huld

Beyond tomorrow

AI FDIR

Ondrej Harwot

ADCSS 2023, 15th November 2023

Agenda

- Introduction
- Data
- AI design
- AI deployment
- AI V&V
- Summary

huld

huld

Huld (former SSF/SSC Space System Finland/Czech) Technology Design House

Revenue

35 (2020) M€

Huldians

450

Offices

Czech Republic
Prague, Brno

Finland
Espoo, Vantaa, Hyvinkää, Tampere, Jyväskylä, Kuopio, Kotka, Vaasa, Seinäjoki, Oulu, Ylivieska

(On-Board) Software Development

- 17 launched satellites carrying software designed or verified by Huld and
- 15 under work or waiting to be launched
- AI
- GNSS
- FPGA

Introduction

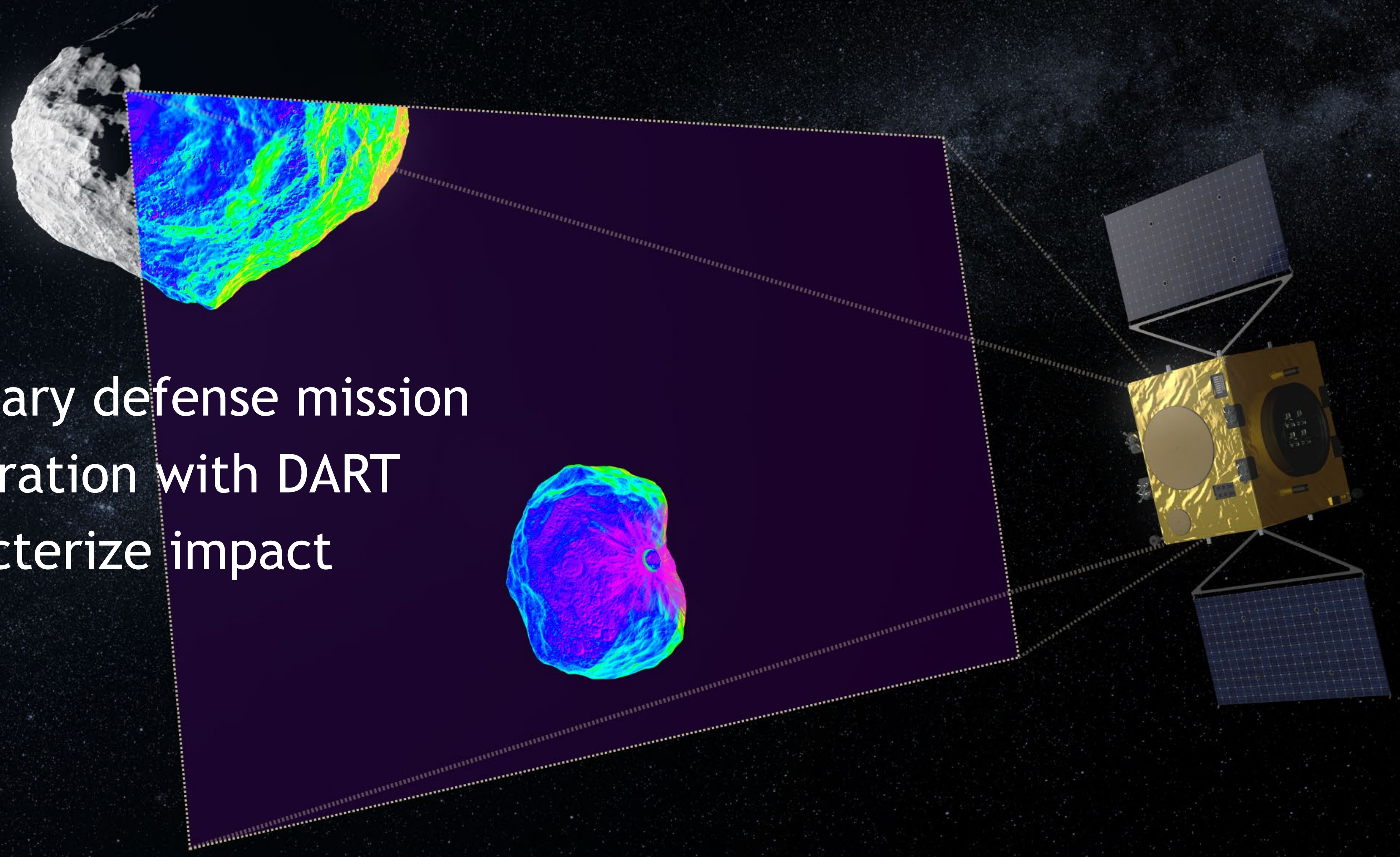
huld

huld

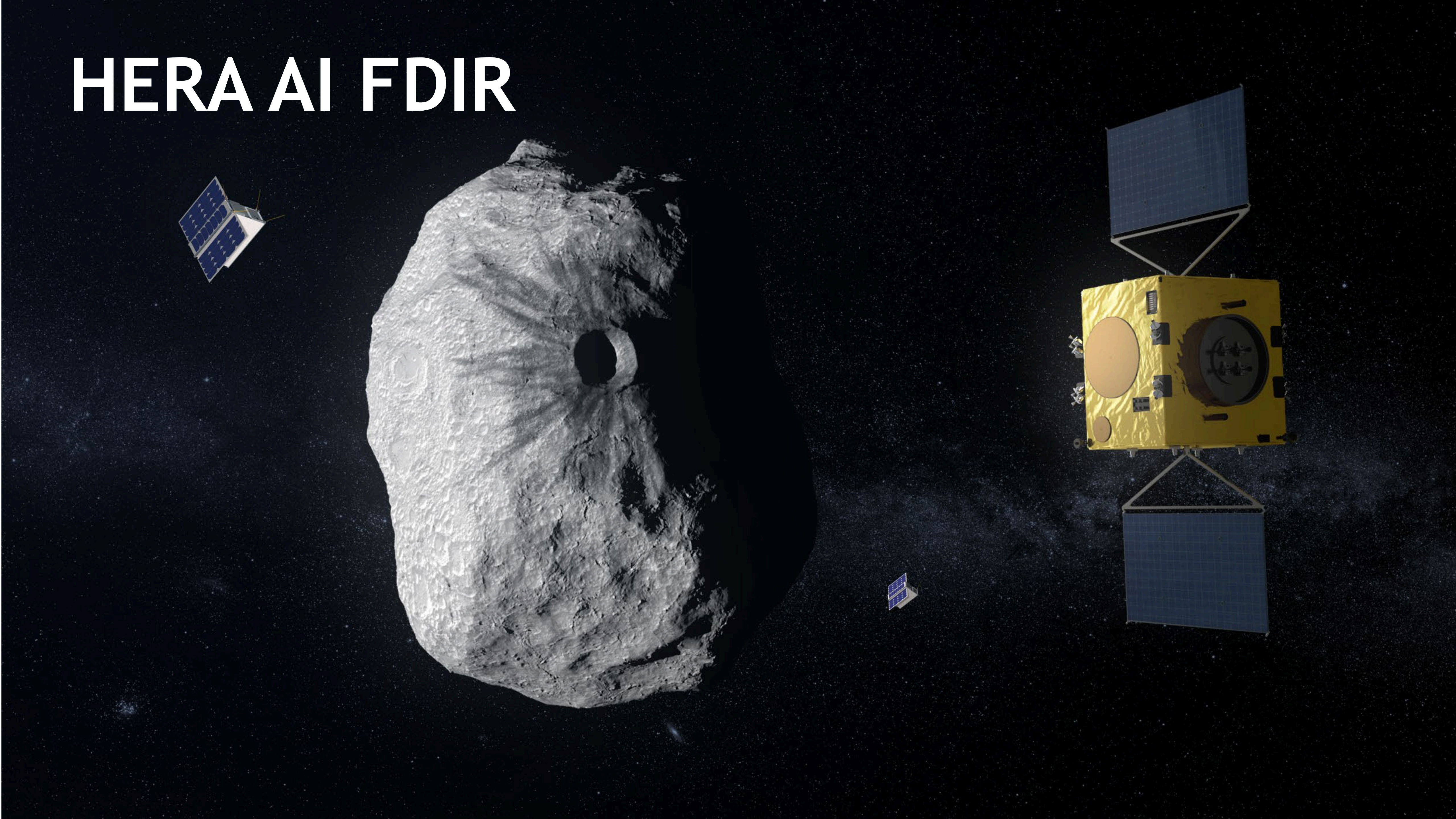
"The idea sounds (a bit) crazy. The real question is if it's crazy enough to actually work."

