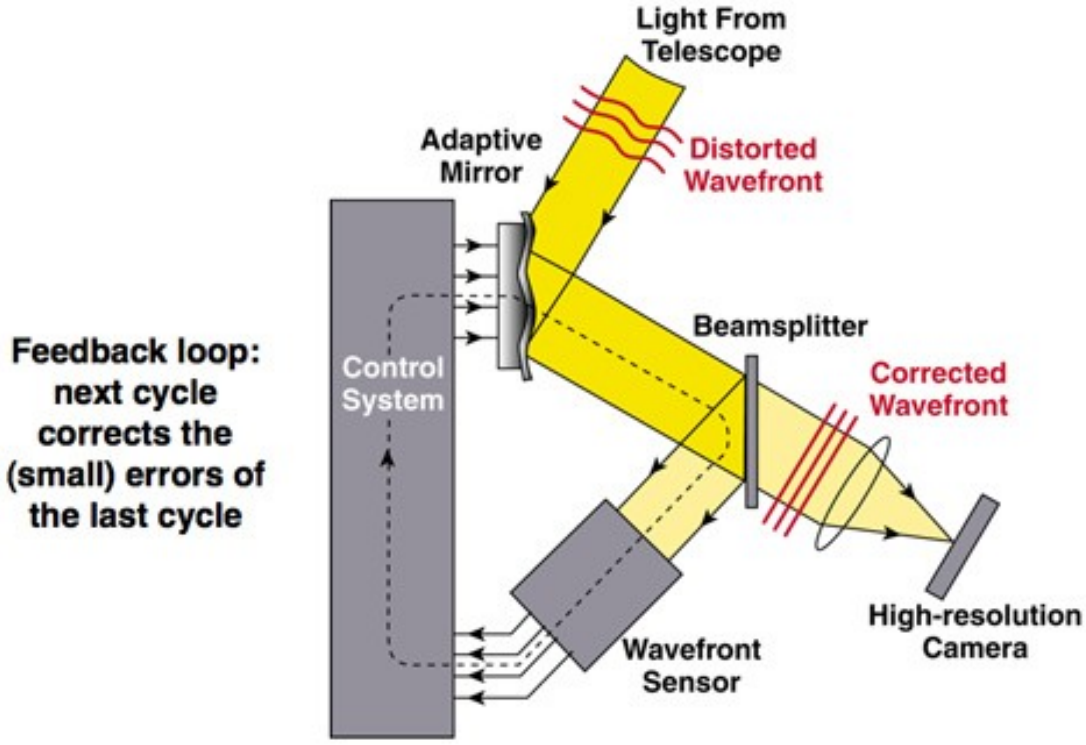
Aberrations must be continuously **measured** and **corrected** to provide sharp images and image exoplanets

Imaging exoplanets is particularly demanding, as the planet is much fainter that the star it orbits: very little room for error !
→ **AO for exoplanet imaging is referred to as Extreme-AO, which is widely recognized <u>as the most challenging application for adaptive optics</u>**

AO OFF AO ON



Palomar obs / NASA JPL



Imaging exoplanets requires 3 techniques to be combined:

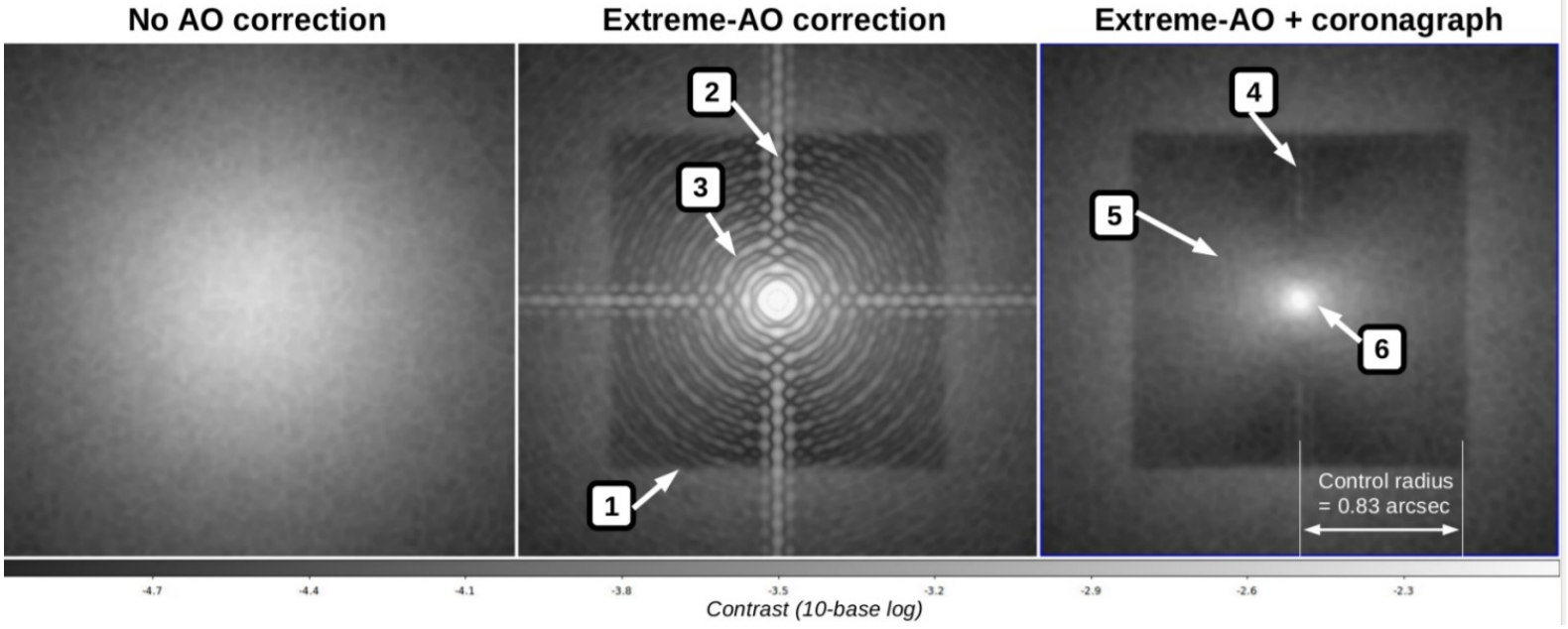
- **Extreme-AO corrects atmospheric turbulence**
- **A coronagraph masks the light of the bright star**
- **Smart image processing to recognize planets**

Our team is deploying GPU-powered AI frameworks to address these challenges

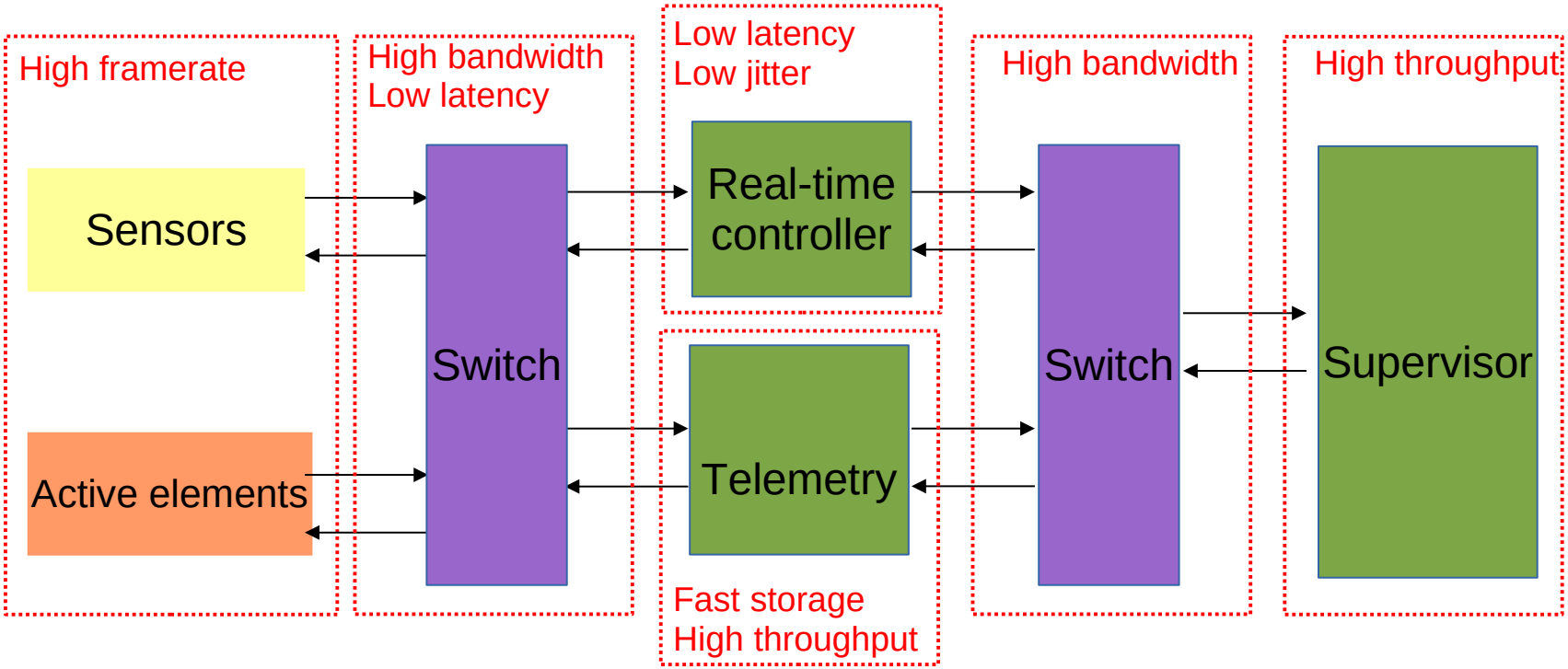
1: ExAO control radius
2: Telescope spider diffraction
3: Diffraction rings
4: Ghost spider diffraction
5: "butterfly" wind effect
6: Coronagraphic leak (low order aberrations)

Monochromatic PSFs, 1.65um
No photon noise
10m/s wind speed, single layer
4ms wavefront control lag

Simulated images below show how Extreme-AO and Coronagraphy deliver high contrast image of a star

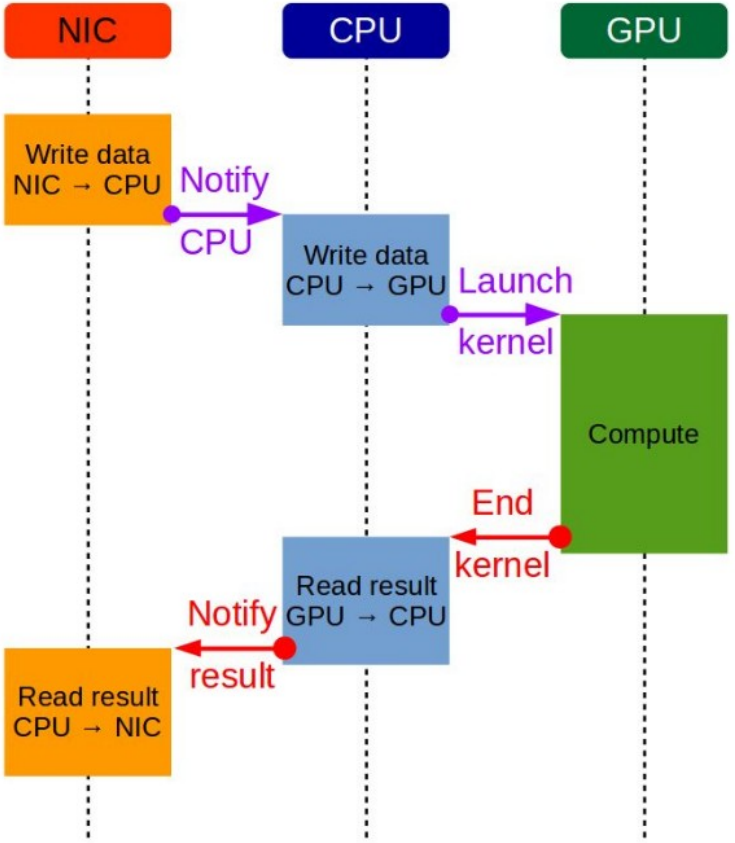


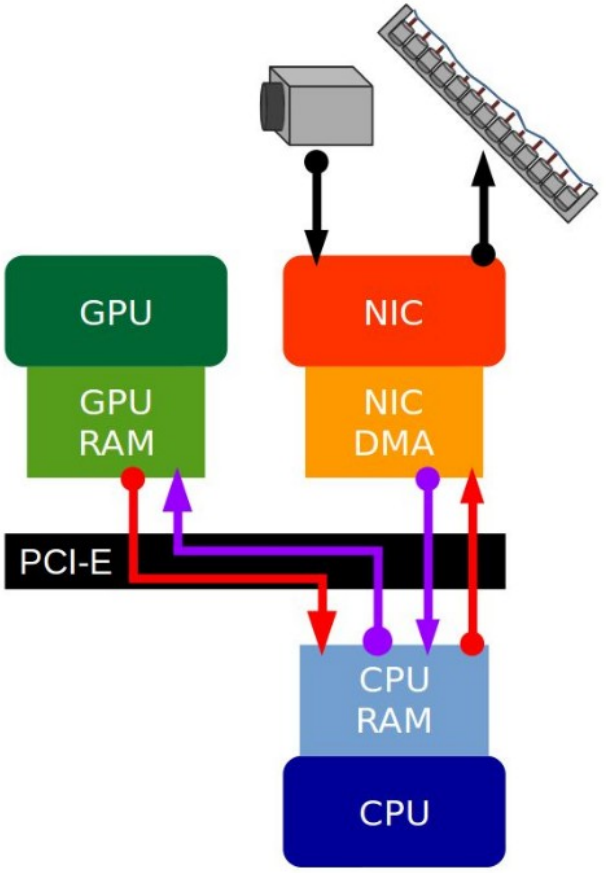
AO Real-time controller (RTC)



Enabling technologies for AO

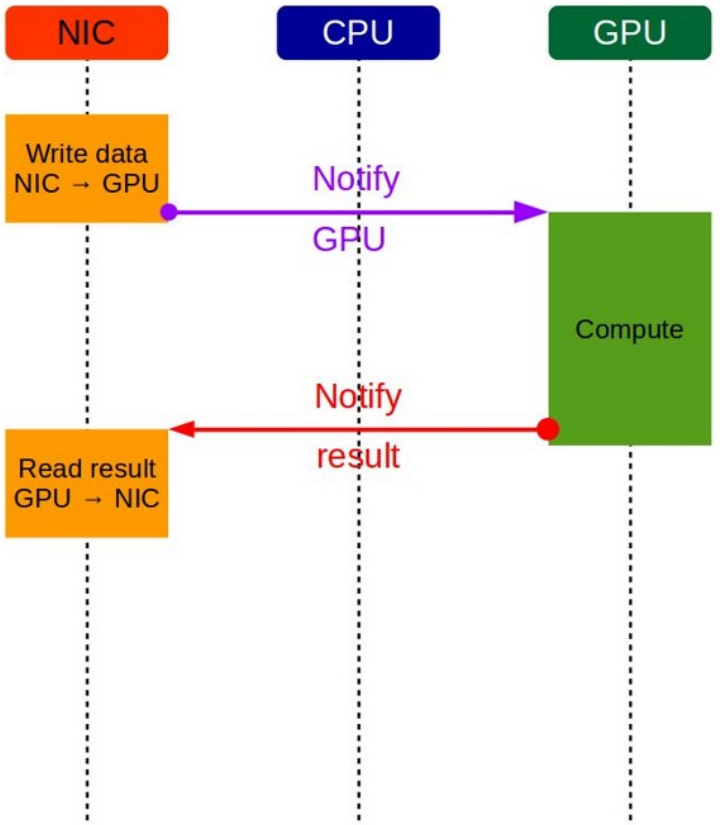
From a standard data acquisition model ...

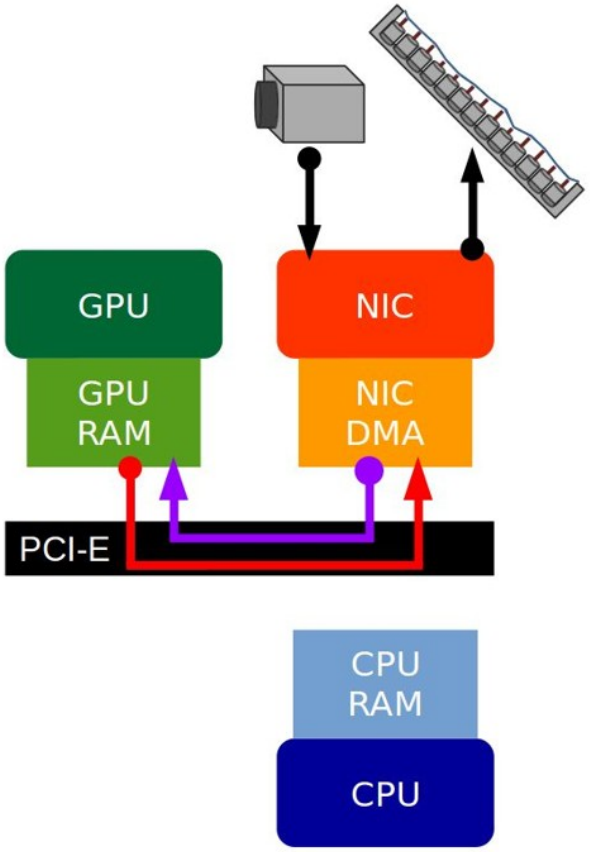




Enabling technologies for AO

… to low latency low jitter data acquisition

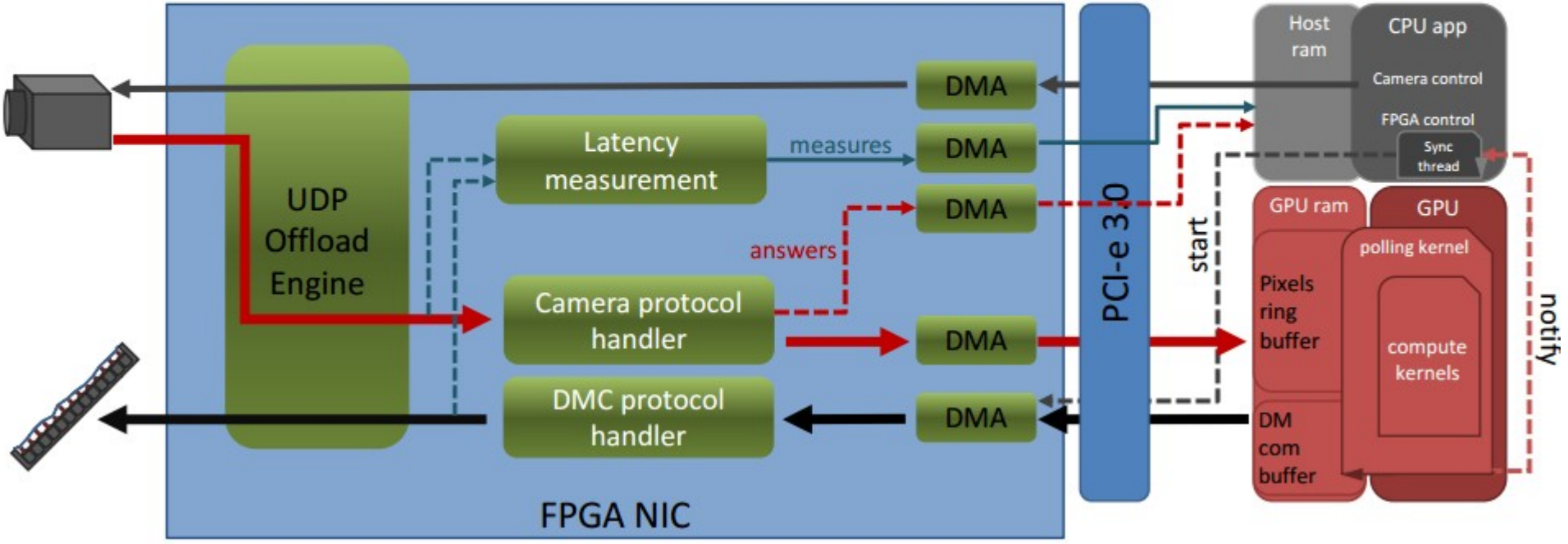




Real-time data acquisition

FPGA writes/reads directly to/from GPU memory

Using only writes would be better though

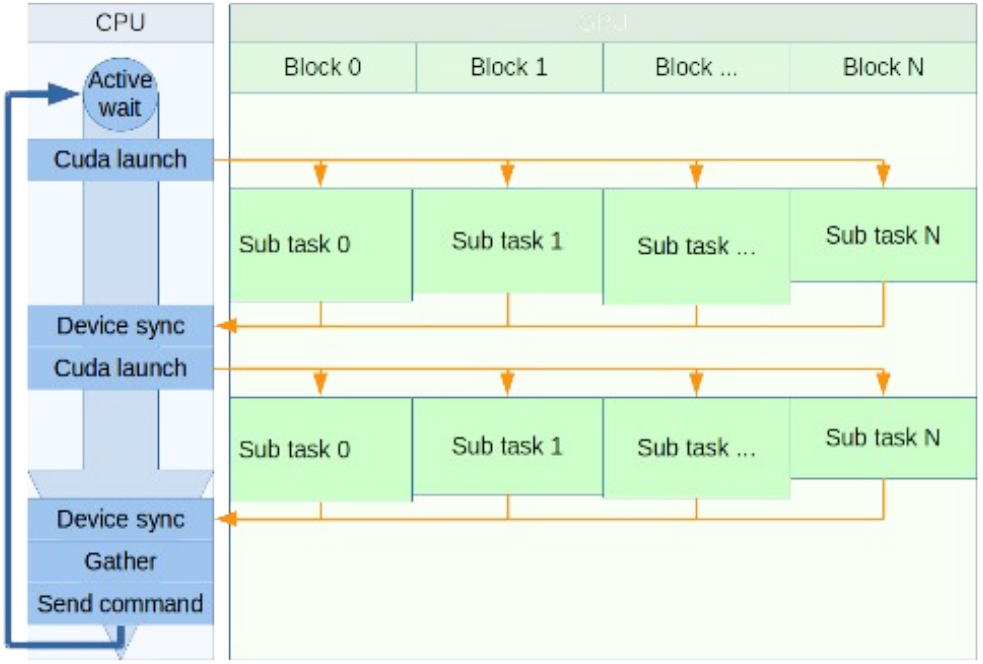


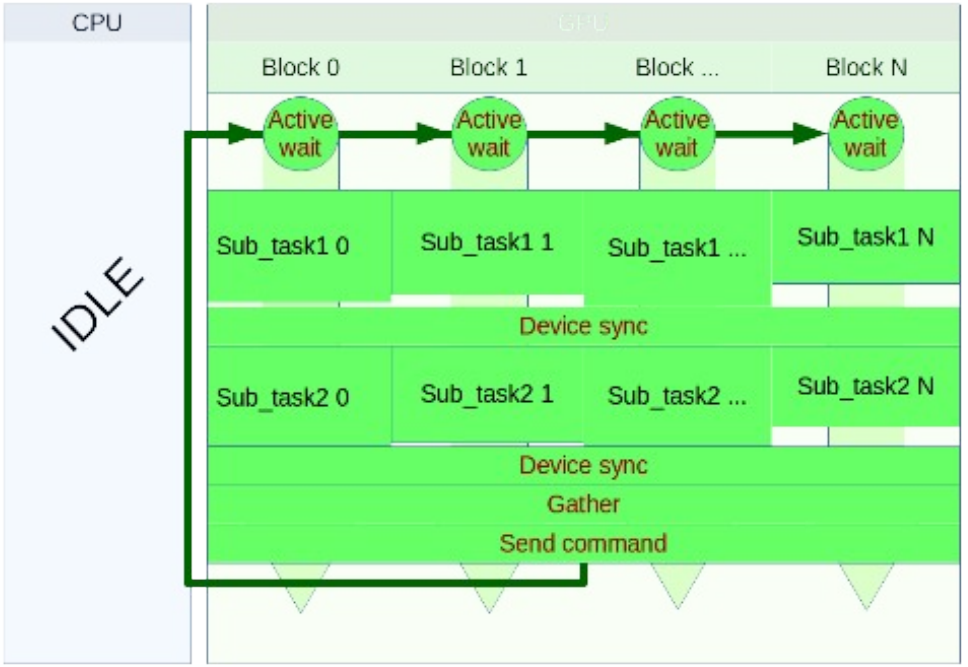
Real-time HPC

Persistent kernels : avoid any communication between GPU kernel and CPU process during execution

