The On-board AI Research Platform

Enabling Autonomy Beyond Earth

James Marshall (james.marshall-1@nasa.gov)
NASA GSFC Science Data Processing (587)
Smith College Invited Talk
March 2nd, 2023

NASA Goddard SPACE FLIGHT CENTER

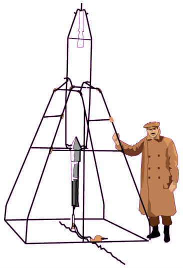
About Us

James Marshall

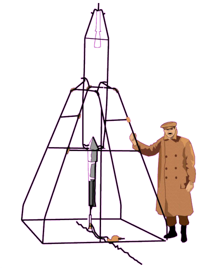
Senior Computer Engineer, NASA Goddard Space Flight Center
PhD in Computer Science, George Washington University
Started working on CubeSats as an intern (https://intern.nasa.gov/)
Entered NASA Pathways program while at George Washington
Software lead for an in-house processor card
Interests: Sci-Fi

Evana Gizzi

Artificial Intelligence Researcher/Mission Resilience Lead NASA
PhD in Artificial Intelligence, Tufts University
Born and raised in Massachusetts (NASA duty station: Lowell MA)
Interests: Not Sci-Fi (Never seen Star Wars)



Outline



- Motivating problem: New Horizons Anomaly
- Background
- The On-board Artificial Intelligence Research (OnAIR) Platform
- Previous Work
- Ongoing Work
- Demonstration
- Conclusion
- Acknowledgements/References

Motivating Problem: New Horizons Anomaly

New Horizons

Flyby of Pluto in 2015

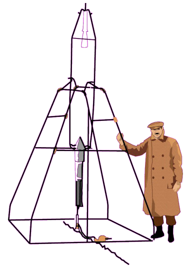
NASA / Applied Physics Laboratory
TODO

Pluto's Heart

Why was the mission a flyby?

Image Credit: NASA (https://www.nasa.gov/feature/new-horizons-top-10-pluto-pics)

New Horizons, Background



Spacecraft telemetry points:

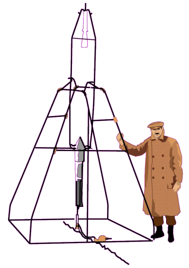
- Temperature, current, voltage, pressure, etc.
- Each point may have associated limits
- Example: processor temperature limits **-10**, **10**, **60**, **80**

Safe Mode: spacecraft maintains only critical functionality such as stabilization, and communication with ground

New Horizons, The Anomaly

An anomaly occurred 10 days prior to arrival

Automatically detected, switched to “safe mode”, re-established contact



New Horizons, The Anomaly

An anomaly occurred 10 days prior to arrival

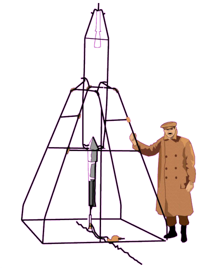
Automatically detected, switched to “safe mode”, re-established contact

9 hour round trip communication time (~3 billion miles / 4.9 billion km from Earth)

Spacecraft was recovered with no impact to science mission

Cause was a “hard-to-detect timing flaw” that occurred during the flyby preparations

New Horizons, The Anomaly



An anomaly occurred 10 days prior to arrival

Automatically detected, switched to “safe mode”, re-established contact

9 hour round trip communication time (~3 billion miles / 4.9 million km from Earth)

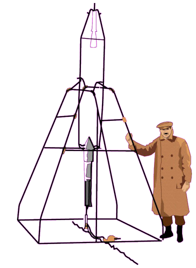
Spacecraft was recovered with no impact to science mission

Cause was a “hard-to-detect timing flaw” that occurred during the flyby preparations

Waiting for ground intervention is not always feasible!

Sources: https://www.nasa.gov/nh/new-horizons-responds-spacecraft-anomaly, https://www.nasa.gov/nh/new-horizons-plans-july-7-return-to-normal-science-operations

Onboard AI is Becoming Necessary



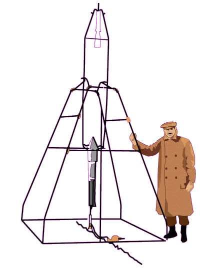
Fundamentally, we can not overcome:

- Latency: limited to the speed of light
- Bandwidth: (very roughly) inverse square law

Must deal with faults, including unanticipated faults

Need to process data onboard as science instruments collect more and more data

Onboard AI is Becoming Necessary



Fundamentally, we can not overcome:

- latency: limited to the speed of light
- bandwidth: (very roughly) inverse square law

Must deal with faults, including unanticipated faults

Need to process data onboard as science instruments collect more and more data

Spacecraft need tools to increase their autonomy for both fault recovery and science data processing.