HERA mission

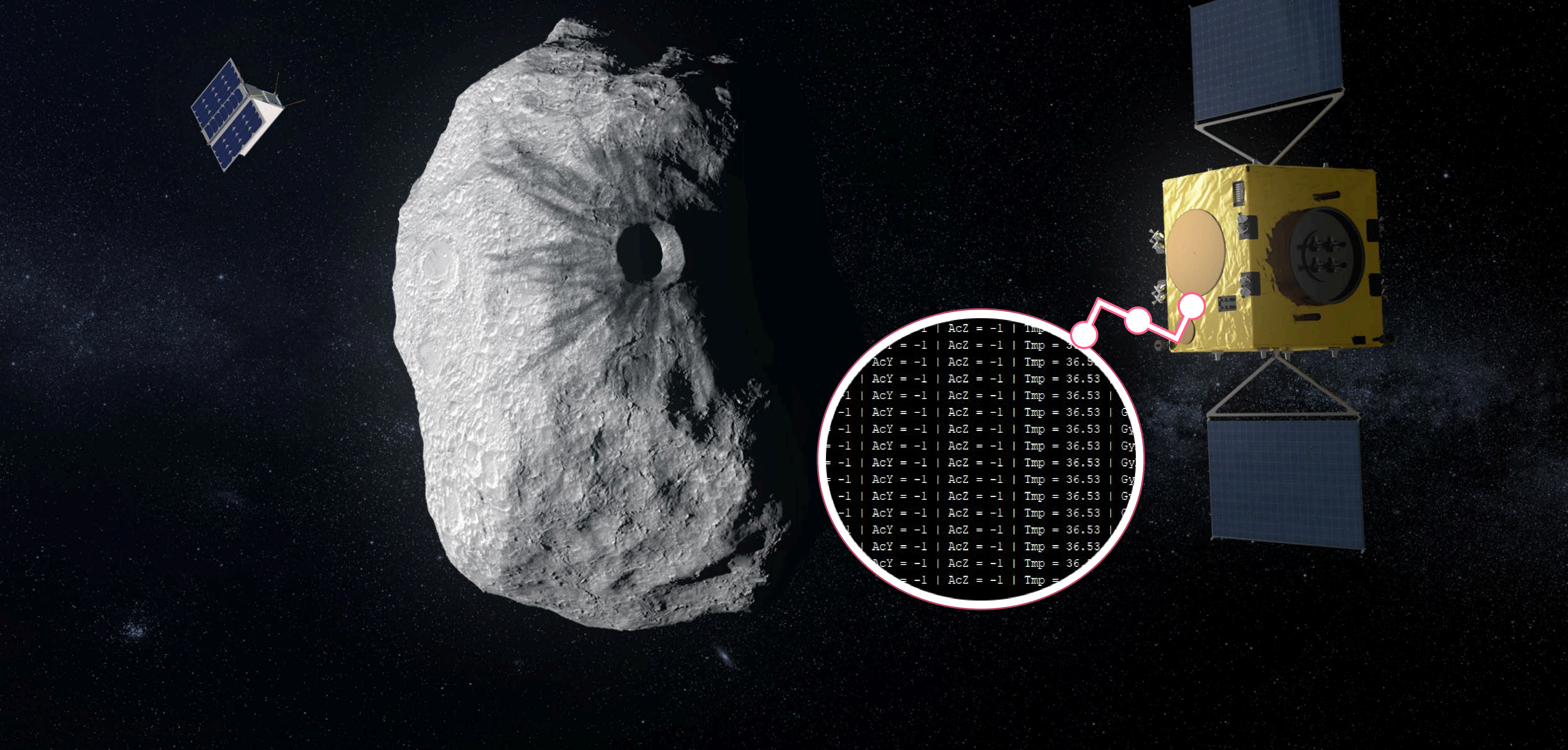
- Planetary defense mission
- Cooperation with DART
- Characterize impact