Maximize overall performance => towards low latency

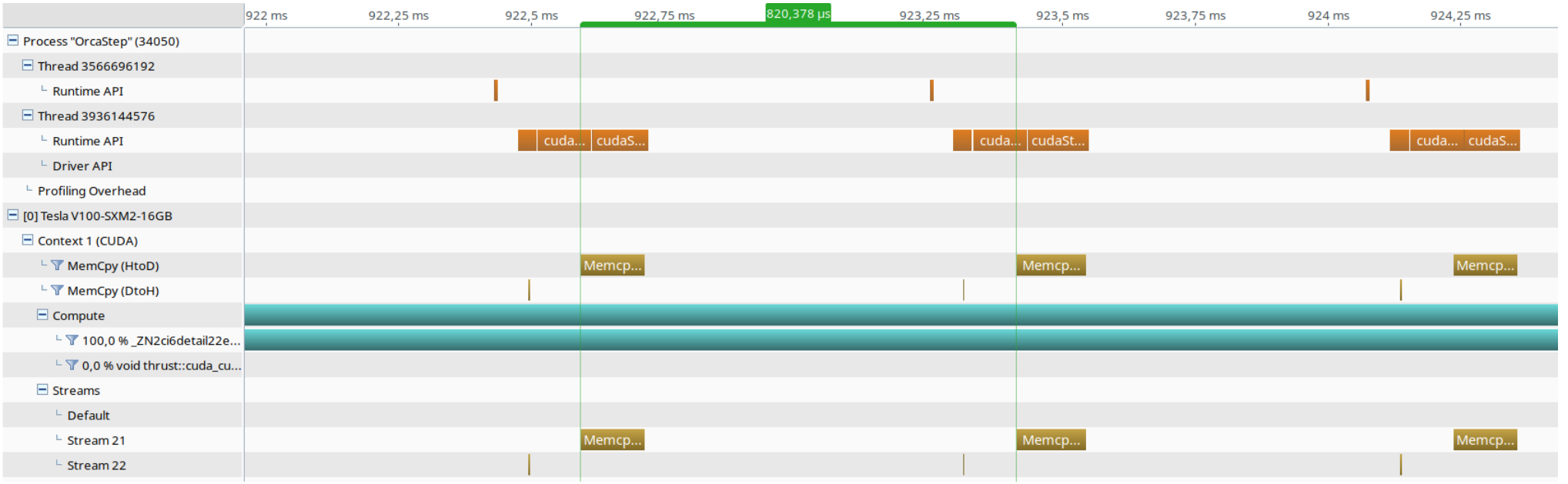
Minimize jitter => towards high level of determinism (real-time computing)





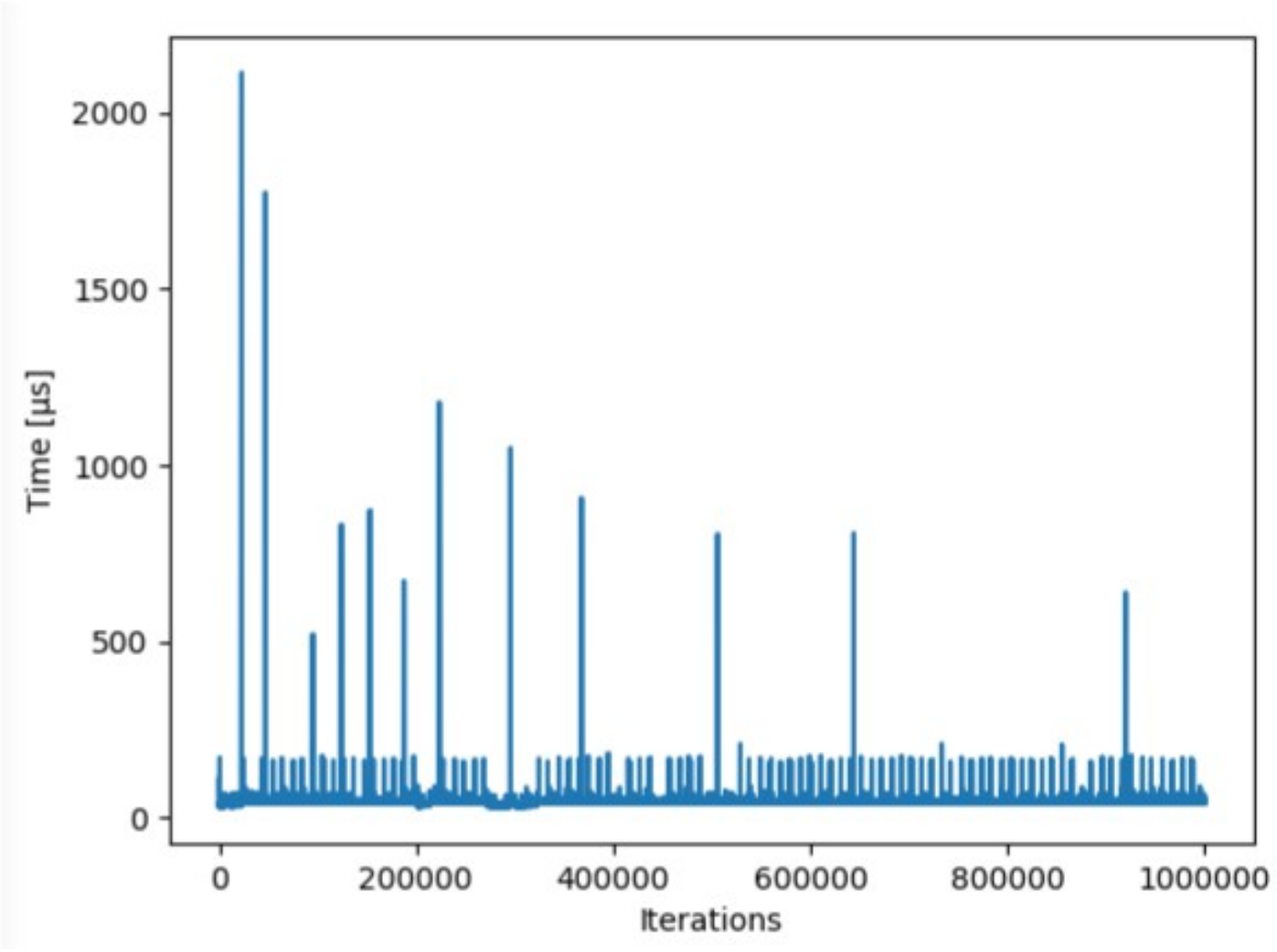
Real-time HPC

Persistent kernels for low jitter



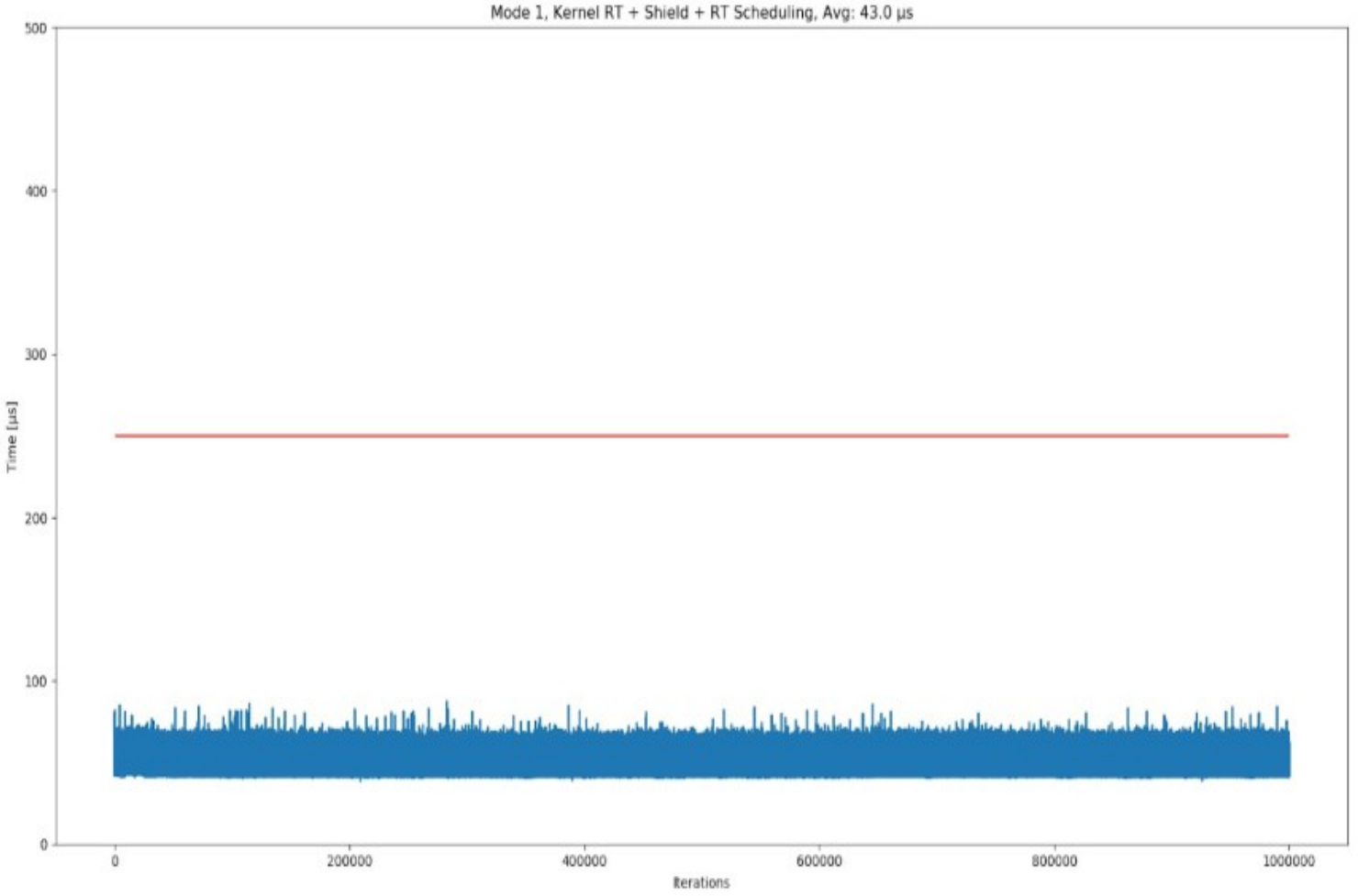
Real-time HPC

Standard execution model (multiple kernel launches)



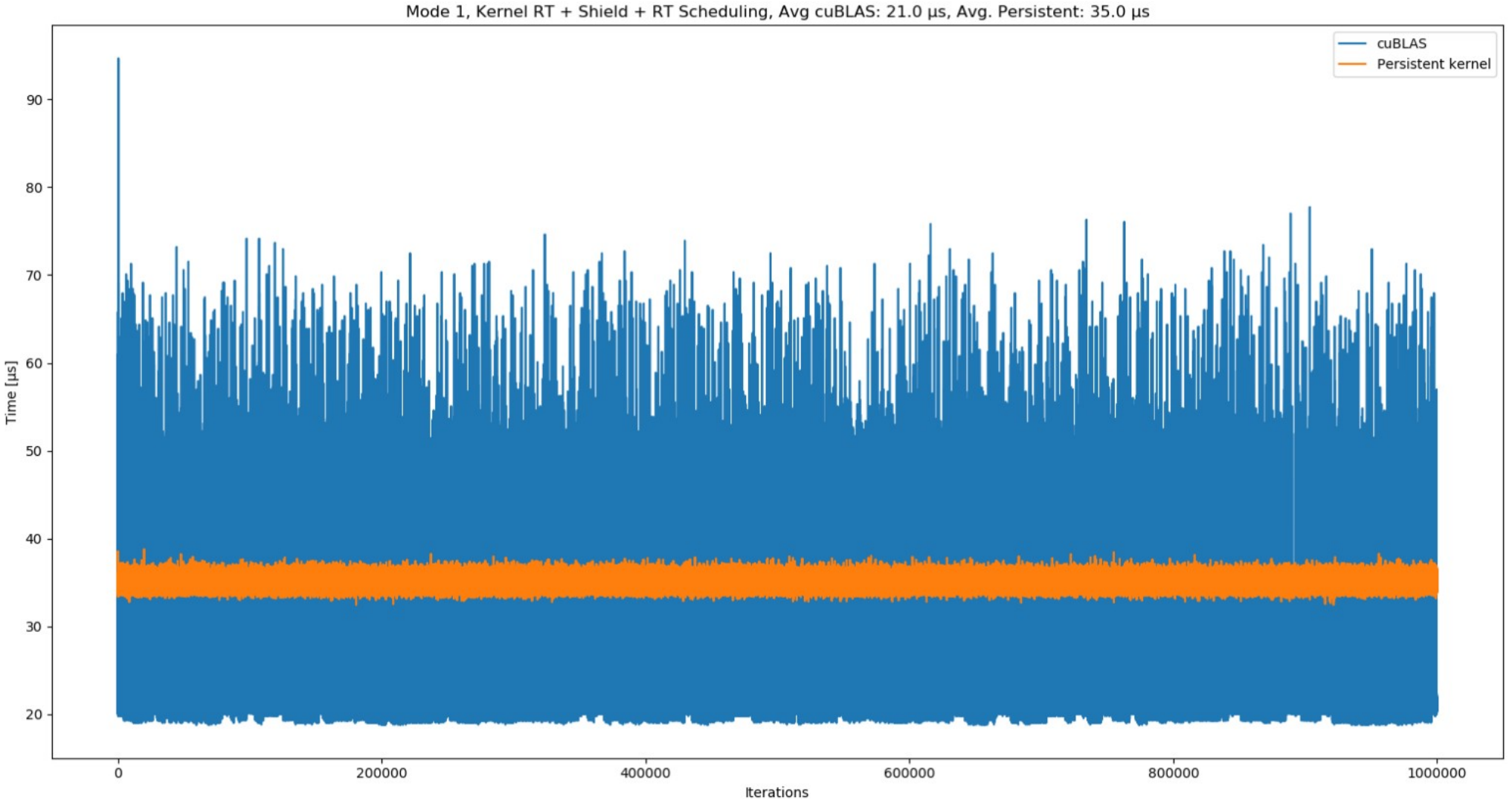
Real-time HPC

Standard execution model + RT patch + process shielding + RT scheduling



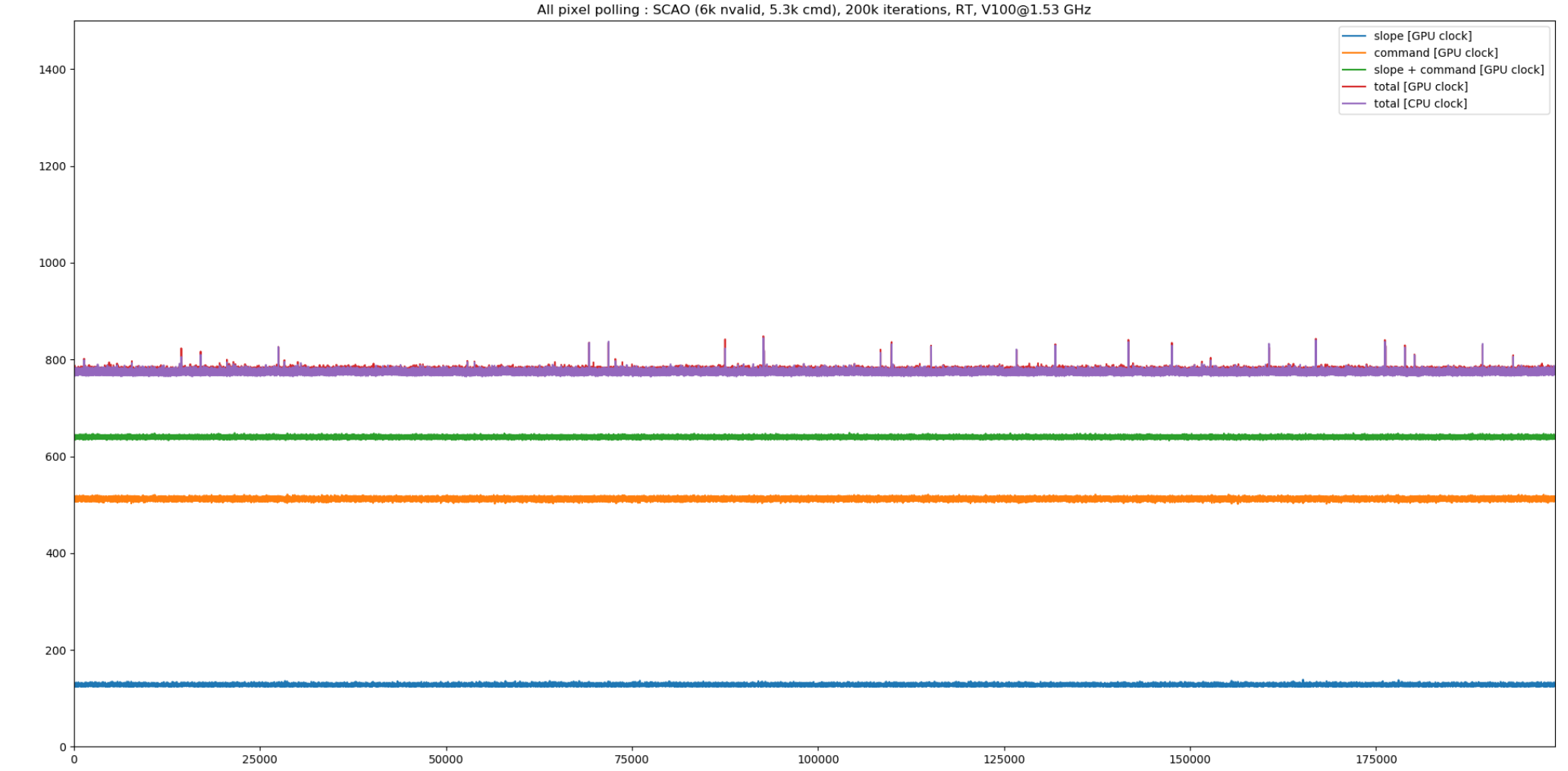
Real-time HPC

Standard execution model versus persistent kernels



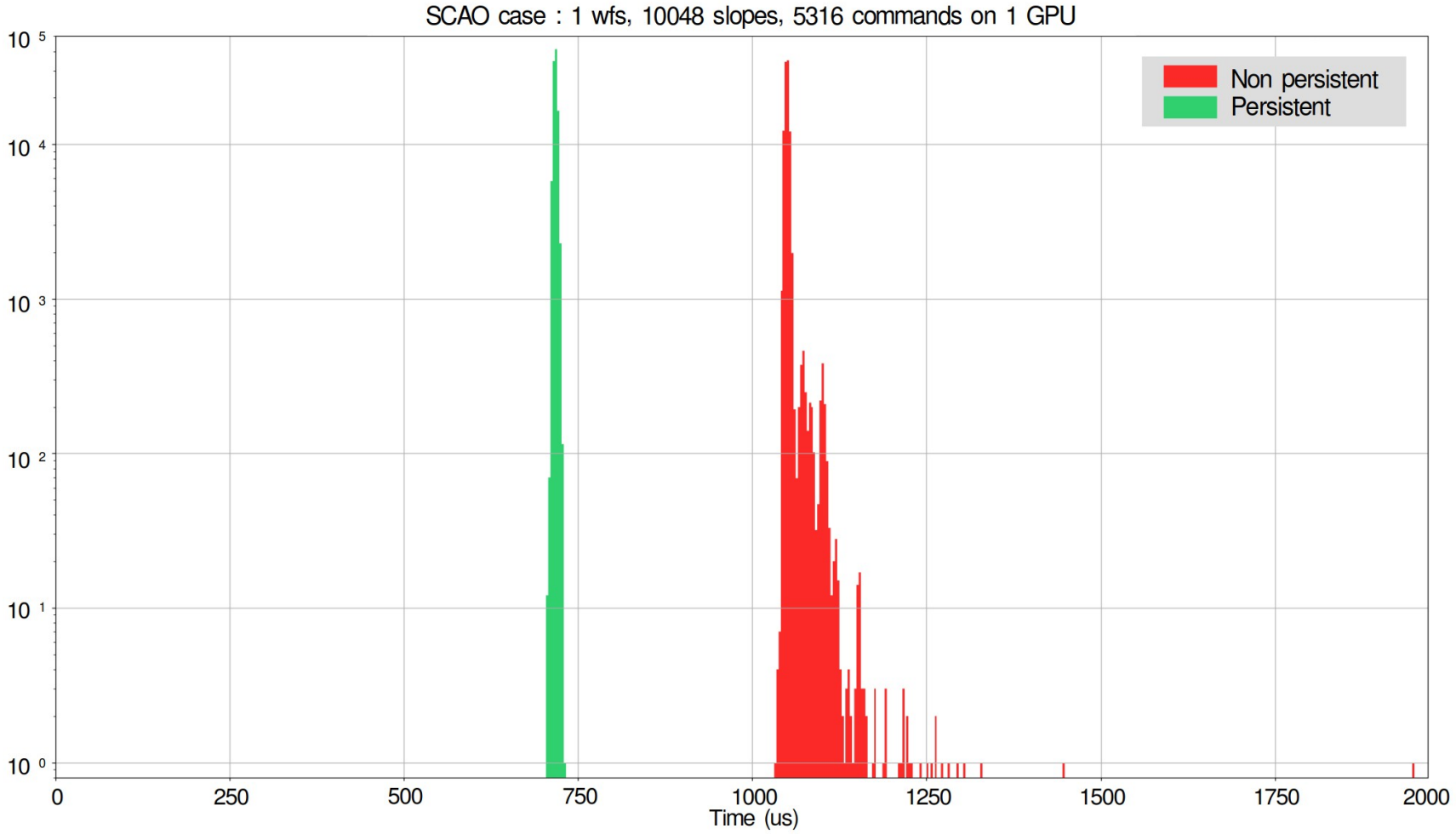
Real-time HPC

Monitoring the pipeline execution with internal profiling tools to avoid profiling through a CPU process (introducing jitter)