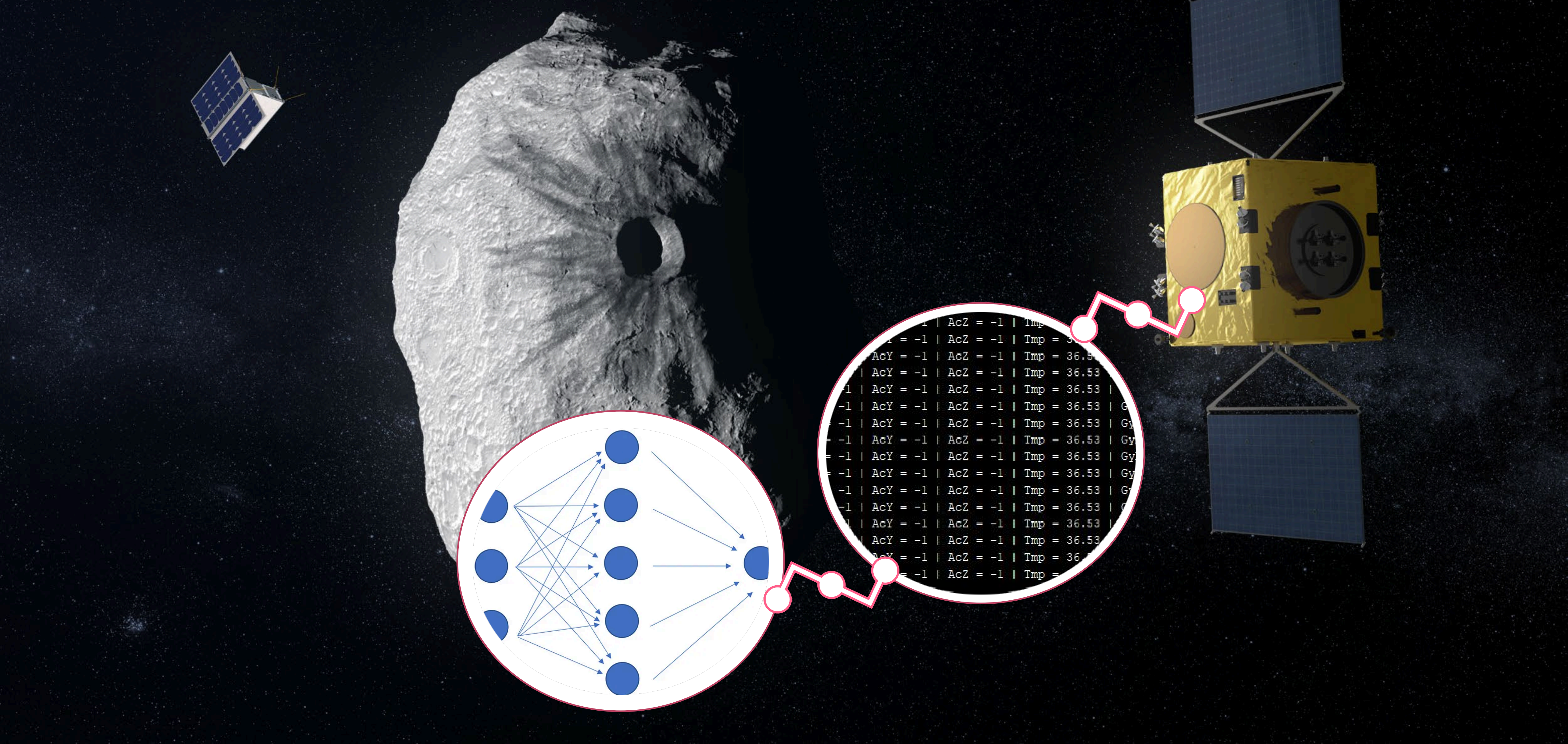
HERA AI FDIR



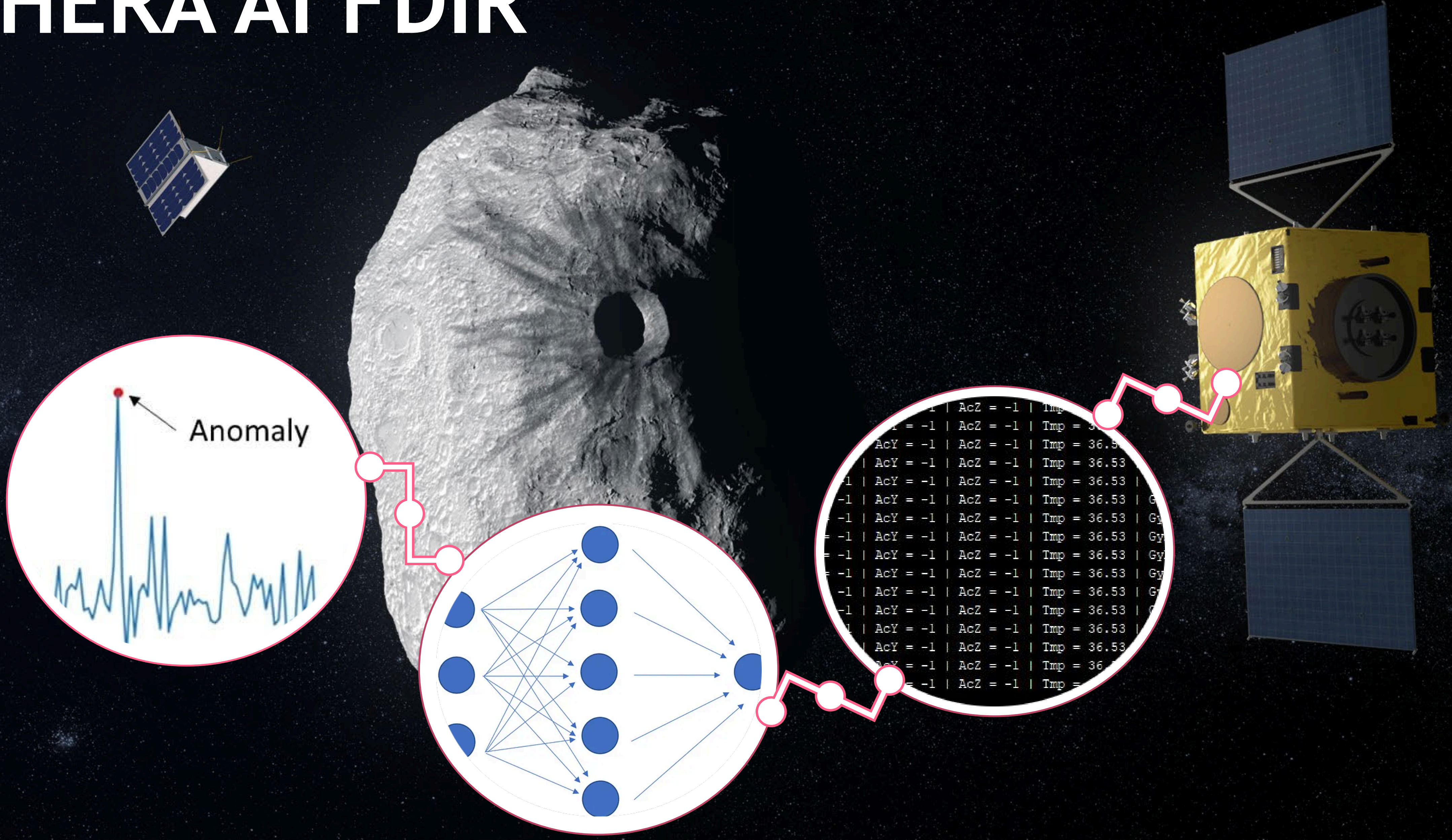
HERA AI FDIR


AcY = -1 | AcZ = -1 | Tmp = 36.53
AcY = -1	AcZ = -1	Tmp = 36.53
AcY = -1	AcZ = -1	Tmp = 36.53
AcY = -1	AcZ = -1	Tmp = 36.53
AcY = -1	AcZ = -1	Tmp = 36.53
AcY = -1	AcZ = -1	Tmp = 36.53
AcY = -1	AcZ = -1	Tmp = 36.53
AcY = -1	AcZ = -1	Tmp = 36.53
AcY = -1	AcZ = -1	Tmp = 36.53
AcY = -1	AcZ = -1	Tmp = 36.53
AcY = -1	AcZ = -1	Tmp = 36.53
AcY = -1	AcZ = -1	Tmp = 36.53

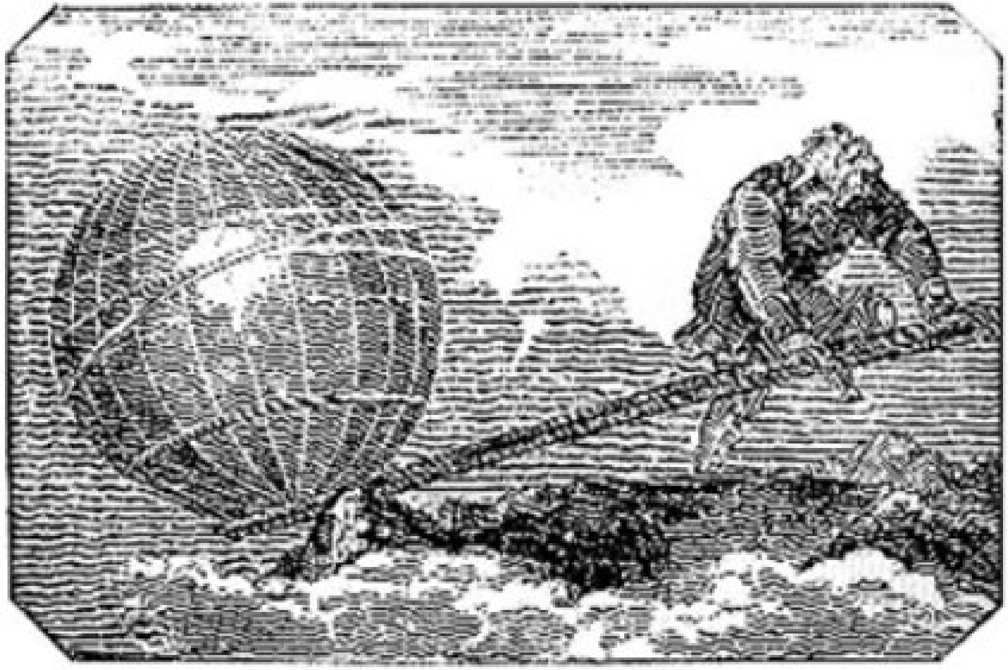
HERA AI FDIR


| AcZ = -1 | Tmp
= -1 | AcZ = -1 | Tmp =
AcY = -1 | AcZ = -1 | Tmp = 36.5
| AcY = -1 | AcZ = -1 | Tmp = 36.53
| AcY = -1 | AcZ = -1 | Tmp = 36.53 |
-1 | AcY = -1 | AcZ = -1 | Tmp = 36.53 |
-1 | AcY = -1 | AcZ = -1 | Tmp = 36.53 |
-1 | AcY = -1 | AcZ = -1 | Tmp = 36.53 |
-1 | AcY = -1 | AcZ = -1 | Tmp = 36.53 |
-1 | AcY = -1 | AcZ = -1 | Tmp = 36.53 |
-1 | AcY = -1 | AcZ = -1 | Tmp = 36.53 |
| AcY = -1 | AcZ = -1 | Tmp = 36.53 |
| AcY = -1 | AcZ = -1 | Tmp = 36.53 |
| AcY = -1 | AcZ = -1 | Tmp = 36.53
= -1 | AcZ = -1 | Tmp = 36
= -1 | AcZ = -1 | Tmp =

HERA AI FDIR



AI FDIR Data



“Give me ~~a lever and a place to stand~~ data
and I will ~~move the earth.~~” design AI for you

Archimedes

What are we looking for?

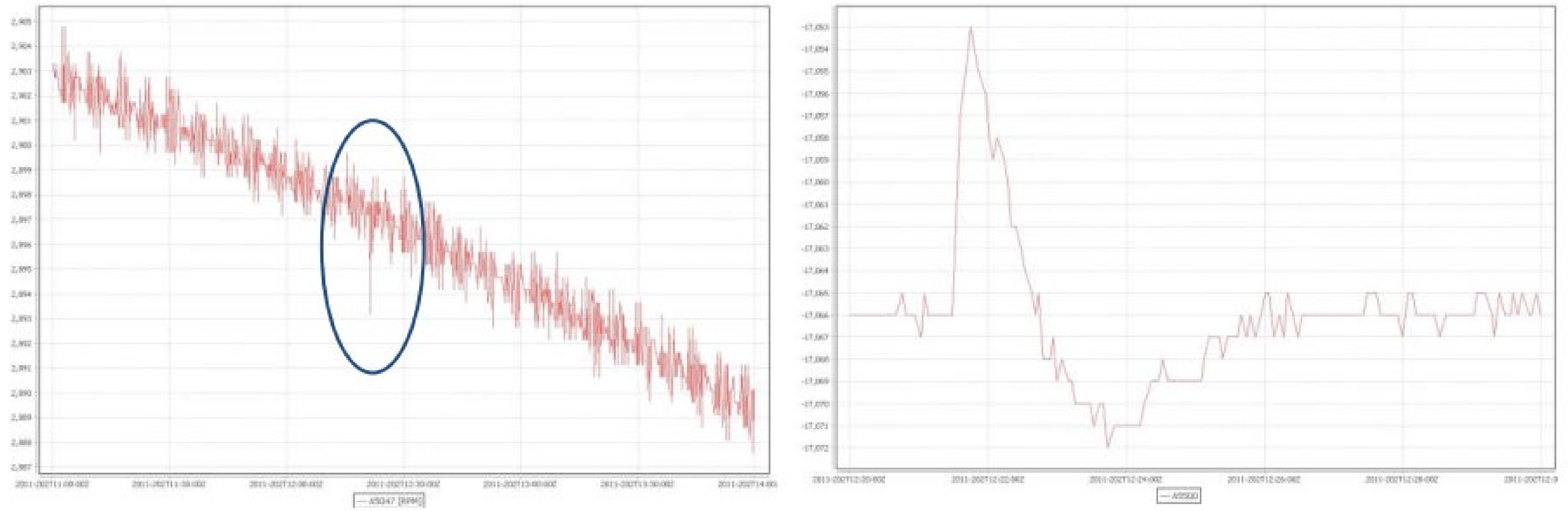


Figure 4.– RW speed variation (left in rpm) and Star Tracker de-pointing (right in CCD units = 0.245. arcsec). The speed change in not much larger than the RW thaco pulse speed sensor, but the de-pointing is up to 10 arcseconds.

[1] Curing XMM-Newton's reaction wheel cage instability: the in-flight re-lubrication experience

[2] Cage Instability of XMM-Newton's Reaction Wheels Discovered during the Development of an Early Degradation Warning System

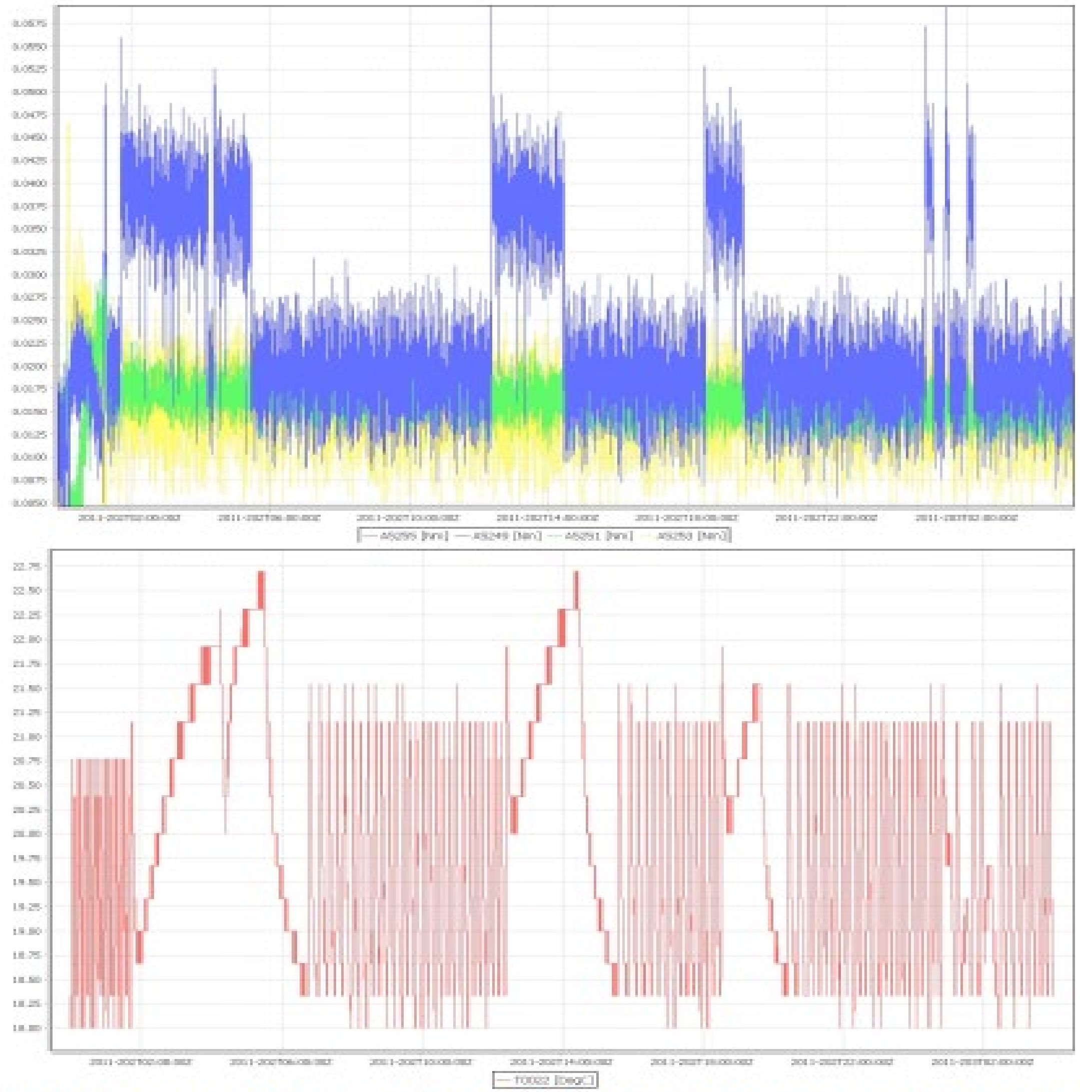


Figure 4 Upper panel: commanded torque of all three reaction wheels during a stable pointing (blue: RW1, green RW2, yellow RW3. The jumps in torque indicate that the RW1 is entering cage instability. Lower panel: The temperature of RW1 stops cycling when the wheel enters cage instability state and increases without heater power added to the system