Real-time HPC

Standard execution model versus persistent kernels



Real-time HPC

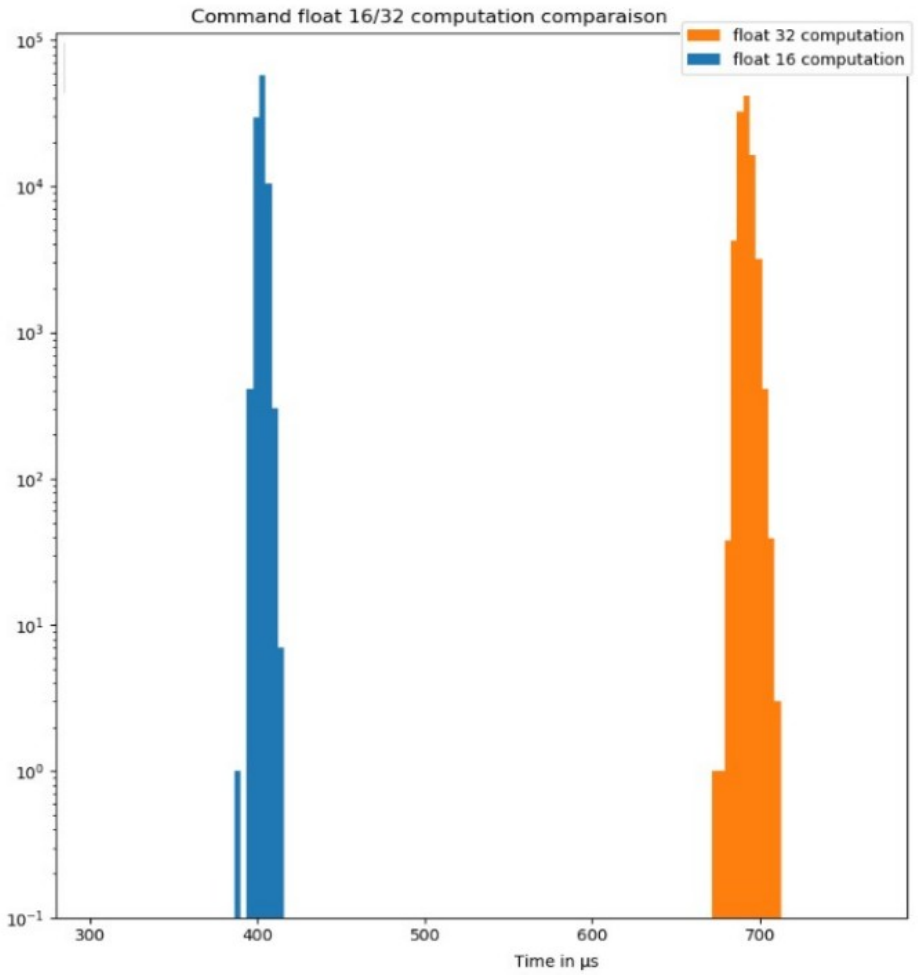
Persistent kernels concept
works as well with 16 bits floats
(a.k.a. half precision)

Input vector needs reformating
(2xFP16 per individual kernel
instance)

Increased performance
(x1.8 faster)

Same level of determinism

Accuracy compatible with AO
application (verified with
end-to-end AO simulator)

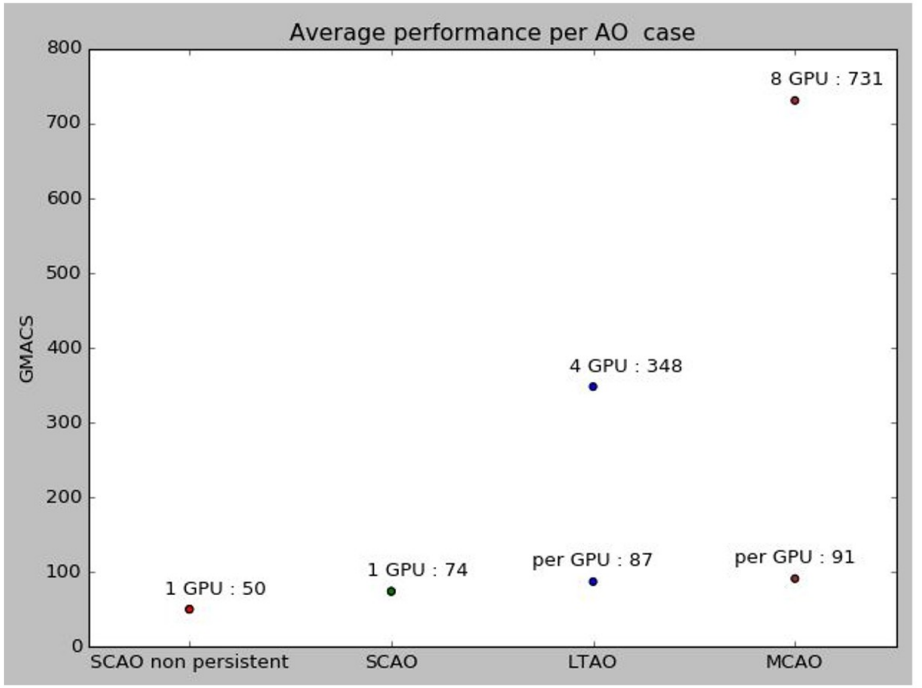


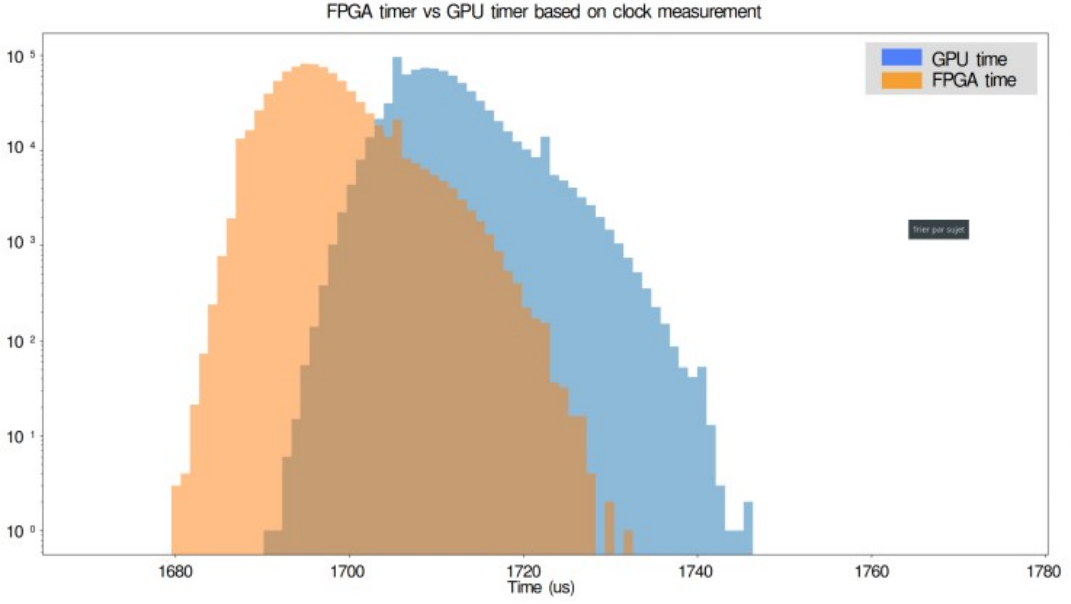
Real-time HPC

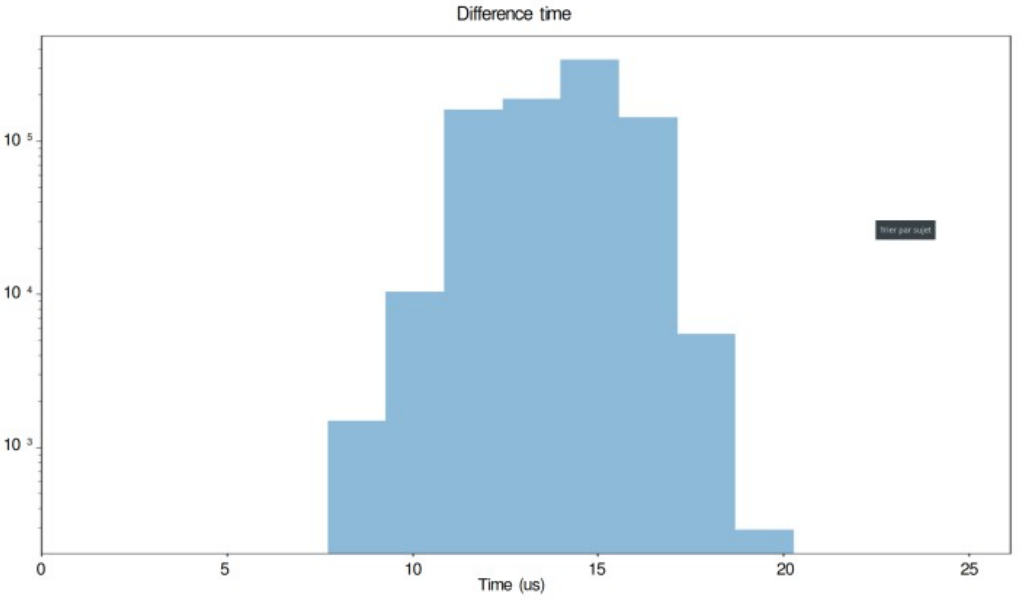
Multi-GPU scalability on
NVIDIA DGX-1

We are counting MAC/s
(memory bound application)

Determinism checked with timing
from FPGA interface

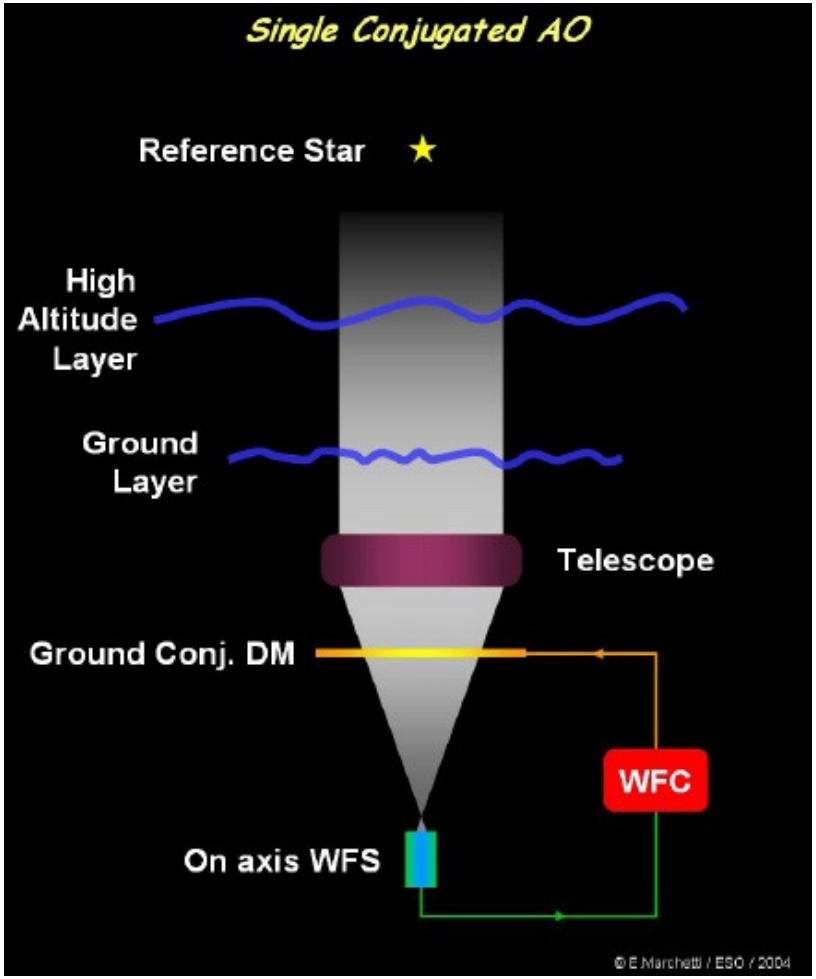


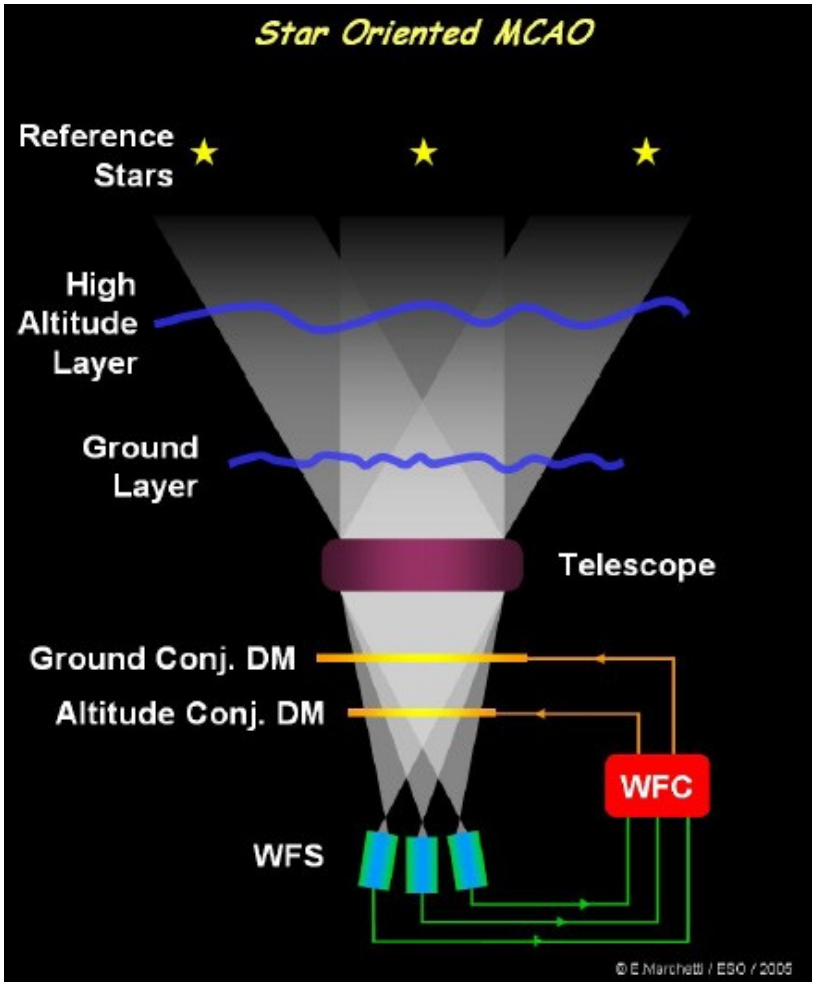




The case for tomography

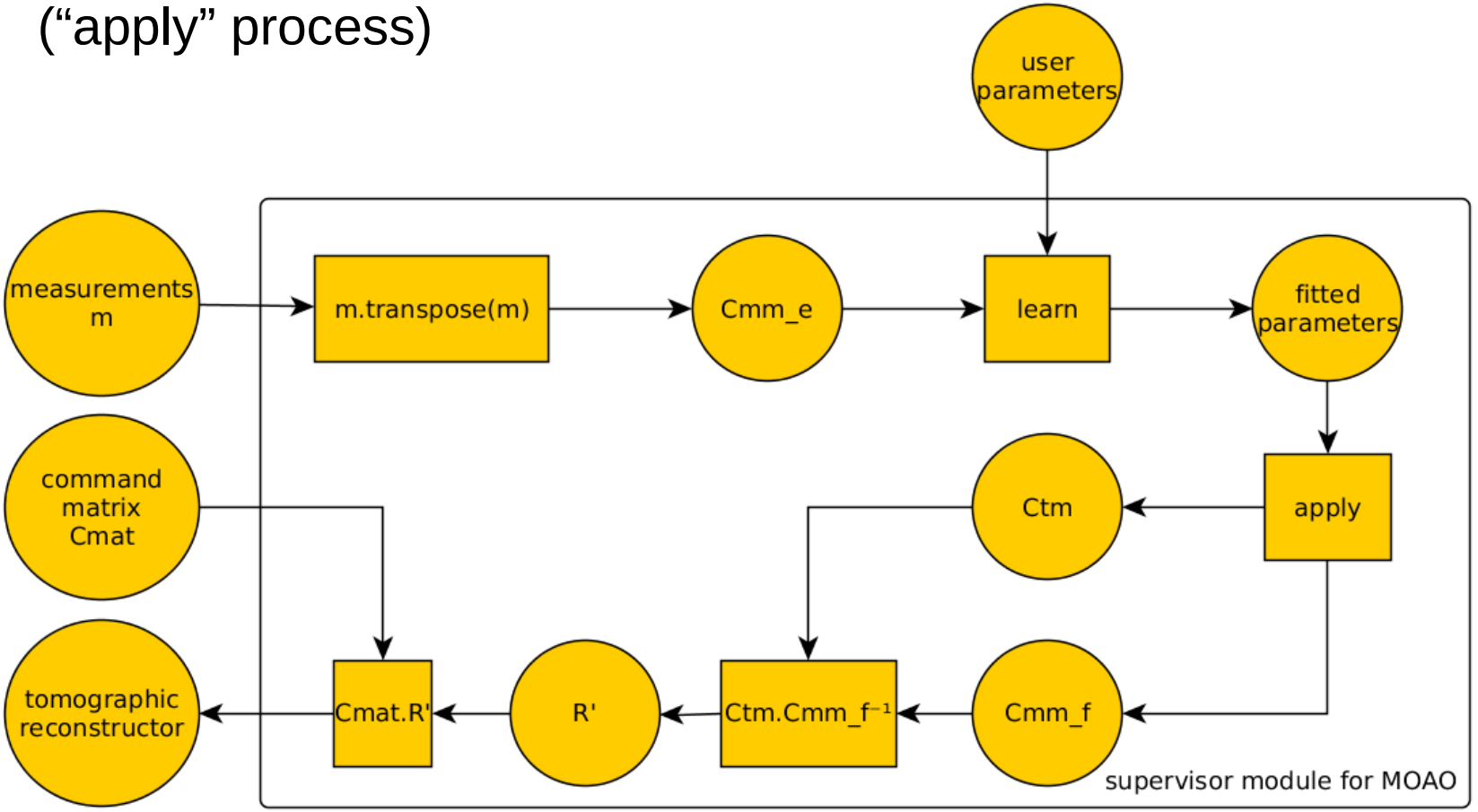
Multiple guide stars (Laser), multiple deformable mirrors





Loop supervision module

Mix of cost function optimization for parameters identification (“Learn” process) and linear algebra for reconstructor matrix computation (“apply” process)