huld

[3] New Telemetry Monitoring Paradigm with Novelty Detection

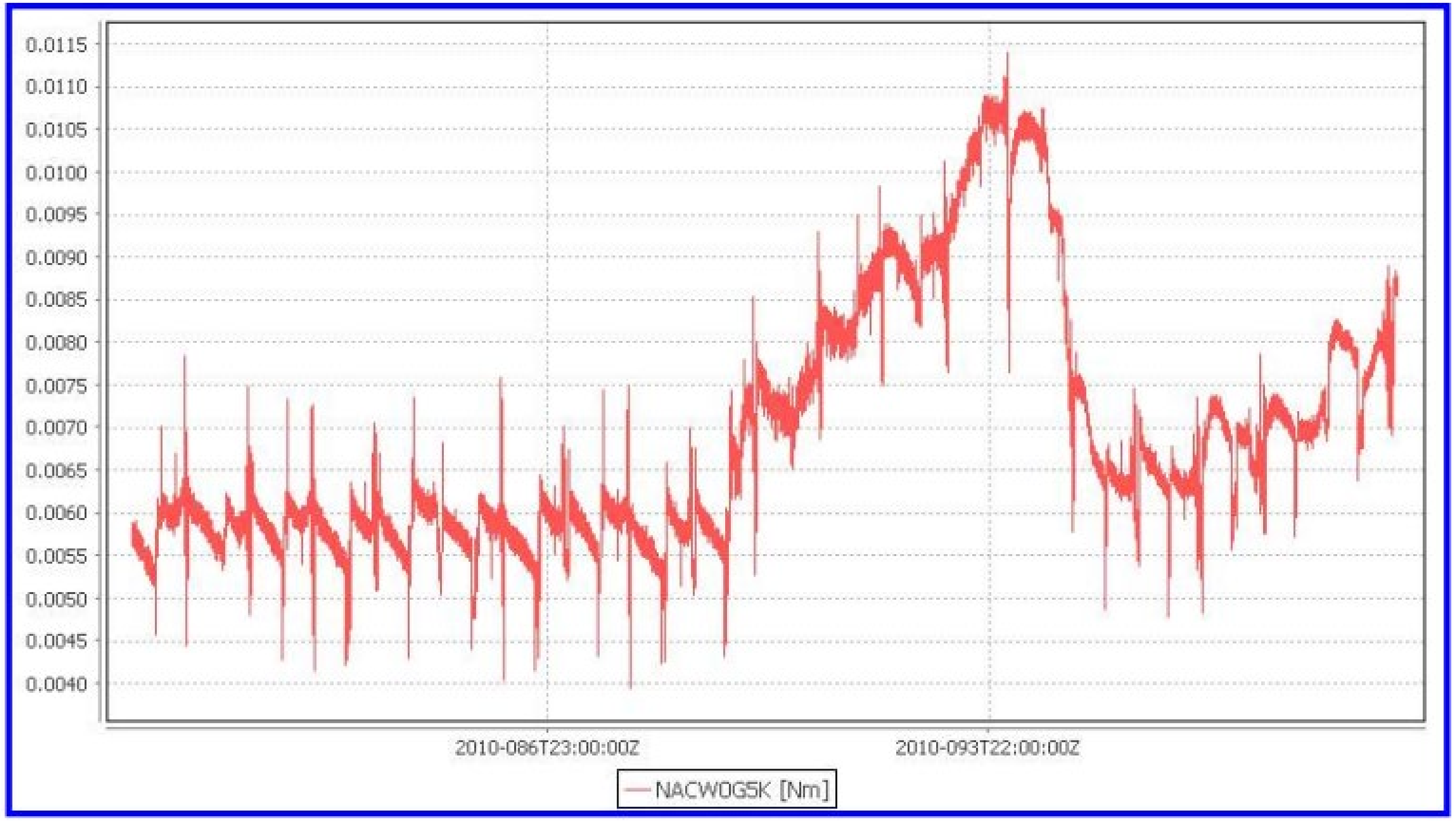


Figure 1. Venus Express Reaction Wheel 4 Friction increases starting in Day of Year 2010.89. *This behavior was not detected by Out-Of-Limits alerts as the upper limit was set to 0.02 at that time. In this case, this behavior was recognized by Venus Express engineers as they closely monitor the reaction wheels even if they do not trigger any Out-Of-Limits.*

[3] New Telemetry Monitoring Paradigm with Novelty Detection

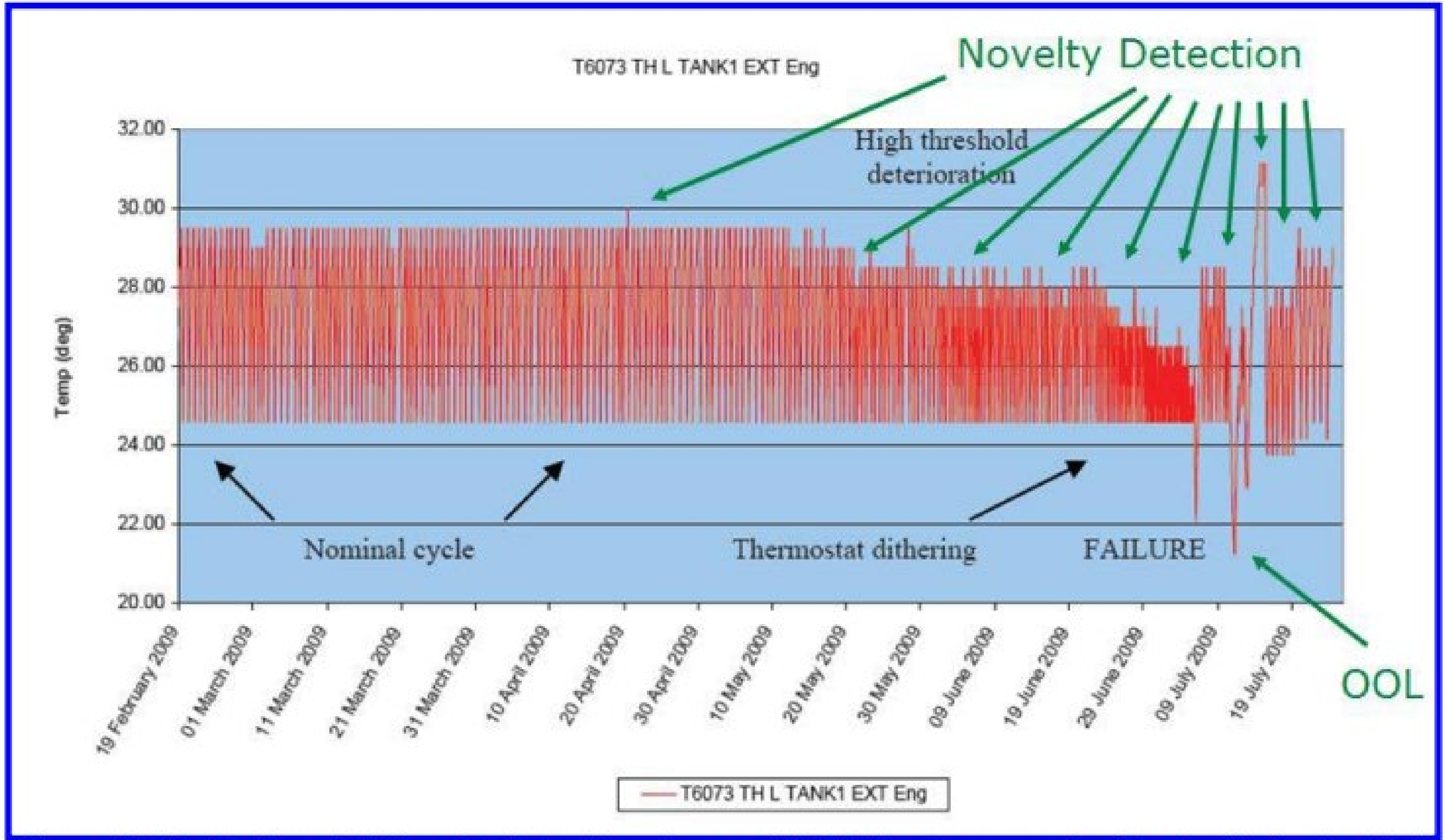


Figure 5. Monitoring with Novelty Detection a known XMM-Newton anomaly. *Nominal behavior: February – April, OOL on 13th July. This thermostat has been properly working showing the same behavior for 10 years. However, it started to have a strange behavior since mid-May 2009 and it was only noticed 2 months after (July 2009) when it crossed the lower limit. For this type of anomaly, the Out-Of-Limits checks are not effective because, paradoxically, the behavior of the anomaly was "more in limits" than before. The proposed novelty detection monitoring technique could find this anomaly 2 months before the Out-of-Limit alarm triggered.*

[3] Discovering outliers in the Mars Express thermal power consumption patterns

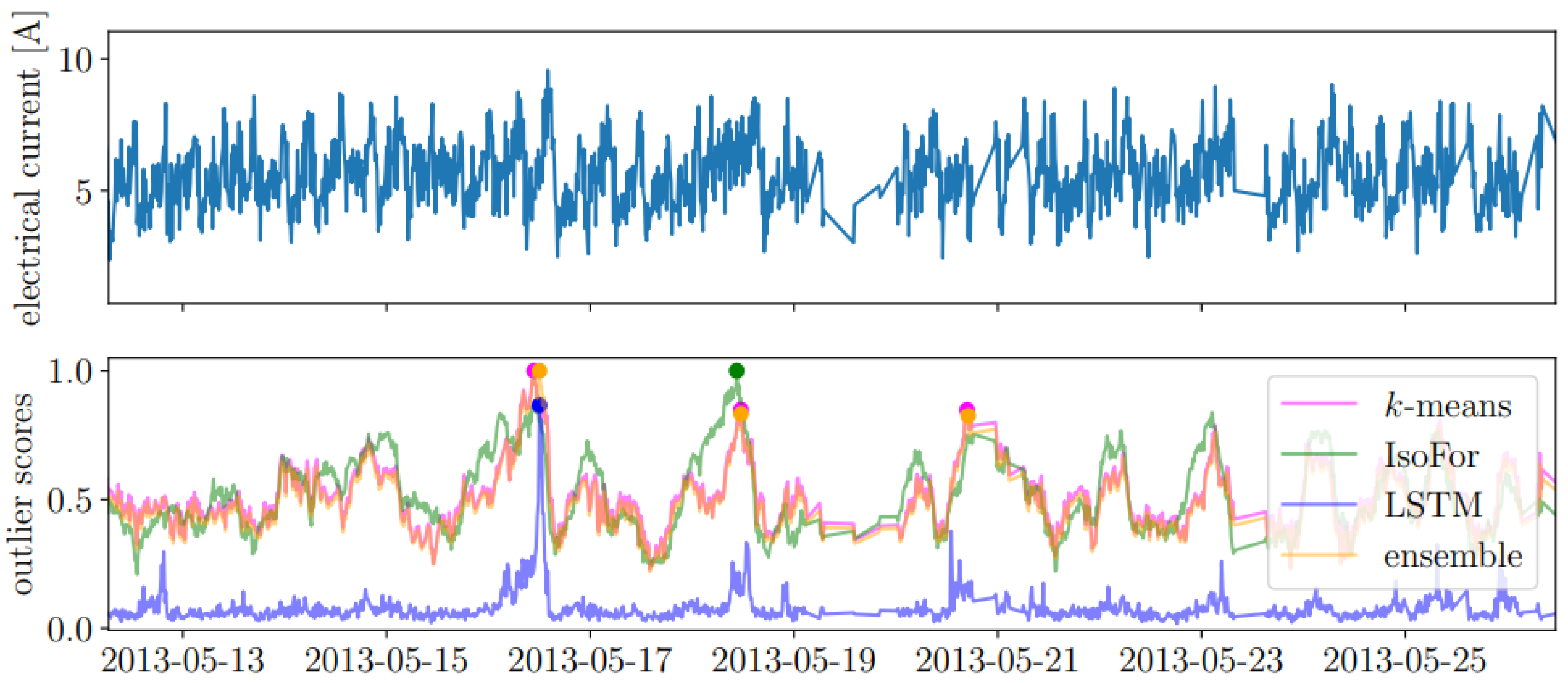


Figure 8: Outliers in May 2013: a) a high peak, followed by quick drop, and b) missing data.

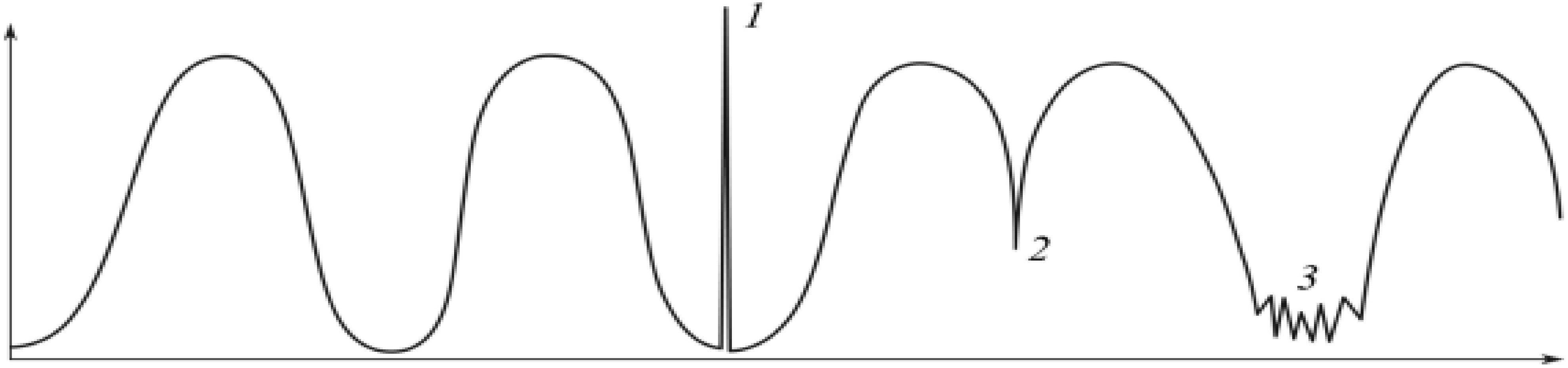


Fig. 1. Anomaly types: (*1*) point, (*2*) context, (*3*) collective.

[4] Modern Machine Learning Methods
for Telemetry-Based Spacecraft Health Monitoring

Dataset preparation is problematic

- Big thanks to ESA for providing datasets from two spacecrafts
- Data are under NDA
- Failure event usually missing
- Annotations, cleaning
- Understanding meaning of each telemetry point requires deep understanding of the spacecraft/payload/instrument
- Data are spacecraft specific
- Data availability – train model before actual spacecraft exists

- ESA activity: Annotating Large Satellite Telemetry Dataset For Esa International AI Anomaly Detection Benchmark

AI FDIR design

Decision made

- Observe each subsystem separately
  - Smaller models
- Observe all subsystems in one big model
  - KPI not better than above approach
  - Can discover (in theory) unseen an unpredicted dependence between systems
- Include TC
  - Model knows what shall happen
  - Very complex development, system understanding necessary, time sync