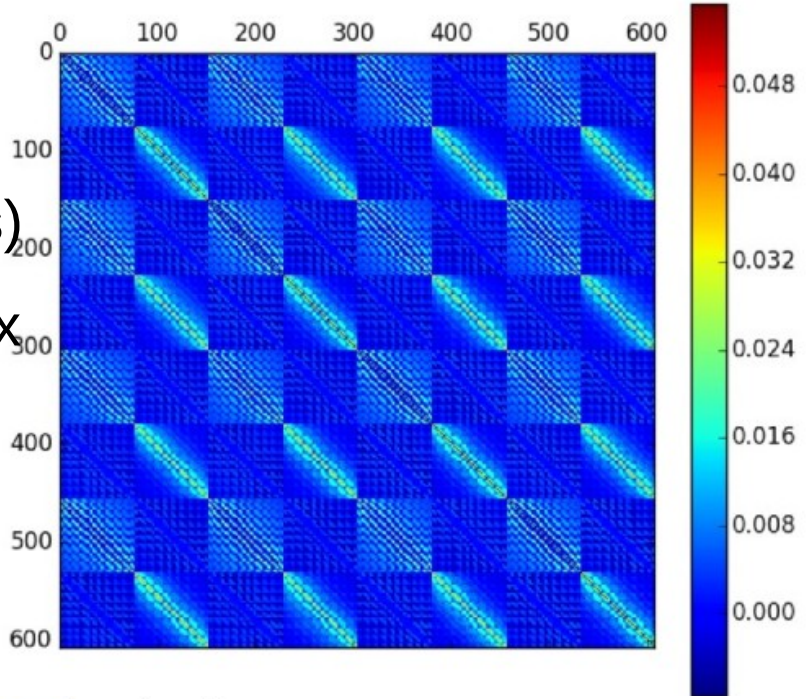
Loop supervision module

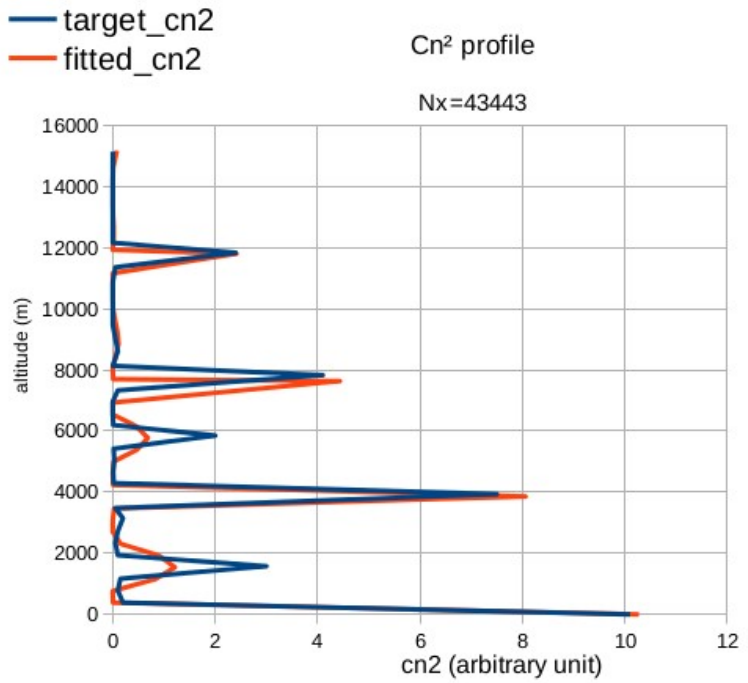
Parameters identification (“Learn” process)

- Fitting measurements covariance matrix on a model including system and turbulence parameters
- Using a score function

$$F(x) = \sum_{k=1}^{N^2} [Cmm_k - f_k(x)]^2$$

- Levenberg-Marquardt algorithm for function optimization
- Exemple of turbulence profile reconstruction
- Dual stage process (5 layers + 40 layer





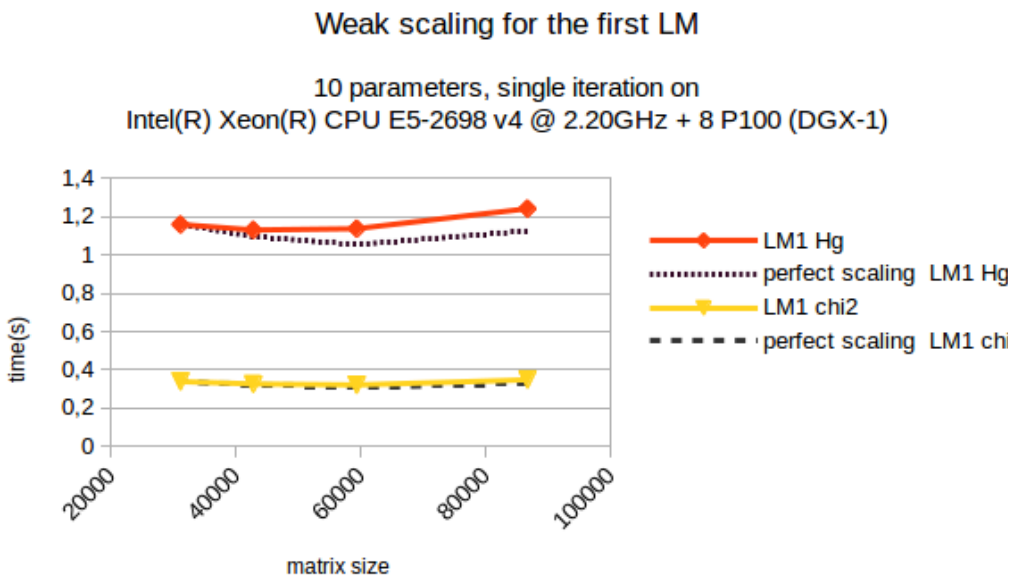
Loop supervision module

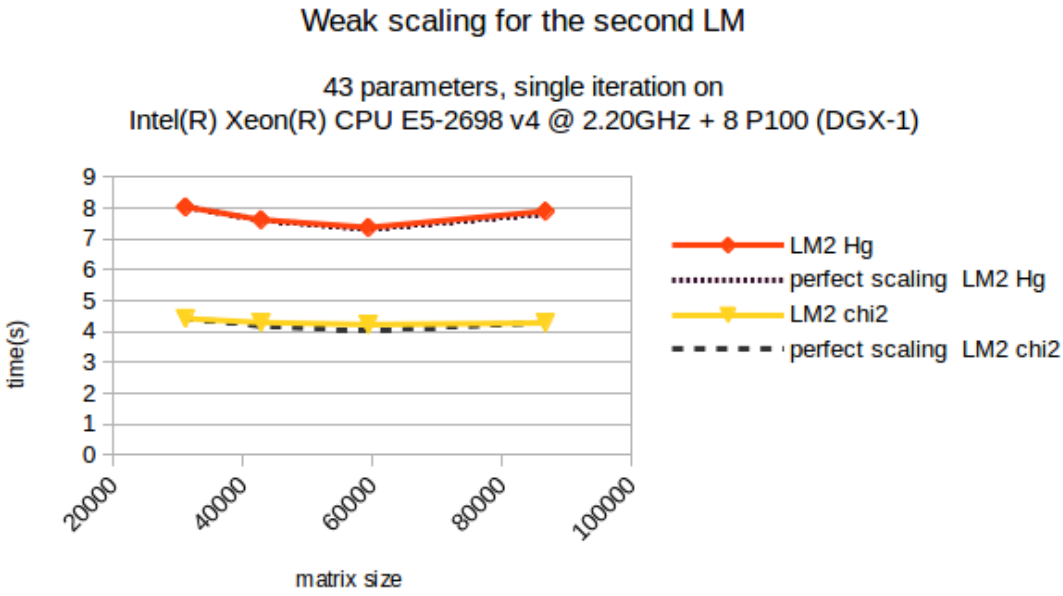
Performance for parameters identification (“Learn” process)

Multi-GPU process, including matrix generation and LM fit

Time to solution for a matrix size of 86k : 240s (4 minutes)

- first pass (5 layers) : 25s
- Second pass (40 layers) : 213s





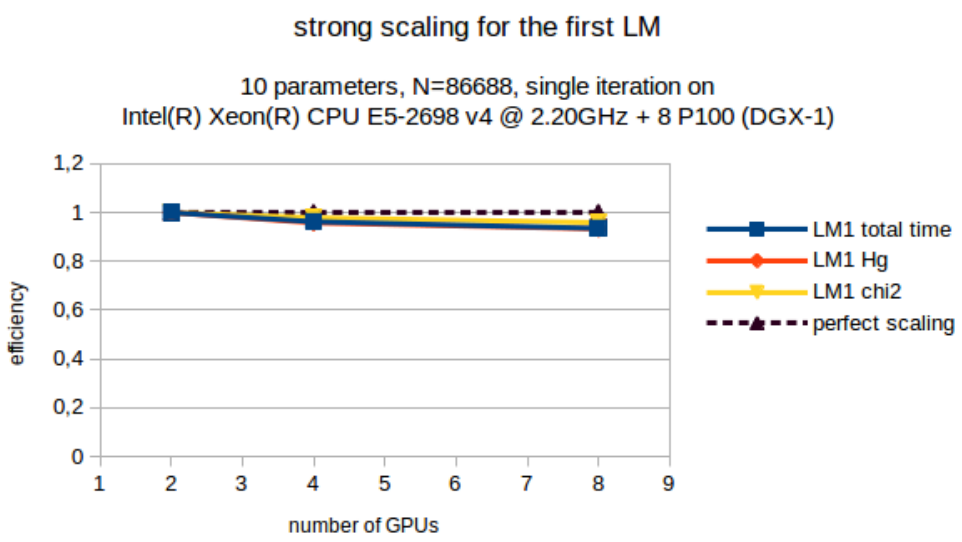
Loop supervision module

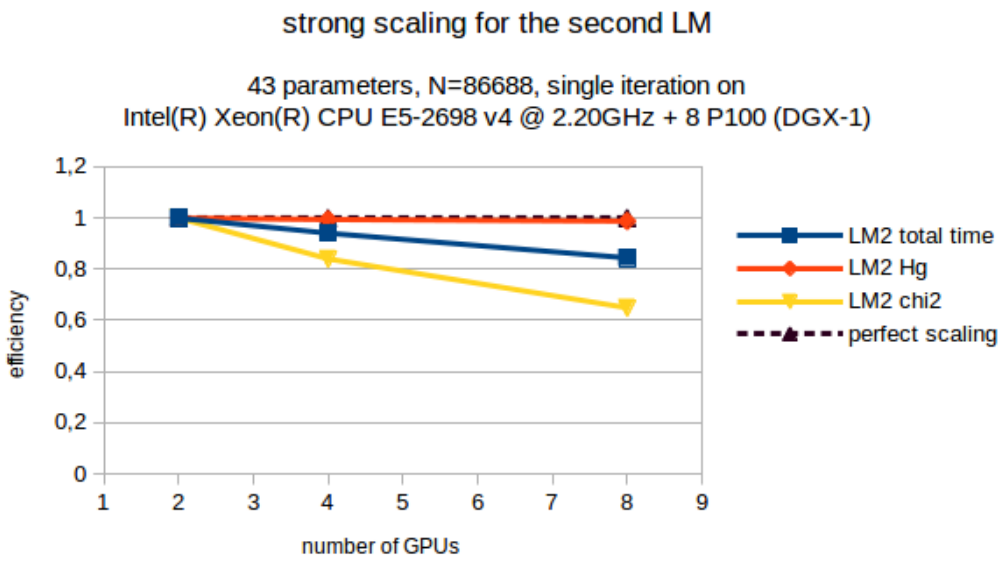
Performance for parameters identification (“Learn” process)

Multi-GPU process, including matrix generation and LM fit

Time to solution for a matrix size of 86k : 240s (4 minutes)

- first pass (5 layers) : 25s
- Second pass (40 layers) : 213s





Loop supervision module

Reconstructor matrix computation (“apply” process)

- Compute the tomographic reconstructor matrix using covarince matrix between “truth” sensor and other WFS and invert of measurements covariance matrix

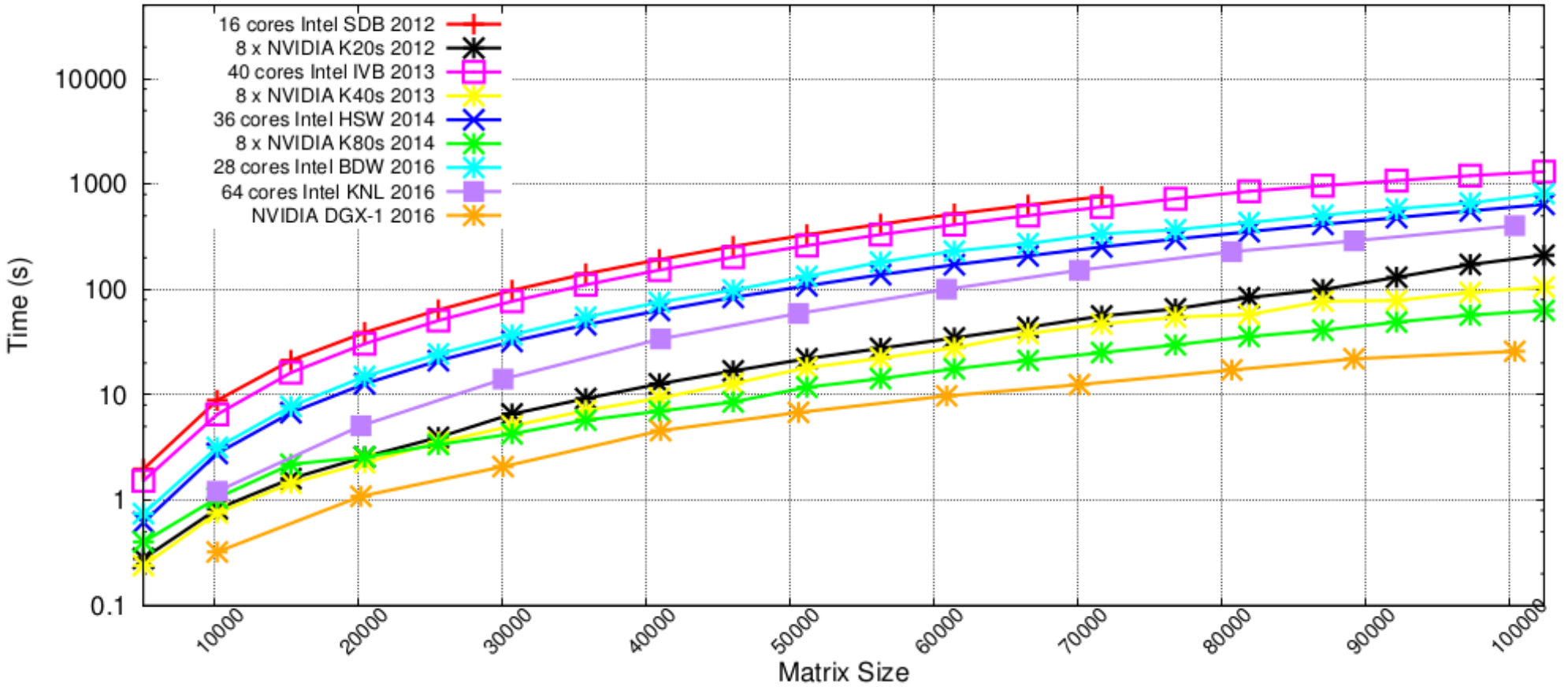
$$R' = Ctm \cdot Cmm_f^{-1}$$

- Can use various methods. “Brute” force : direct solver
- Standard Lapack routine : “posv” : mostly compute-bound, high level of scalability
- Highly portable code : explore various architectures by using standard vendor provided maths libraries

Loop supervision module

Performance evolution over time on different platforms

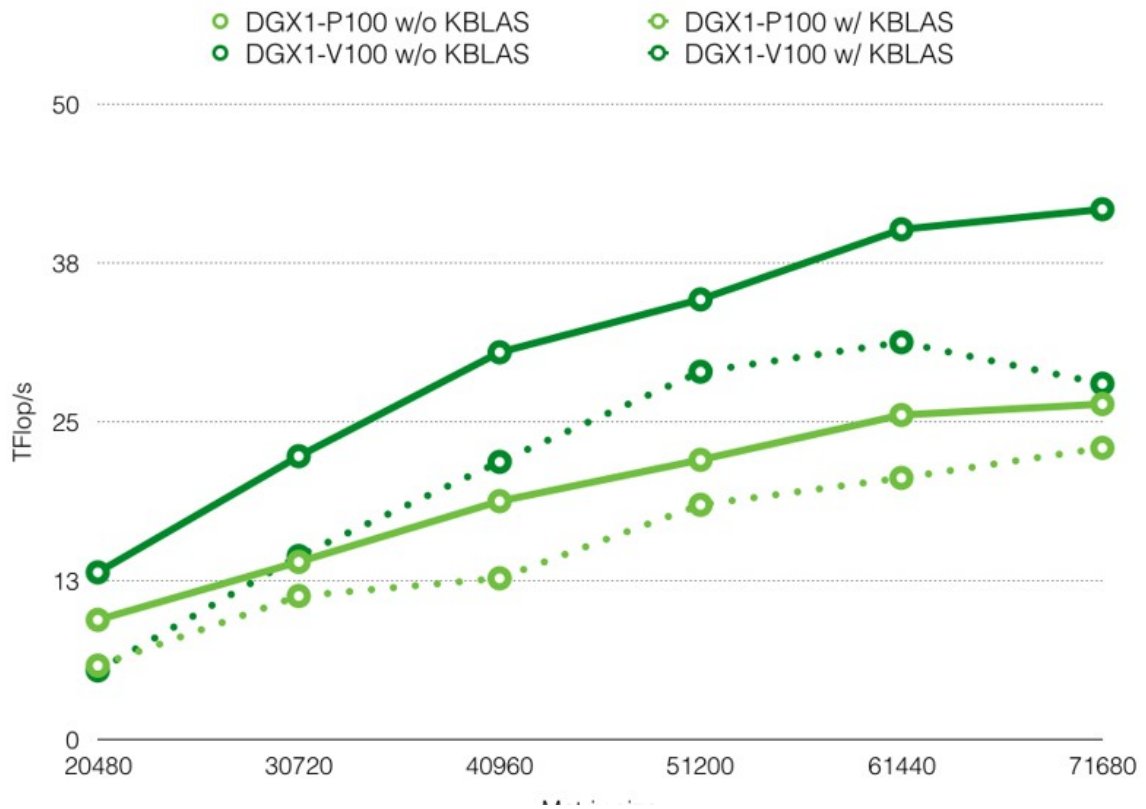
- Comparing generations of GPU and CPUs (+Xeon Phi)



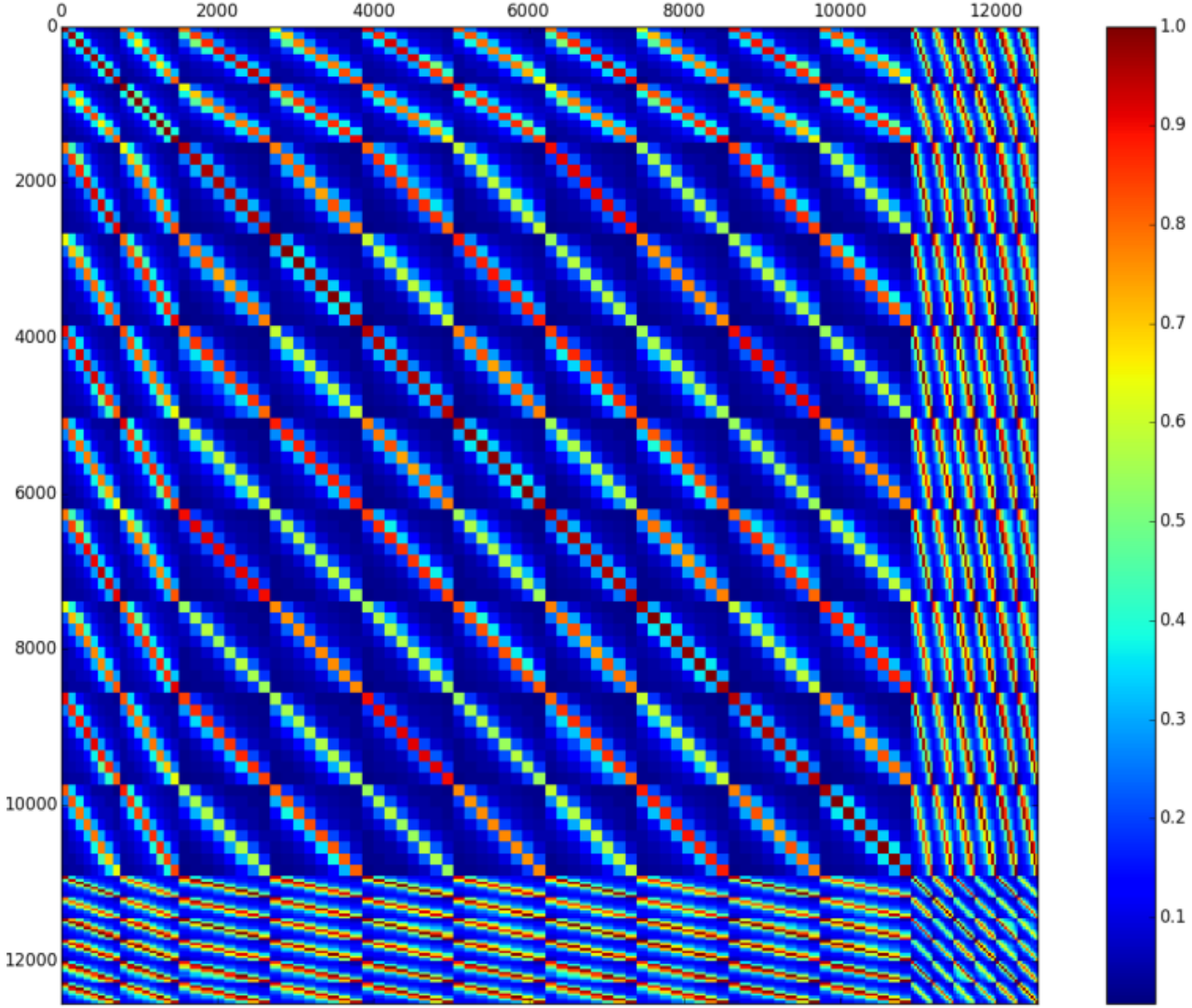
Loop supervision module

State of the art performance on NVIDIA DGX-1 with V100

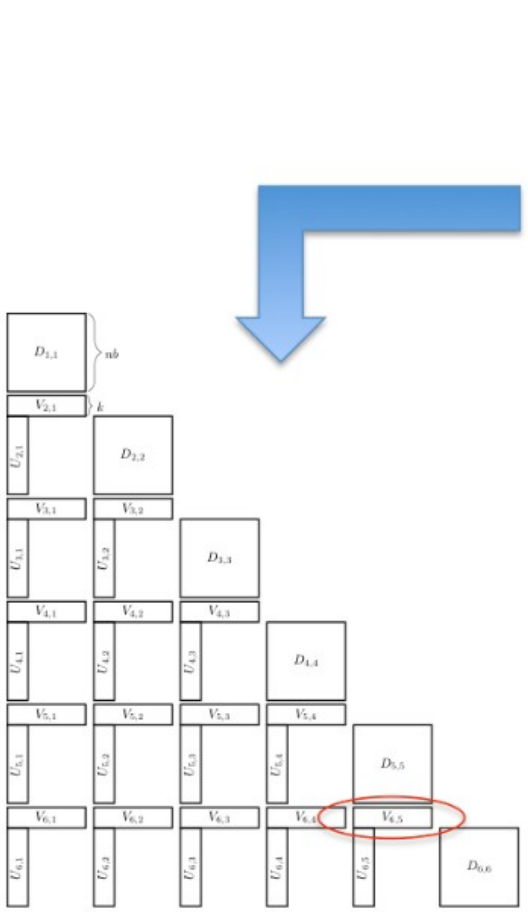
- Versus P100 using BLAS library from KAUST: x1.6