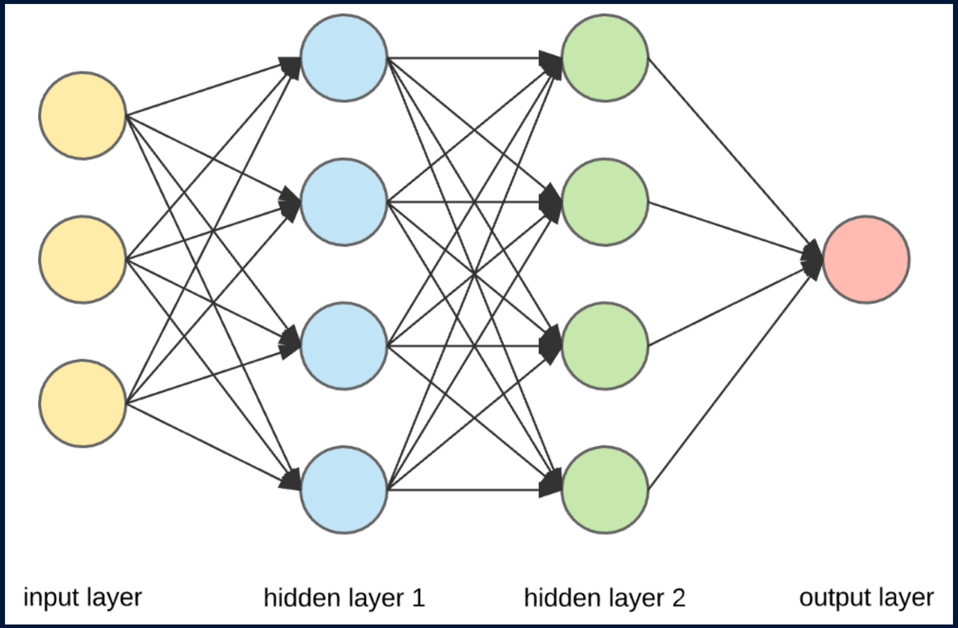
Training approach

- Supervised learning:
  - a model is trained on labeled data to classify anomalies based on previous encounters. Only known anomalies from the training data can be classified correctly.
- Unsupervised learning:
  - a model is trained on unlabeled data to automatically detect anomalies (e.g., 1% of the most suspicious events).
- Semi-supervised learning:
  - a model is trained solely on nominal data to effectively identify new anomalies that may arise but testing on artificially created anomalies

huld

AI architecture

- Encoders - Siamese networks
  - identical subnetworks sharing weights and are trained on pairs of inputs, one being true and one false
- Outlier detector - KNN
  - k-nearest neighbors algorithm
- Other tested approaches include
  - Convolutional encoder, convolutional variational, LSTM, isolation forest, one class SVM



huld

AI architecture

Encoder	Detector	Precision	Recall	F1 score
Conv. AE	KNN	0.840	0.808	0.804
Conv. AE	I-forest	0.742	0.723	0.718
Conv. AE	Reconstru-ction error	0.766	0.764	0.764
Var. conv. AE	Reconstru-ction error	0.895	0.895	0.895
Siamese Conv. AE	KNN	0.995	0.995	0.995
Siamese LSTM	KNN	0.994	0.994	0.994

AI FDIR integration

huld

Our target: fit one core of Leon 3, no OS, limited memory, avoid affecting main OBSW

Convert Python into C/C++/HDL

- Fdeep
  - https://github.com/Dobiasd/frugally-deep
  - C++ templates, Eigen, JSON
  - Need OS to compile
- uTensor
  - https://github.com/uTensor/uTensor
  - Primary target platform ARM
  - extremely light-weight
- TFMin
  - https://github.com/PeteBlackerThe3rd/TFMin
  - converts CNN into pure C++, Eigen dependency
  - Supports only TF 1

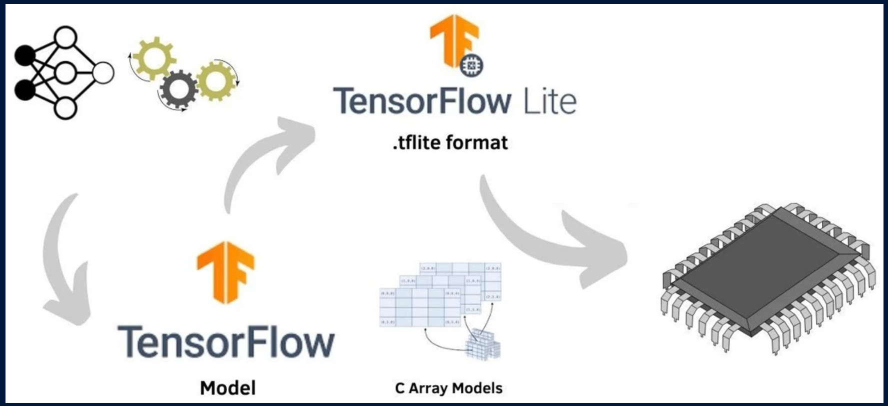
huld

Convert Python into C/C++/HDL

- Apache TVM
  - https://github.com/apache/tvm
  - Compilation of deep learning models into minimum deployable modules
  - CPU, GPU, Xilinx, C, C++
- MATLAB Coder
  - Not free – no GitHub
  - Output can be C or HDL
- Manual Approach
  - The real amount of work makes this approach unfeasible
  - Doubts the output provide better code than automated approach
  - Any change in architecture complicated
- Linux + (C)Python

Selected solution

- TensorFlow Lite for Microcontrollers
  - https://github.com/tensorflow/tflite-micro
  - Good TF support
  - Possible in-orbit update
  - Internally use FlatBuffers
  - No big-endian architecture support (Leon Sparc is BE)
  - It is necessary byte-swap SOME bytes in FlatBuffer
  - The version of TF and TFLM must match
  - Hard to port any patch between TF/TFLM versions

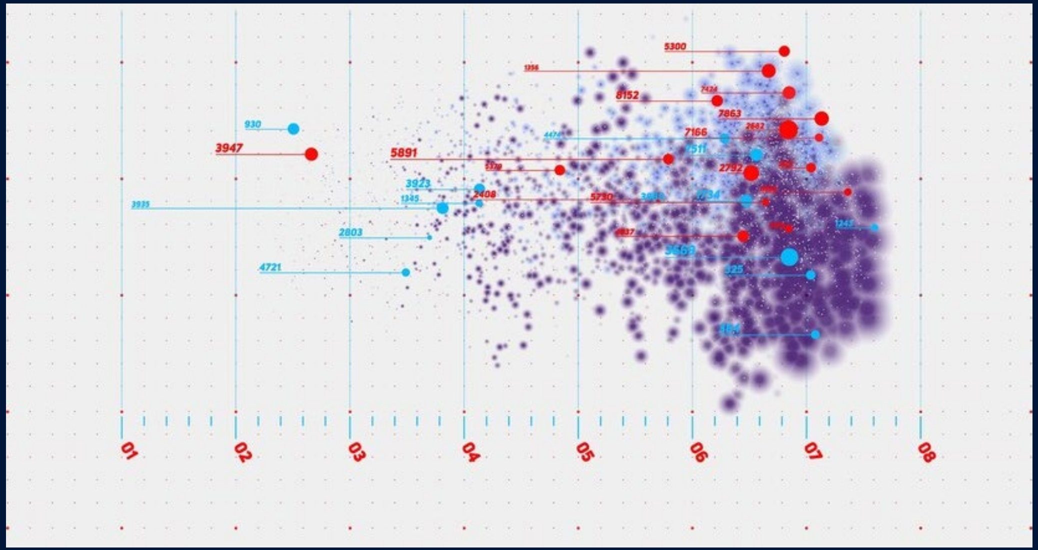


Quantization, range tuning

- Model training in float/double
- Execution in int32/int16/int8 faster (SIMD, reduce memory bandwidth)
- Dynamic range limited, quantize coefficients into integers

- Leon Sparc target no SIMD support => both floating-point and integer multiplications require 32-bit numbers and execute in a single cycle.