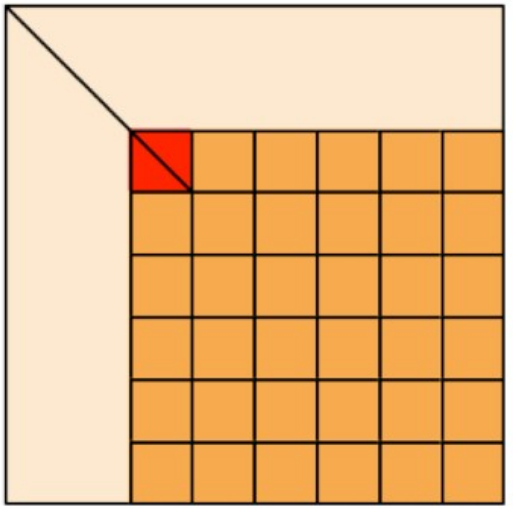


Covariance matrix is data sparse



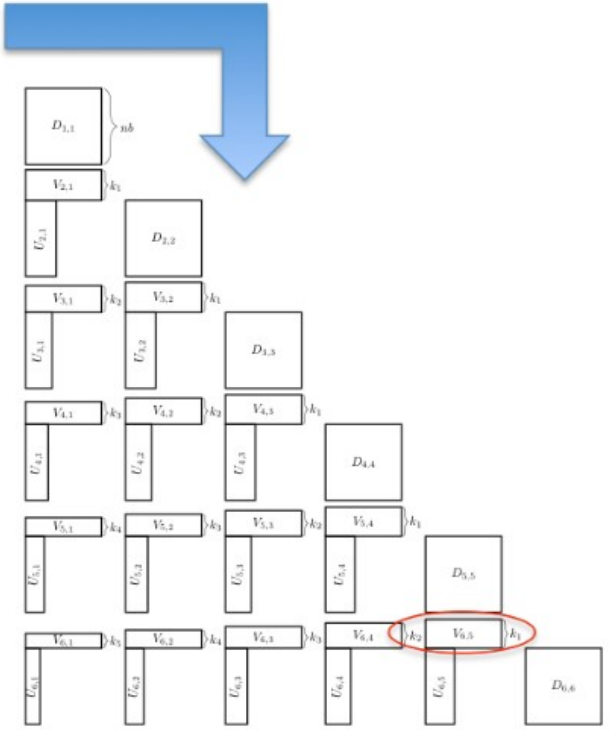
Experimenting with low rank





dense tiles
Cholesky: O(n^3)

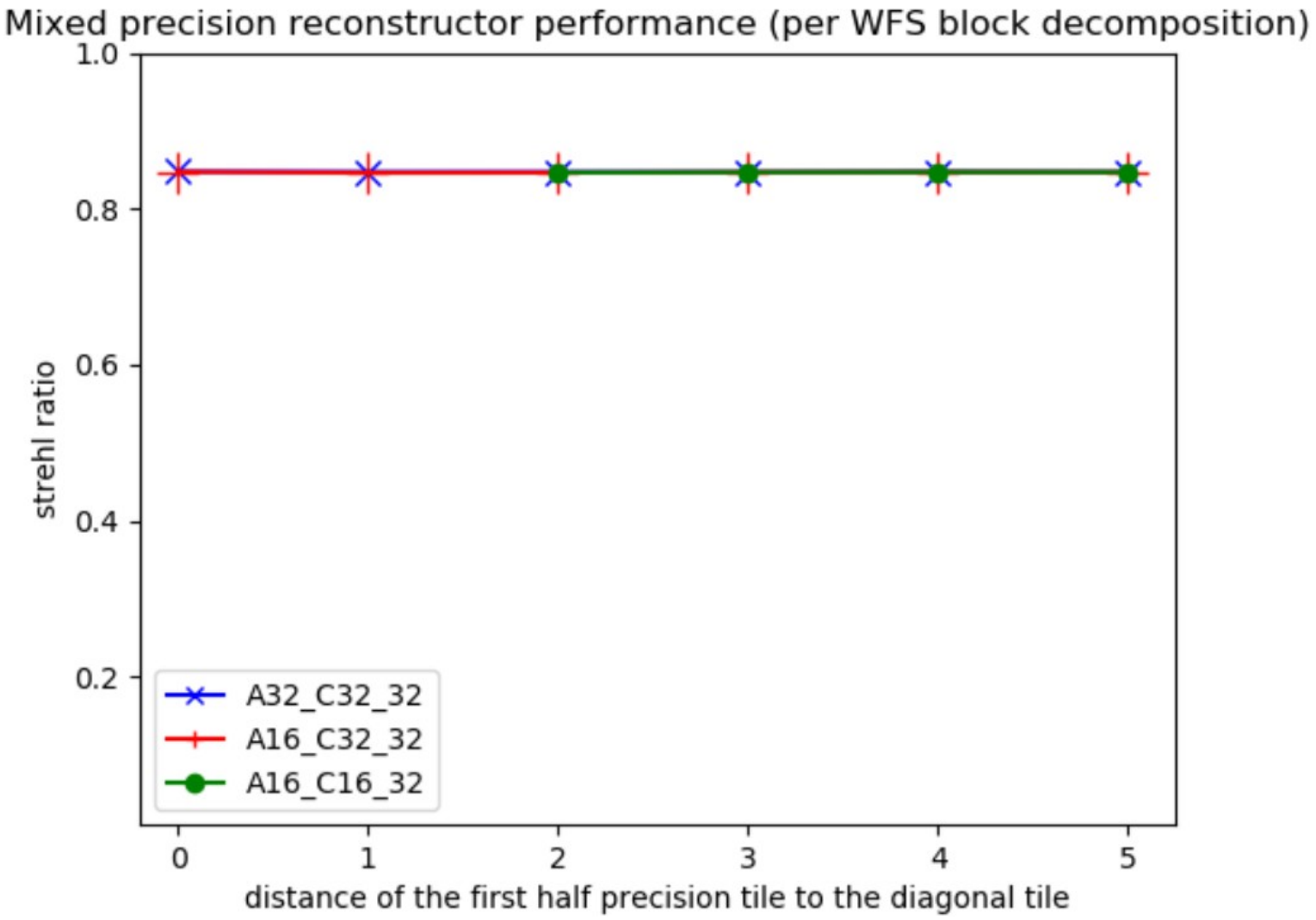
tile low rank
Cholesky: O(kn^2)



Fixed ranks
Preconditioners
Performance oriented

Fixed accuracy
Variable ranks
Dense/Sparse Direct Solvers

Leveraging half-precision



Leveraging half-precision

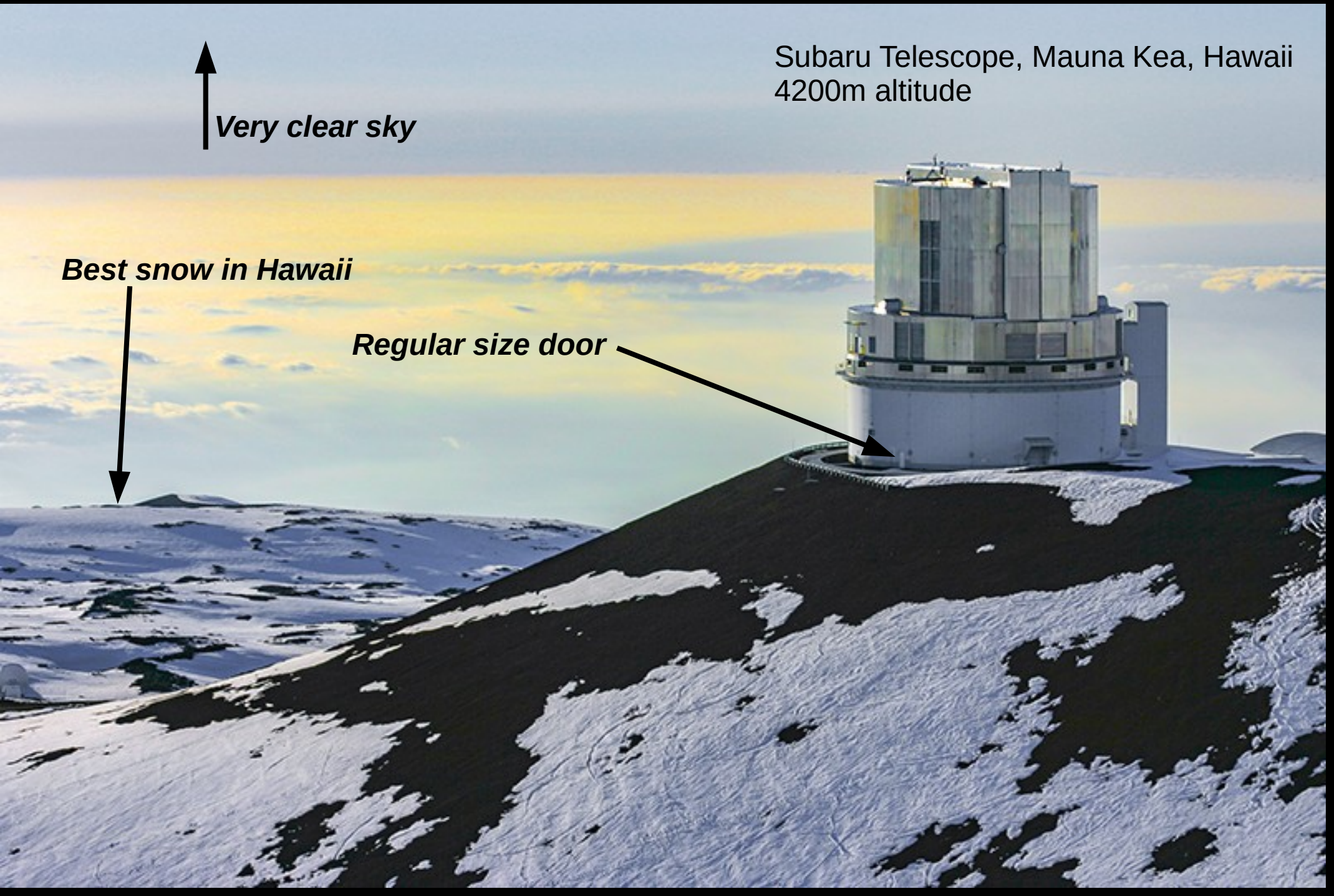
Work in progress on NVIDIA V100 GPUs:

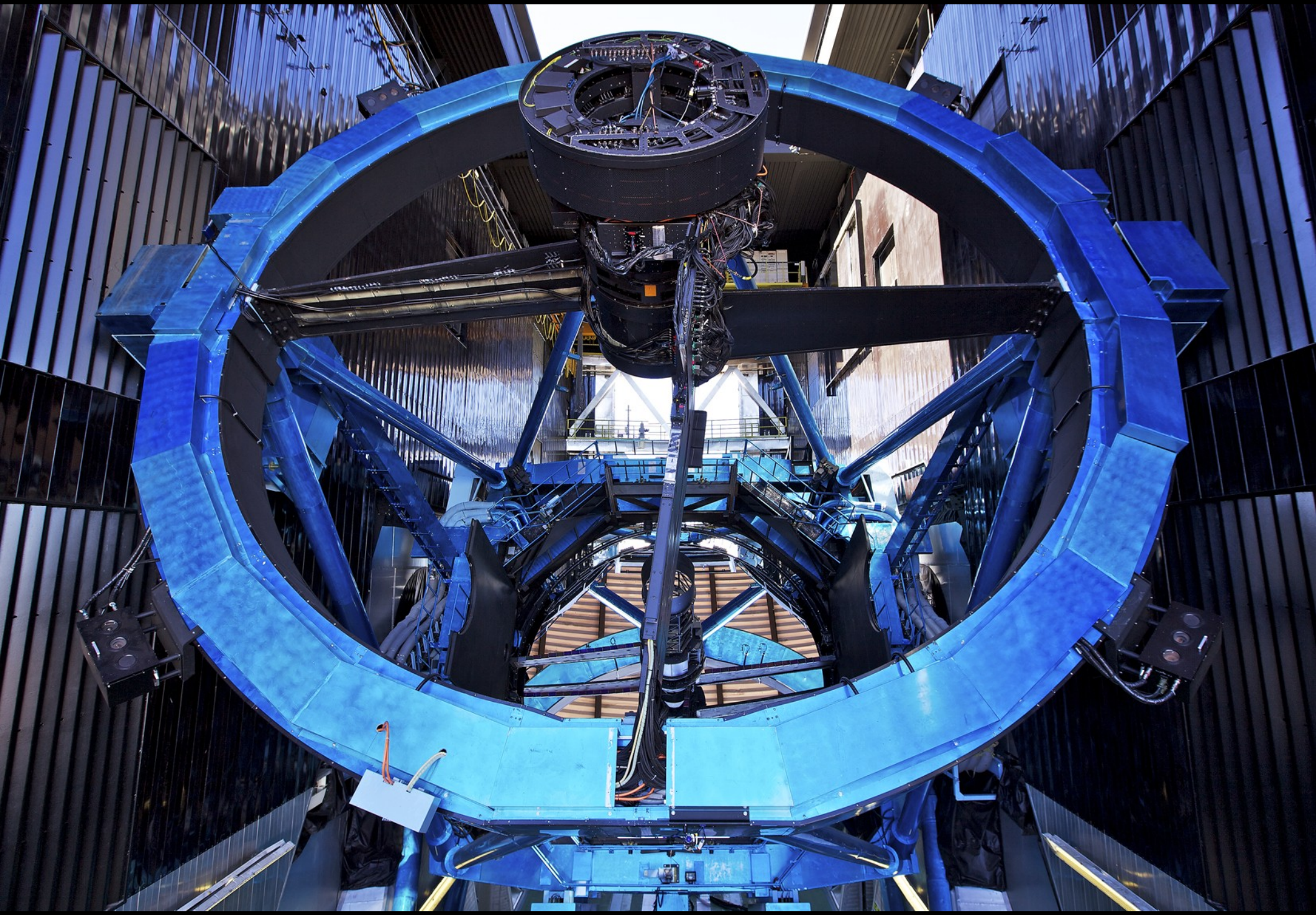
- dgemm achieves about 6.4 Tflop/s on single V100
- sgemm achieves about 14 Tflop/s on single V100
- hgemm achieves about 27 Tflop/s on single V100
- hgemm (w/ tensor cores) reaches about 85 Tflop/s on single V100

- Single precision ToR performance: 42 Tflop/s on 8 V100s
  - That is a ToR at ELT scale computed every 25 seconds
- Speedup factor of 6 between sgemm and hgemm tensor cores
  - 6 x 42 TeraOps/s = 252 PetaOps/s on 8 V100s
  - That is a ToR at ELT scale computed **every 5 seconds**

Probably one of the first real applications amenable for tensor cores usage outside of the traditional AI workloads

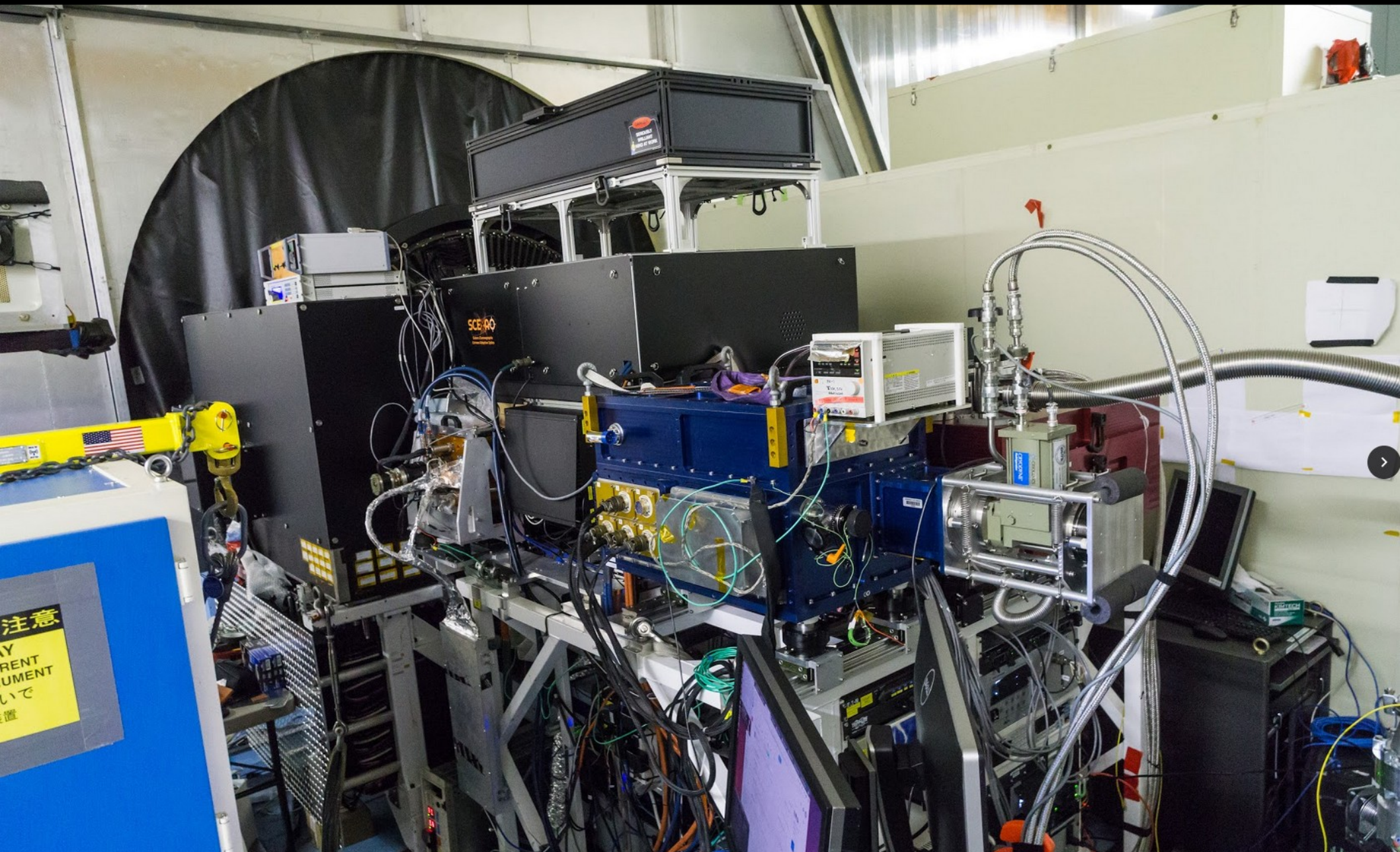
Subaru Telescope (8.2m diameter) has an exoplanet-imaging instrument (SCExAO)
The instrument team is developing advanced Extreme-AO techniques





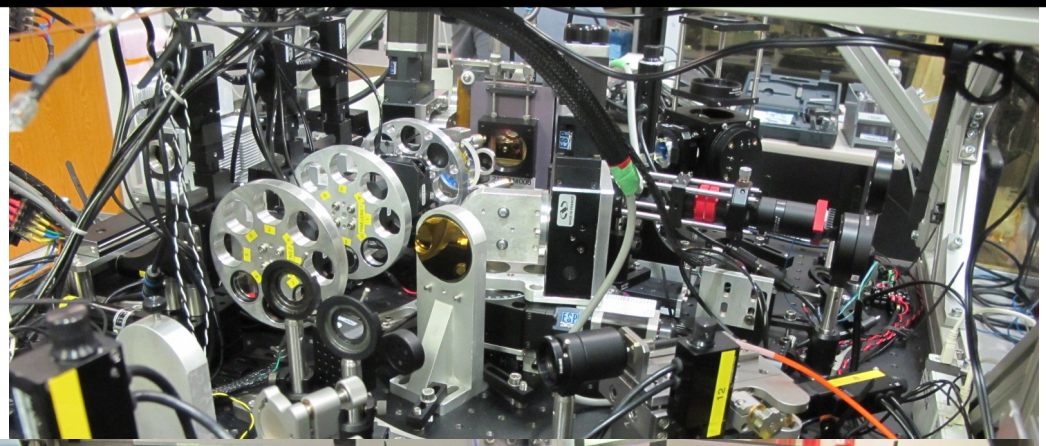
Subaru Telescope (view from inside dome) Photograph by Enrico Sachetti

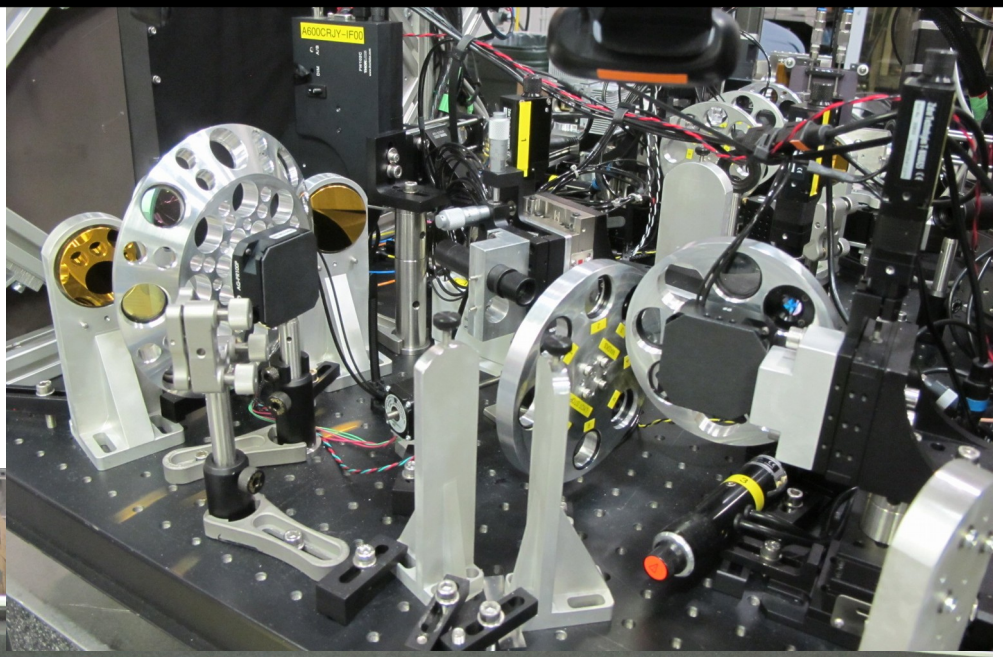
Subaru Coronagraphic Extreme Adaptive Optics

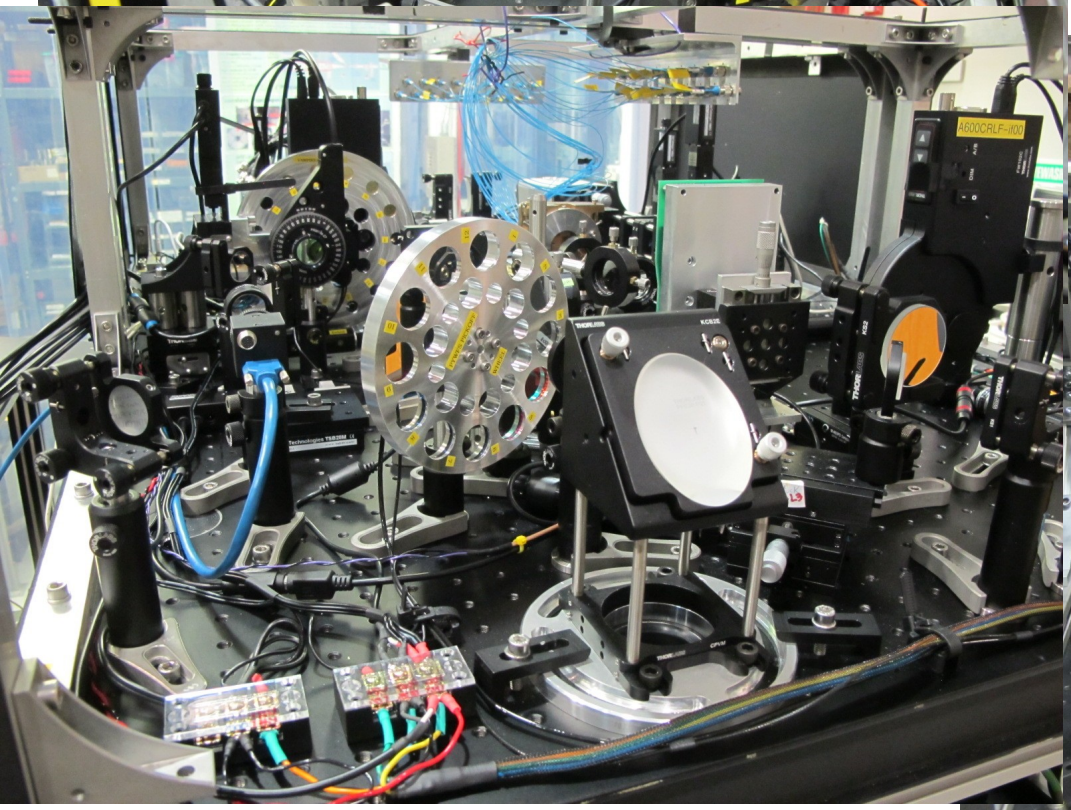


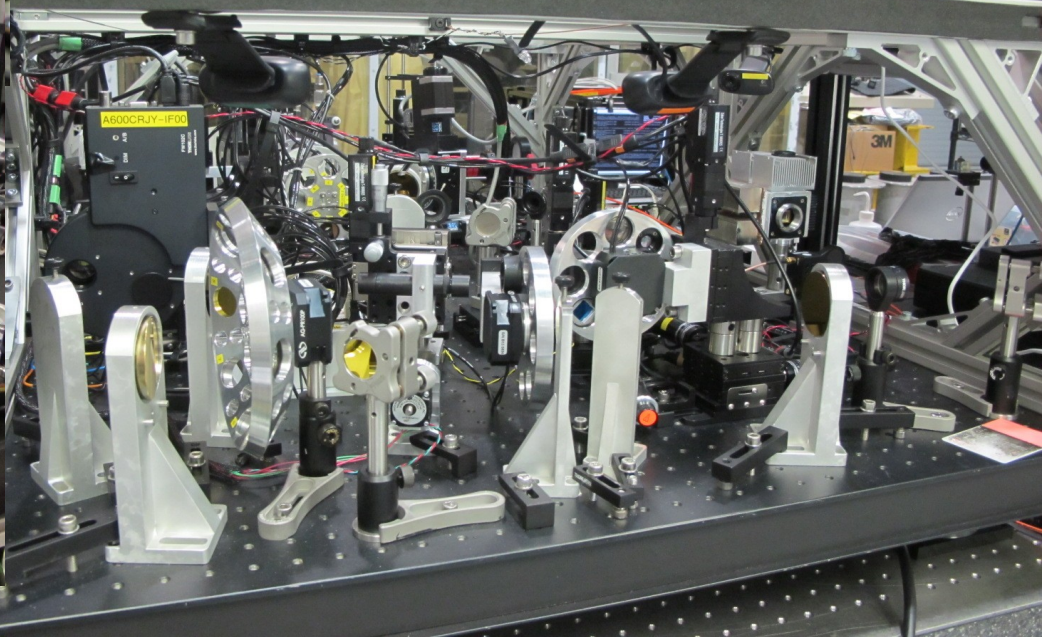
SCExAO

Subaru Coronagraphic Extreme Adaptive Optics





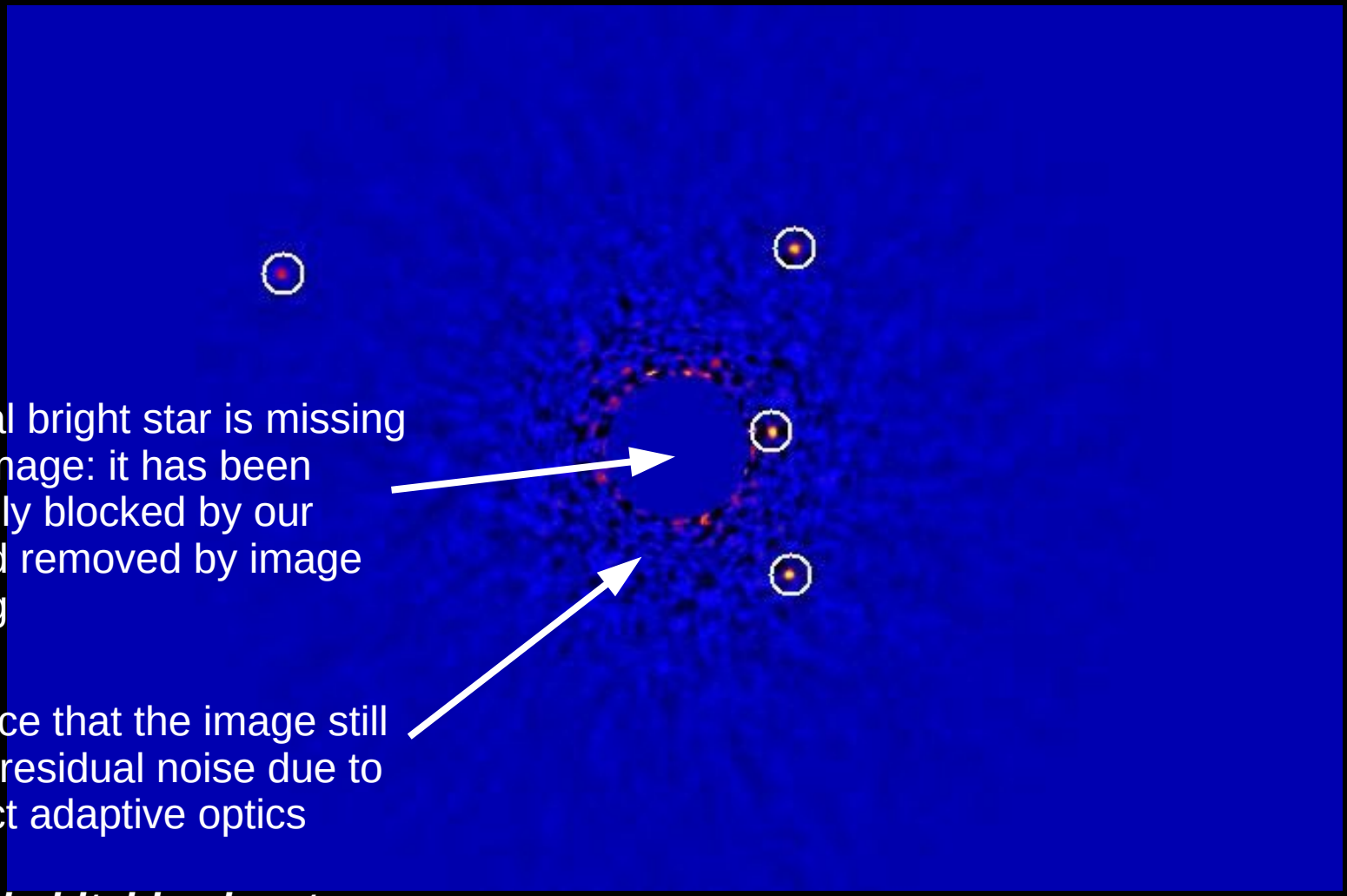




HR8799 system imaged by our instrument

Four planets, orbital periods on the order of 100yr
Each planet 5 to 7 Jupiter Mass

Subaru Telescope/ SCExAO (Currie et. al 2017)



The central bright star is missing from the image: it has been successfully blocked by our optics, and removed by image processing

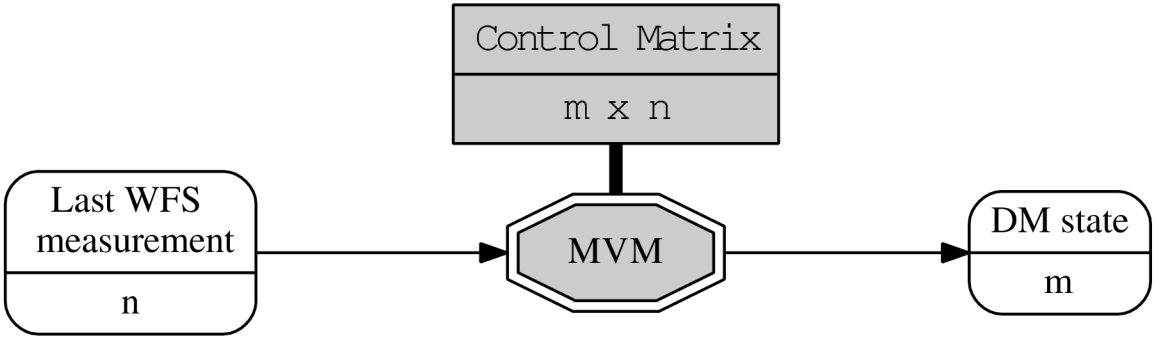
… but notice that the image still has some residual noise due to non-perfect adaptive optics correction.

To image habitable planets, we have to remove this noise

A new approach to Adaptive Optics control

Conventional AO:

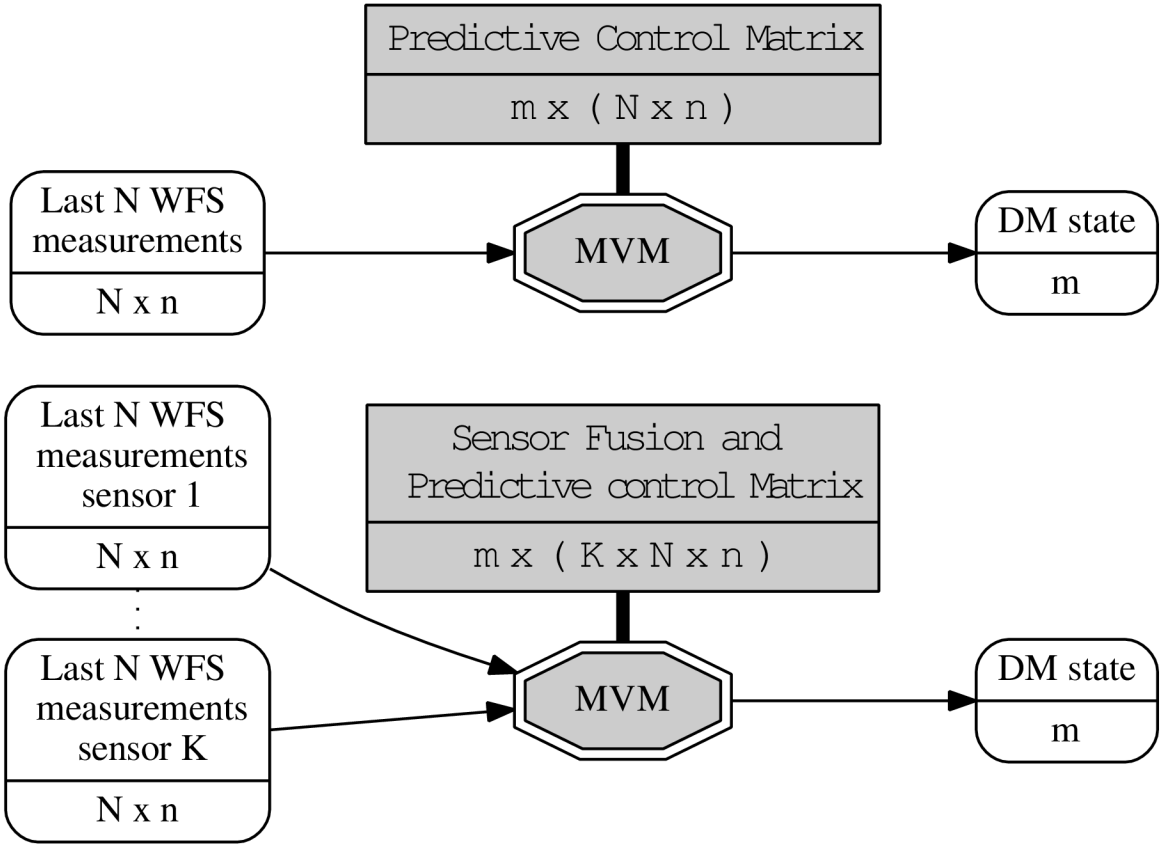
We calibrate the system in the lab, and then apply a linear input-output control law



Advanced AO control:

We continuously, optimally **derive** the input-output controller from the real-time data streams.

This is a formidable computing challenge:
input has ~10,000 degrees of freedom, output has ~2,000 degrees of freedom, and controller must compute solutions at 3 kHz framerate



The Machine Learning challenge

Need to derive 100s of millions of control matrix (CM) values within minutes, using billions of samples...

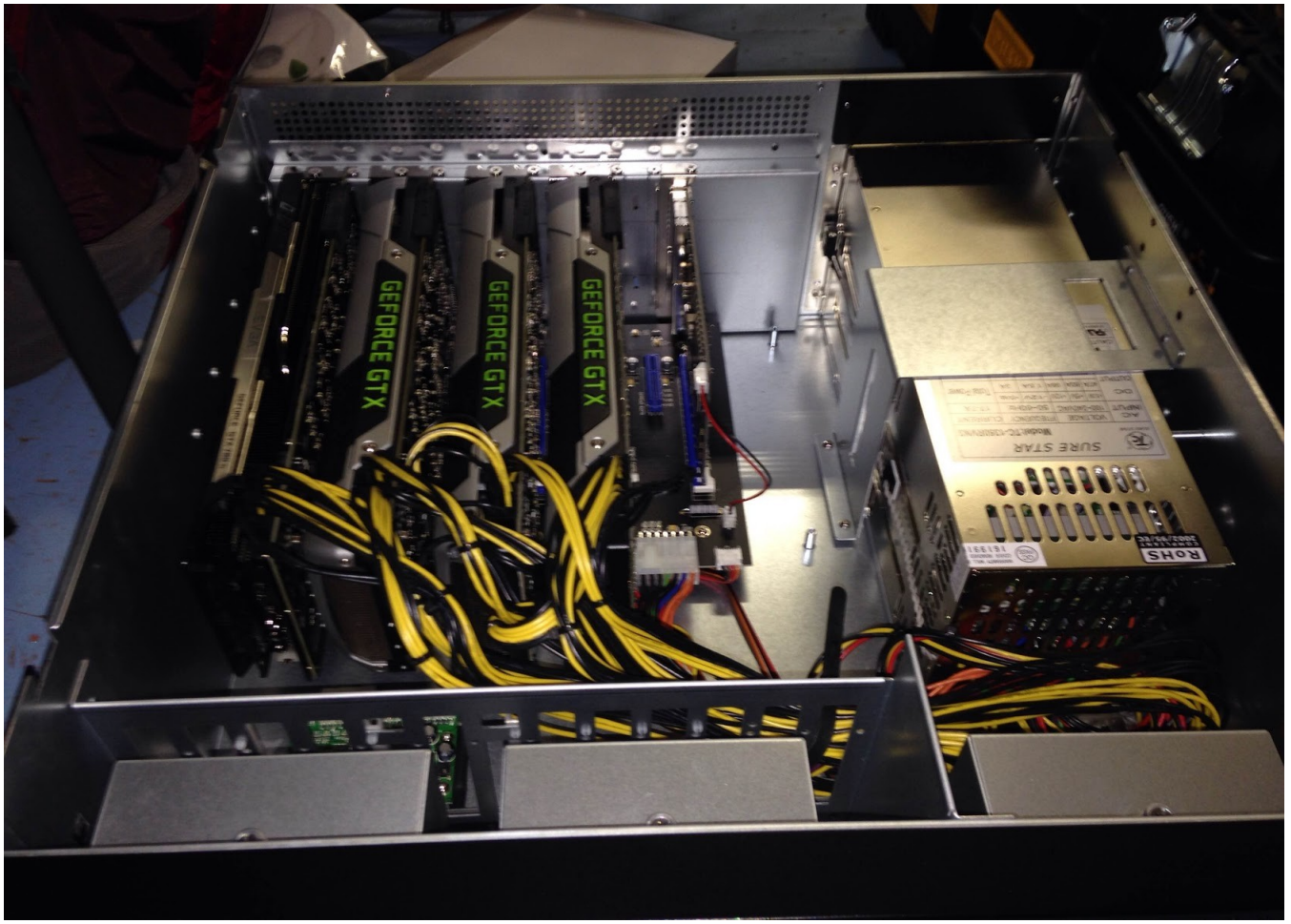
Example:
SCExAO, 3 kHz, 10-step predictive control, 100 sec training
Input: 14,400 x 3,000 x 100 = 4.32e9 measurements
Output: 14,400 x 2000 x 10 = 288e6 CM coefficients

Solution:
We deploy linear ***Machine Learning*** technique on a modal control space (smaller # of dimensions).
We use GPU cores (35,000 cores @ 1.6 GHz in SCExAO main RTC).