- TFLM's 8-bit integer quantization reduces precision with a 4.35% average prediction difference and increases code size by 8.18KB, while reducing the model size by 31.16% (19.36KB).



huld

Results - memory

- RAM – 20 kB
- ROM - 750 kB

Component	Size
TFLM	
baseline	11.61KB
TFLM	211.27KB
all TFLM kernels	494.63KB
used TFLM kernels	124.48KB
baseline + TFLM + all kernels	717.51KB
baseline + TFLM + used kernels	347.36KB
Models	
CNN encoder	62.13KB
KNN code	1.34KB
KNN - 1 000 points	39.01KB
KNN - 5 000 points	195.31KB
KNN - 10 000 points	390.62KB
Others	
main file	7.05KB
Full inference code	
using KNN with 1 000 point	393.42KB
using KNN with 5 000 point	549.72KB
using KNN with 10 000 point	745.03KB

huld

Results - speed

- Total inference time 75 ms

Action	Time
TFLM	
Setup	39.75ms
Passing data to interpreter	0.15ms
Encoder inference	31.58ms
KNN	
KNN with 10 000 points	43.59ms
KNN with 5 000 points	20.61ms
KNN with 1 000 points	3.66ms

AI FDIR V&V

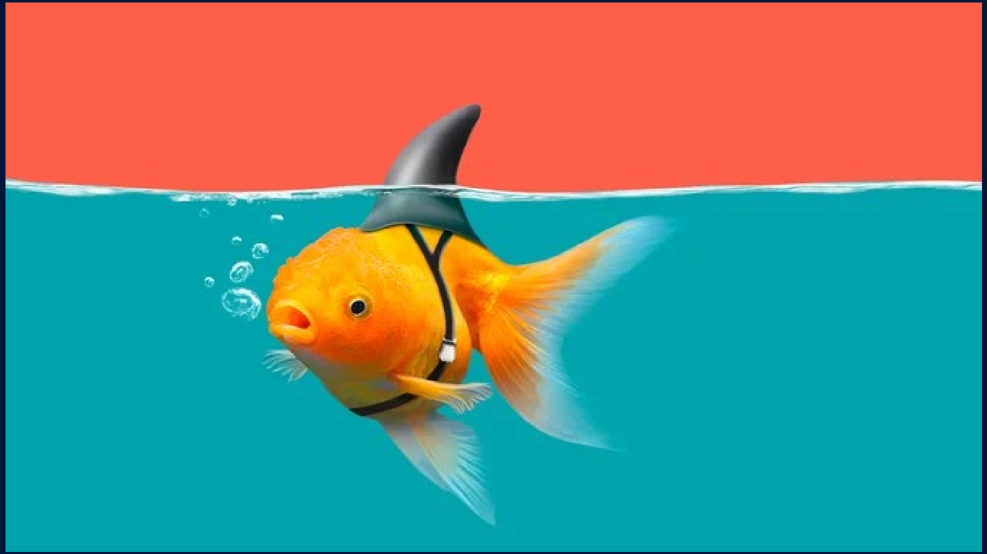
huld

huld

KPI measurements

- Similar problem mentioned in data section - failure event usually missing
- Bias in input data – 99% of events does not contain “failure event”

True positive 1.0 can be achieved simple way – report event all the time – but this not what we want



Integration V&V

- Model after integration behavioral match Python prototype
- Usually 100% match is not possible due to
    - Quantization
    - Double / float / integers
    - Even two versions of TF produce slightly different results
    - Rounding

huld

Rest of system V&V

- The AI is 10% of the whole system
- Remaining 90% is “common” C/FPGA code responsible for execution, input prepare, check output, “FDIR” etc.
- ECSS standards

- Performance (execution time) measurement
- Memory bandwidth can be limitation

Summary

huld

- KPI – much better than assumed at the beginning of the project
- The inference time within limits as well

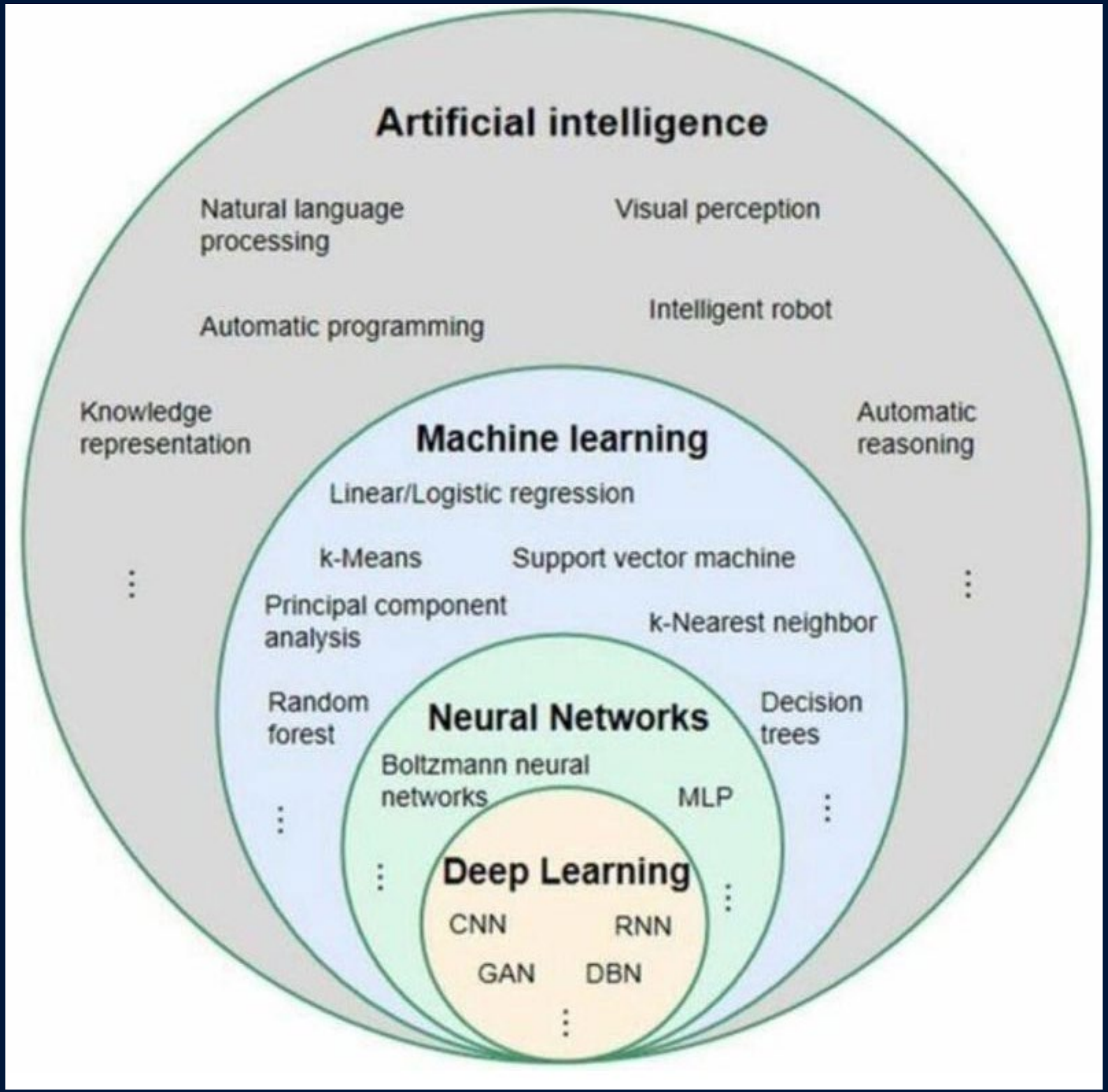
- Model suitable to run in single core of Leon Sparc without any modification, no GPU, no FPGA, no additional “acceleration card”
- Emphasis on early detection of anomalies – while they are small and before they grow and become a problem

- Benefits of running on spacecraft

Details published “ON BOARD TELEMETRY ANOMALY DETECTION USING MACHINE LEARNING, BiDS 2023”

Questions

- Industry adoption curve
- Can it reduce cost?
- Data availability
- AI (FDIR) V&V approach
- Transfer learning
- Benefits to traditional approach



huld

Beyond tomorrow