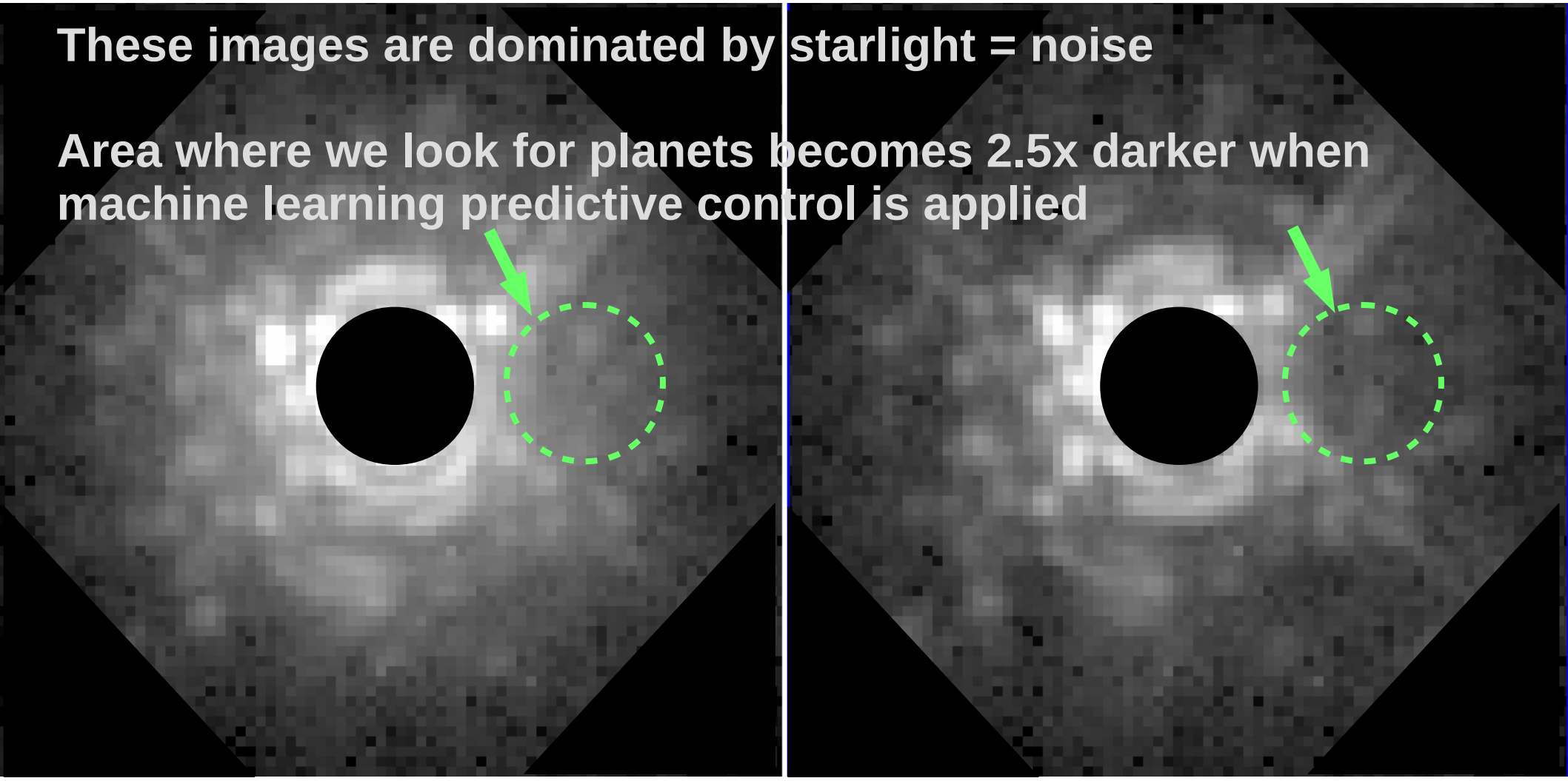
The Machine Learning challenge

SCExAO uses 35,000 cores running @~1.6GHz

One of two GPU chassis



First on-sky results (2 kHz loop)
→ 2.5x contrast improvement



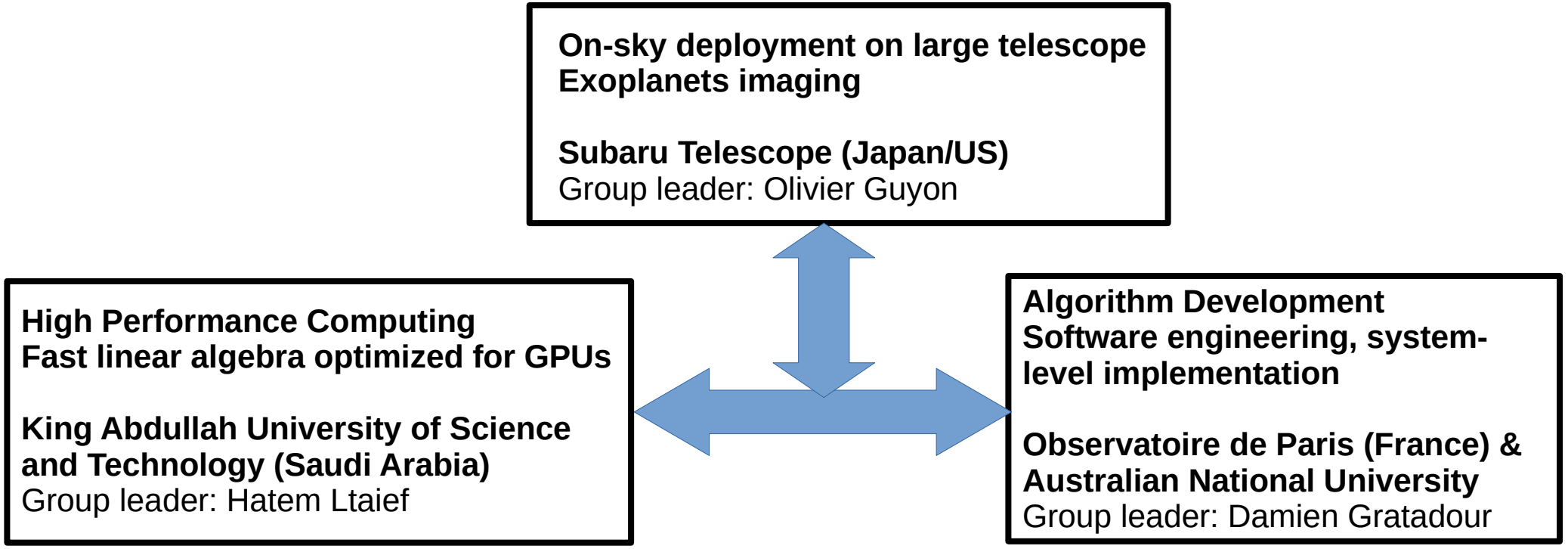
OFF (integrator, gain=0.2) ***ON***

Average of 54 consecutives 0.5s images (26 sec exposure), 3 mn apart
Same star, same exposure time, same intensity scale

Our Team Activities

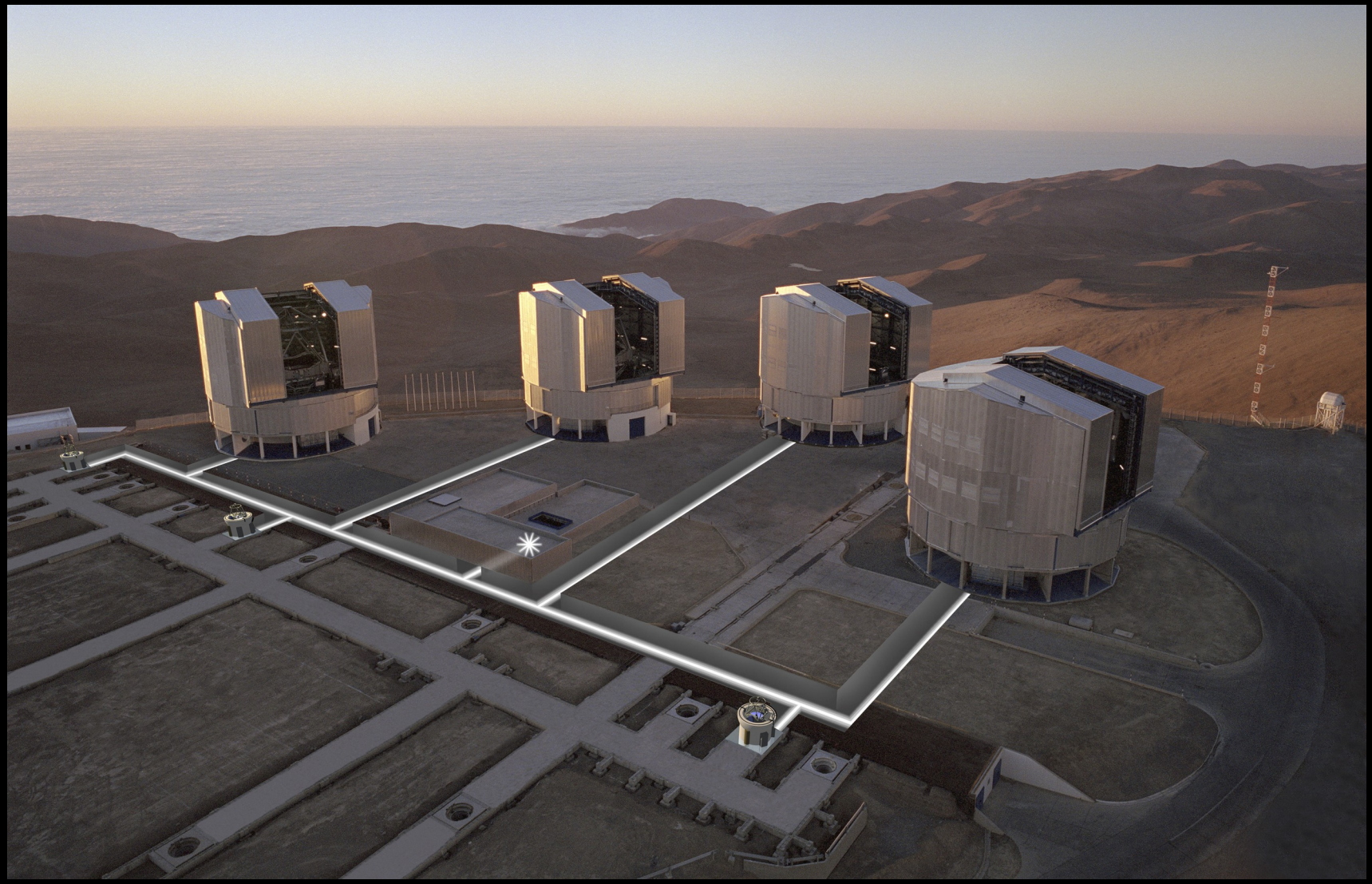
We develop new advanced approaches and algorithms for the exoplanet imaging challenge

We deploy them on the largest telescopes in the world (Subaru, Keck, VLT, ...)

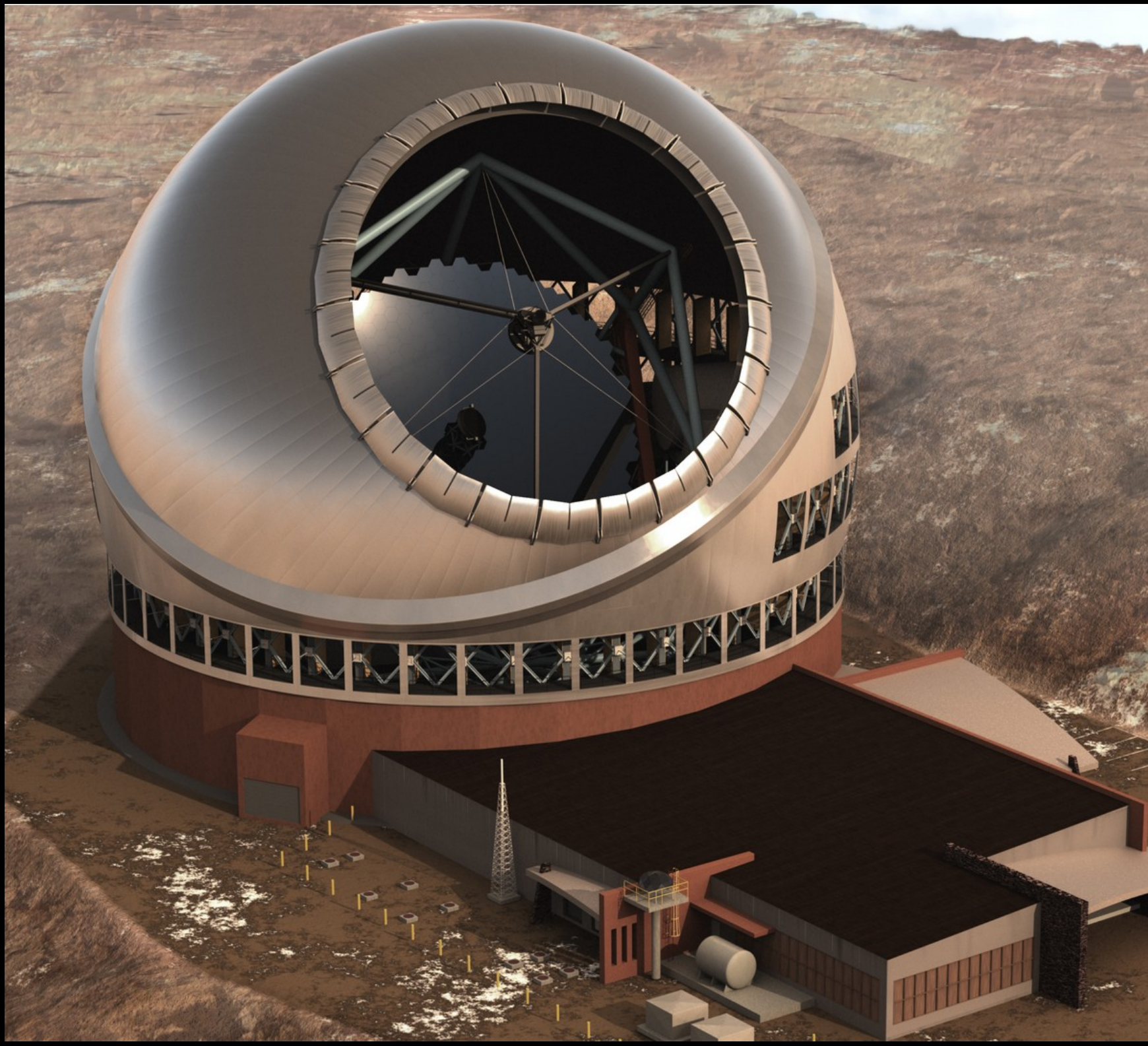


Keck Observatory (US)

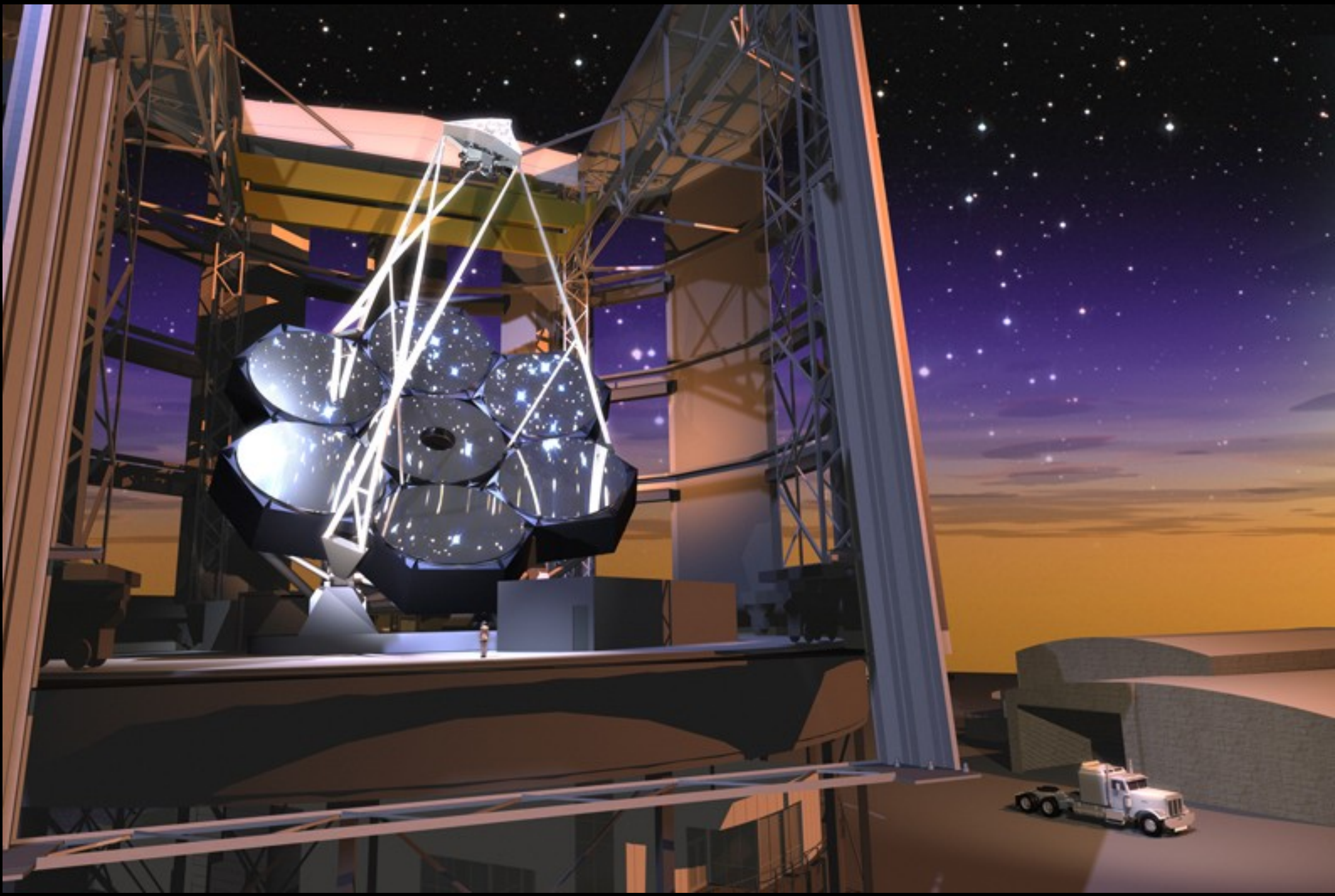
Very Large Telescope
(European facility on Chile)



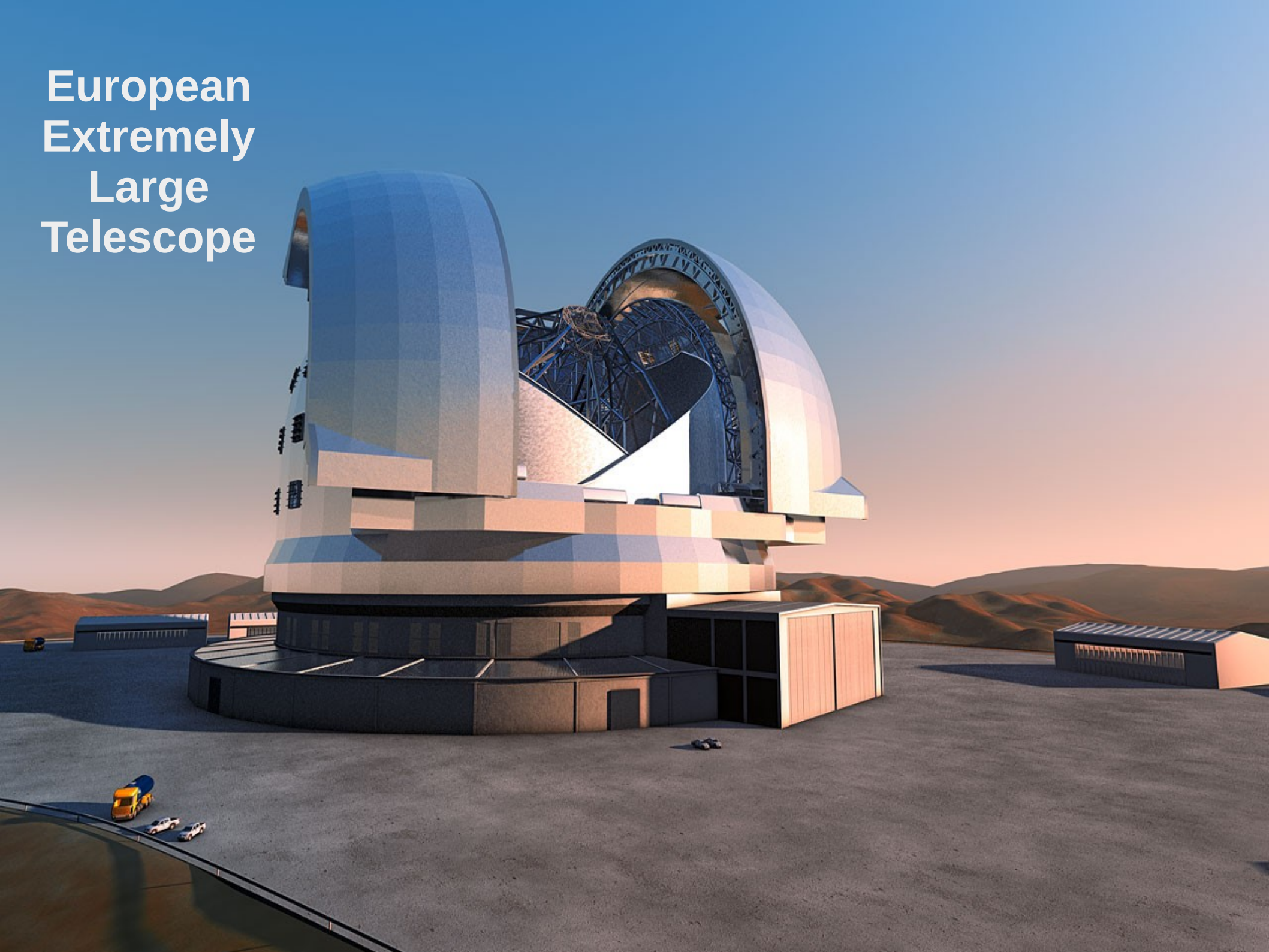
Thirty Meter Telescope



Giant Magellan Telescope

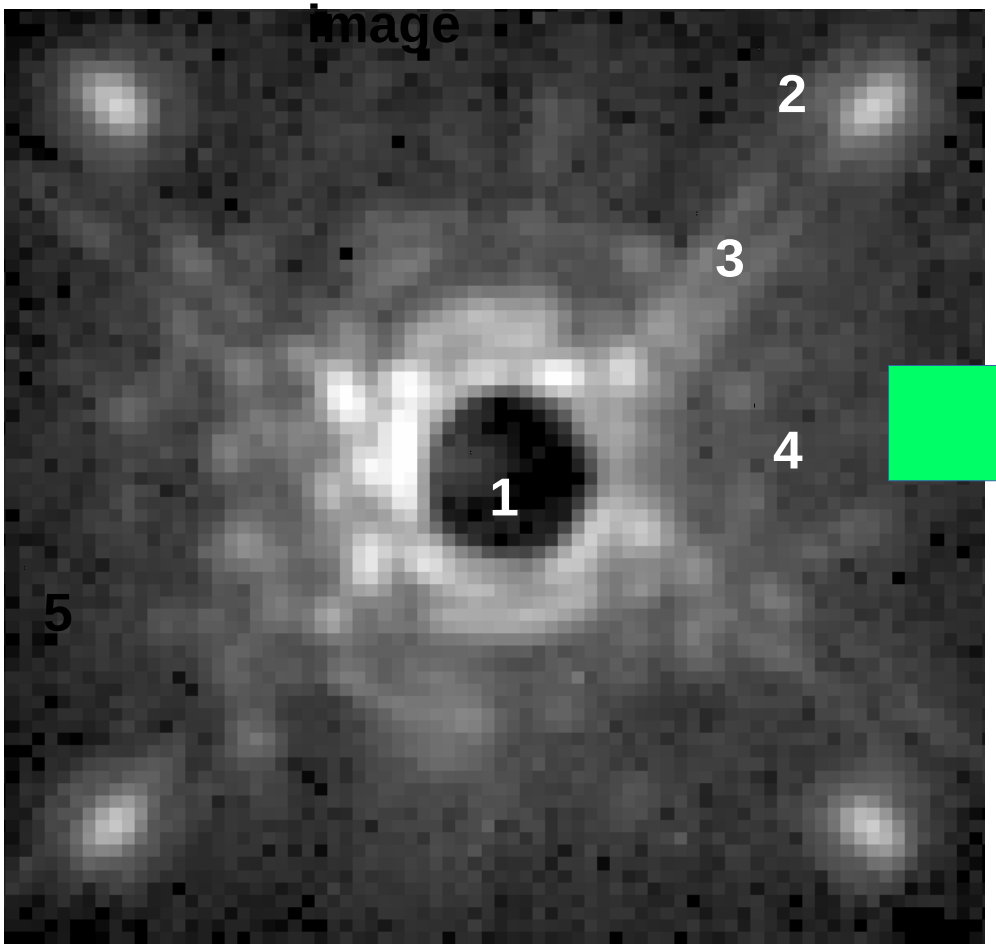


European Extremely Large Telescope

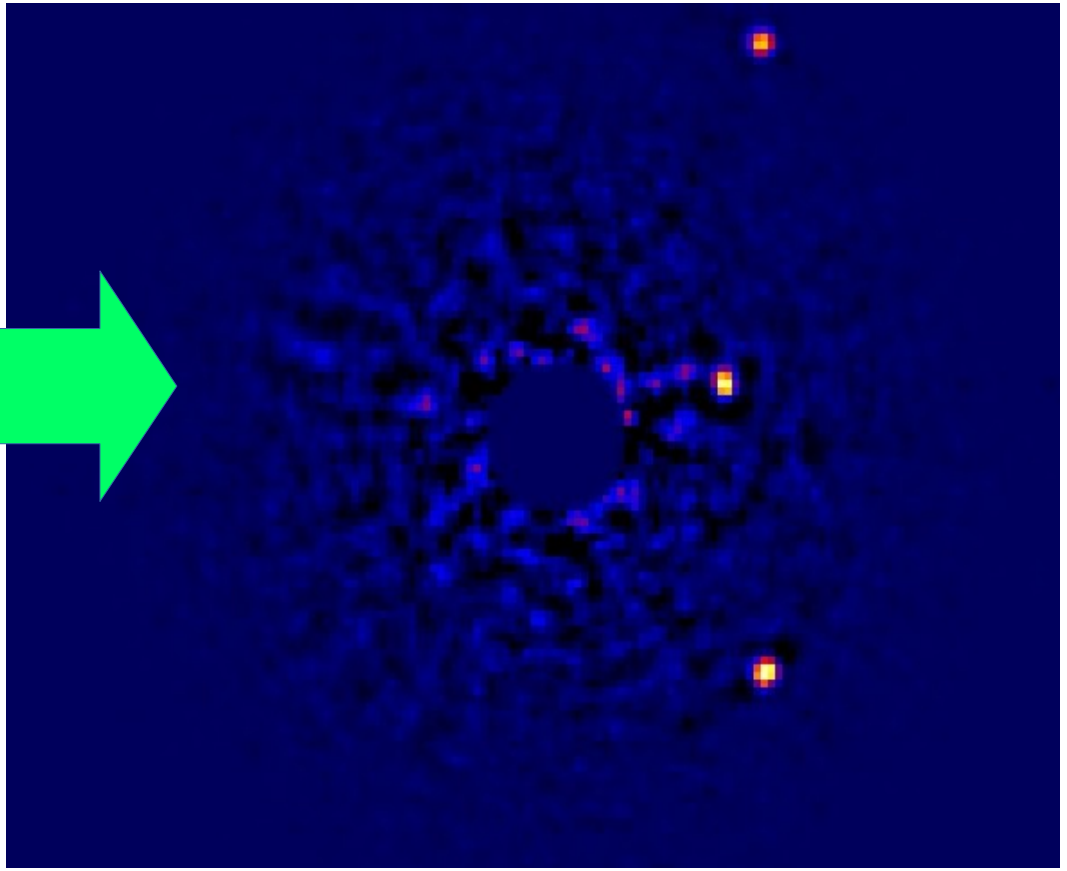


Machine Learning for image processing

RAW image



PROCESSED image



1: Coronagraph Focal plane mask
2: Calibration Speckles (astrometry and photometry)
3: Residual diffraction
4: Speckle Noise
5: Photon and Readout noise

Detection noise dominated by :
- residual speckle noise
- photon noise
- readout noise

On-sky Demonstration (Sept 2018)

We acquired, on the Subaru Telescope, data from two simultaneous cameras:

Science Camera looks at final image for planet, but is dominated by fast-changing unwanted starlight

Sensor Camera looks at starlight that has been rejected by the optical system

Both cameras running at 6.5 kHz frame rate.

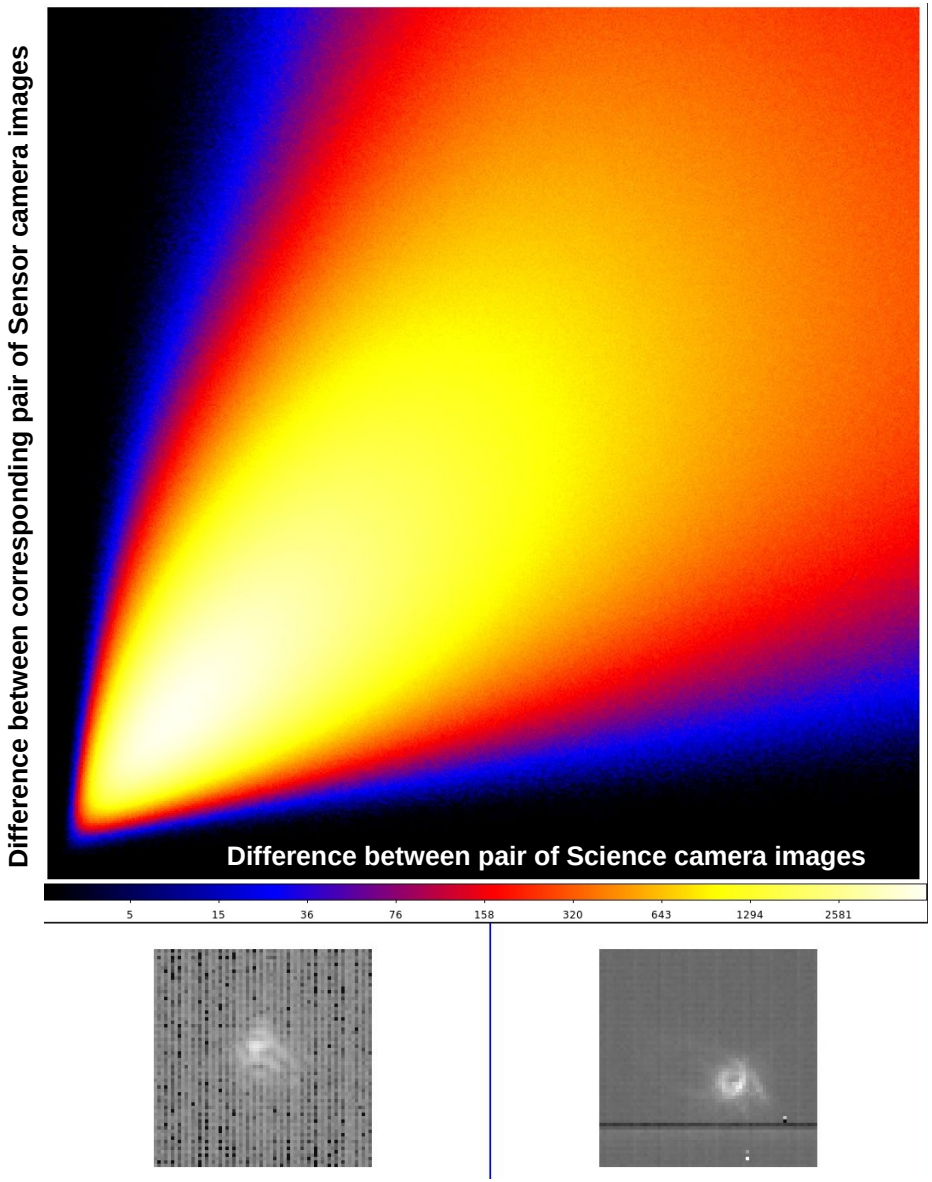
QUESTION: Can we train an algorithm to use the Sensor Camera image to identify where is the unwanted starlight in the Science Camera ?

FIRST TEST (pair-wise comparison): If two sensor camera images are similar, are the two science camera images also similar ? → we can use this information to subtract the starlight

RESULT → On-sky data demonstrates ~10x gain obtained by selecting times when sensor camera images are similar to perform the science camera starlight subtraction.

Computation is extremely challenging due to large number of images (6,500 images per second)

NEXT STEPS (ongoing) : Deployment on GPUs for real-time use.



Neural Net reconstructs on-sky images

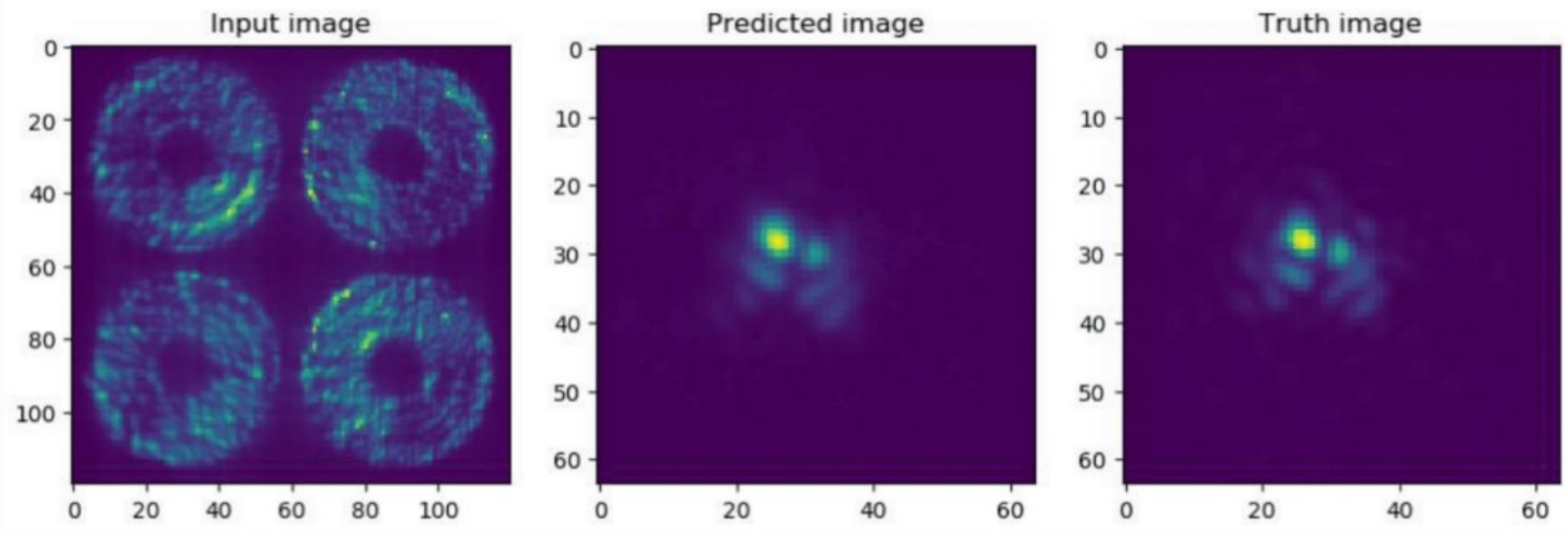


Figure 4: On-sky demonstration of PSF estimation from SCExAO WFS telemetry using a neural network. The training set for this supervised learning problem is constructed by aligning pyramid WFS and visible light PSF frames on the same time reference (hardware lag compensation). While training is slow, inference can be performed in real-time on modern GPUs equipped with tensor cores. We note that visible PSF reconstruction is highly non-linear and particularly sensitive to small wavefront errors. For this simple problem (single input, single output supervised learning), a well-interpolated look-up table built from a clustering algorithm may achieve similar PSF reconstruction quality, but would be considerably more demanding in computing power and memory usage: the main advantage of a neural network approach may here be fast inference speed. Courtesy of Barnaby Norris, Univ. of Sydney.

Courtesy of Barnaby Norris, Univ. Sydney

AO loop learns to optimize image quality

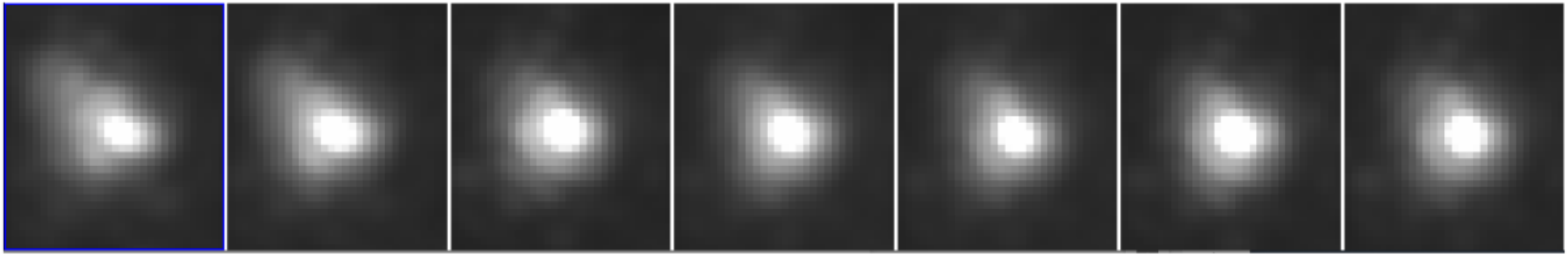
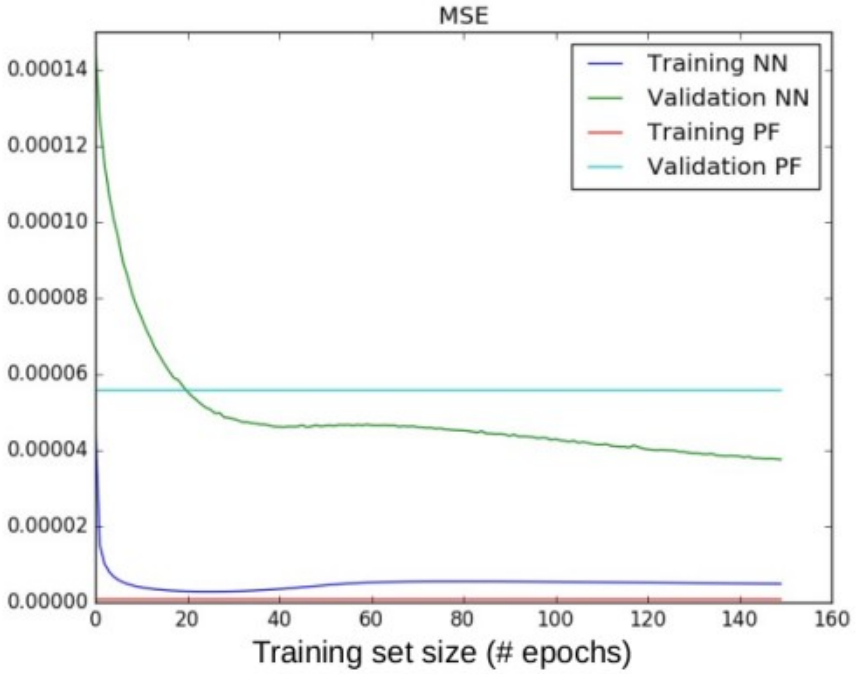


Figure 5: On-sky demonstration of re-inforcement learning for PSF sharpening, using reference updating sensor fusion. The SCExAO pyramid WFS reference on the internal source does not match the on-sky reference due to differences in pupil illumination and variations of chromatic non-common path errors, so it must be learned on-sky from monitoring of the real-time PSF quality. Once the XAO loop is closed, an algorithm identifies the 1% best PSFs and selects the corresponding WFS frames from the real-time WFS telemetry stream. These selected WFS frames are averaged together every 30sec for noise reduction, and the resulting new WFS frame replaces the WFS reference. As the algorithm proceeds, the pyramid WFS is continuously rewarded for high quality PSFs, and the visible light PSF quality improves. The evolution of the on-sky visible (670nm) selected PSFs is shown here over a 21mn period (3.5mn between consecutive PSFs) on the SCExAO system. The strong coma aberration present in at the beginning of the sequence is automatically removed.

Predictive Control using NN



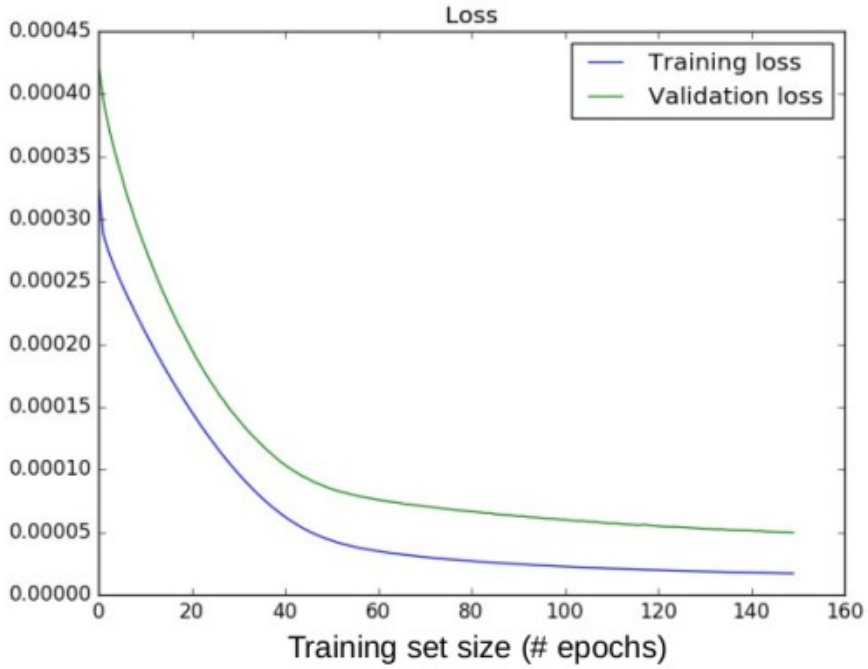


Figure 6: Left: Mean Square Error (MSE) prediction error: comparison between a neural network (NN) and linear EOF predictive filter approach (PF). Right: NN loss function. The NN uses a single hidden layer with 2000 neurons and a sigmoid activation function. The X axis shows the length of the training set (number of epochs). As the training set increases in size, the NN MSE on the validation data improves. The EOF MSE values are only computed for the full training set, so the values are shown as flat horizotal lines. On this very short dataset (150 epochs), the EOF approach overfits the input data, resulting so the training MSE is nearly null, but the validation MSE is poor. NN offers better regularization due to the limited number of neurons, and outperforms the linear EOF approach. Courtesy of Alison Wong and Barnaby Norris, Univ. of Sydney.

Courtesy of Alison Wong & Barnaby Norris, Univ. Sydney

GPUs for optical astronomy

Providing solutions for the most advanced world leading optical telescopes facilities in operation (Subaru, Keck, VLT)

Designing solutions for future giant telescopes leveraging a worldwide collaboration (on 4 continents, Europe, US, Asia, Oceania)

GPUs can power real-time applications at the level of tens of µs

Like the human brain, we use only 10% of their capacity

Extremely high performance available is very promising, will be essential to image nearby habitable planets

Challenging due to high data rate and need for real-time operation