Background

Embedded Systems

Does anyone have experience with Embedded Systems?

Any examples?

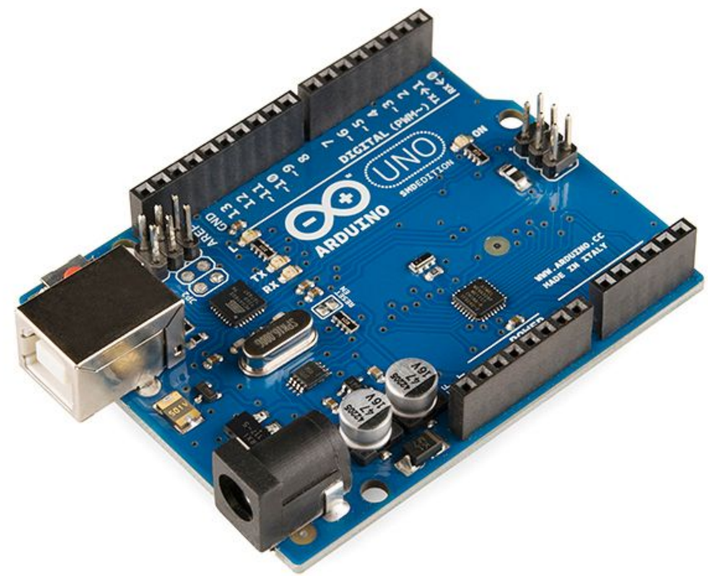


Image credit: SparkFunElectronics under the CC-BY-2.0 license
https://commons.wikimedia.org/wiki/File:Arduino_Uno_-_R3.jpg

Embedded Systems

Does anyone have experience with Embedded Systems?

Any examples?

Embedded systems typically have several constraints:

- Limited resources
- Real-time constraints
- Limited operating system and libraries

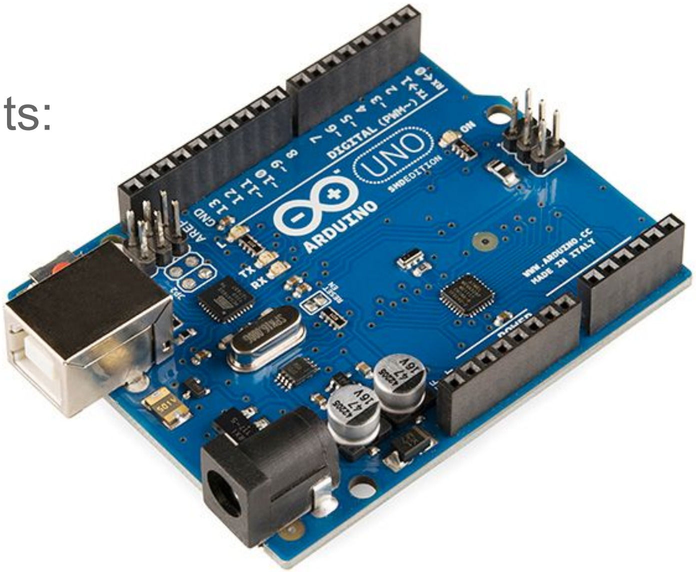
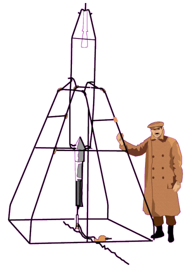


Image credit: SparkFunElectronics under the CC-BY-2.0 license
https://commons.wikimedia.org/wiki/File:Arduino_Uno_-_R3.jpg

Flight Software



Embedded software with more rules

Flight software challenges:

- One-off missions
- High cost of failure
- Risk averse
- Restricted use of C
- Thread-based
- Intensive processes to reduce risk: [NASA Procedural Requirement 7150.2D (NASA Software Engineering Requirements)]

NASA's core Flight System (cFS)

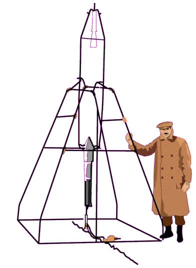
Flight software framework

Used in the Magnetospheric Multiscale (MMS) mission

Open Source!
https://github.com/nasa/cfs

Mailing list:
https://github.com/nasa/cfs#join-the-mailing-list

Image Credit: NASA/Ben Smegelsky (https://www.nasa.gov/content/magnetospheric-multiscale-observatories-processed-for-launch)



What cFS Provides

Platform Abstraction

Core Services:

- Tables, Events, Time, Software Bus (message passing)

Common Applications:

- Command In, Telemetry Out, Stored Command, House Keeping…

Framework for Libraries and Custom Applications

- Mission and hardware specific code is isolated, promotes reusability

Flight heritage and a community

Typical cFS Lollipop Diagram

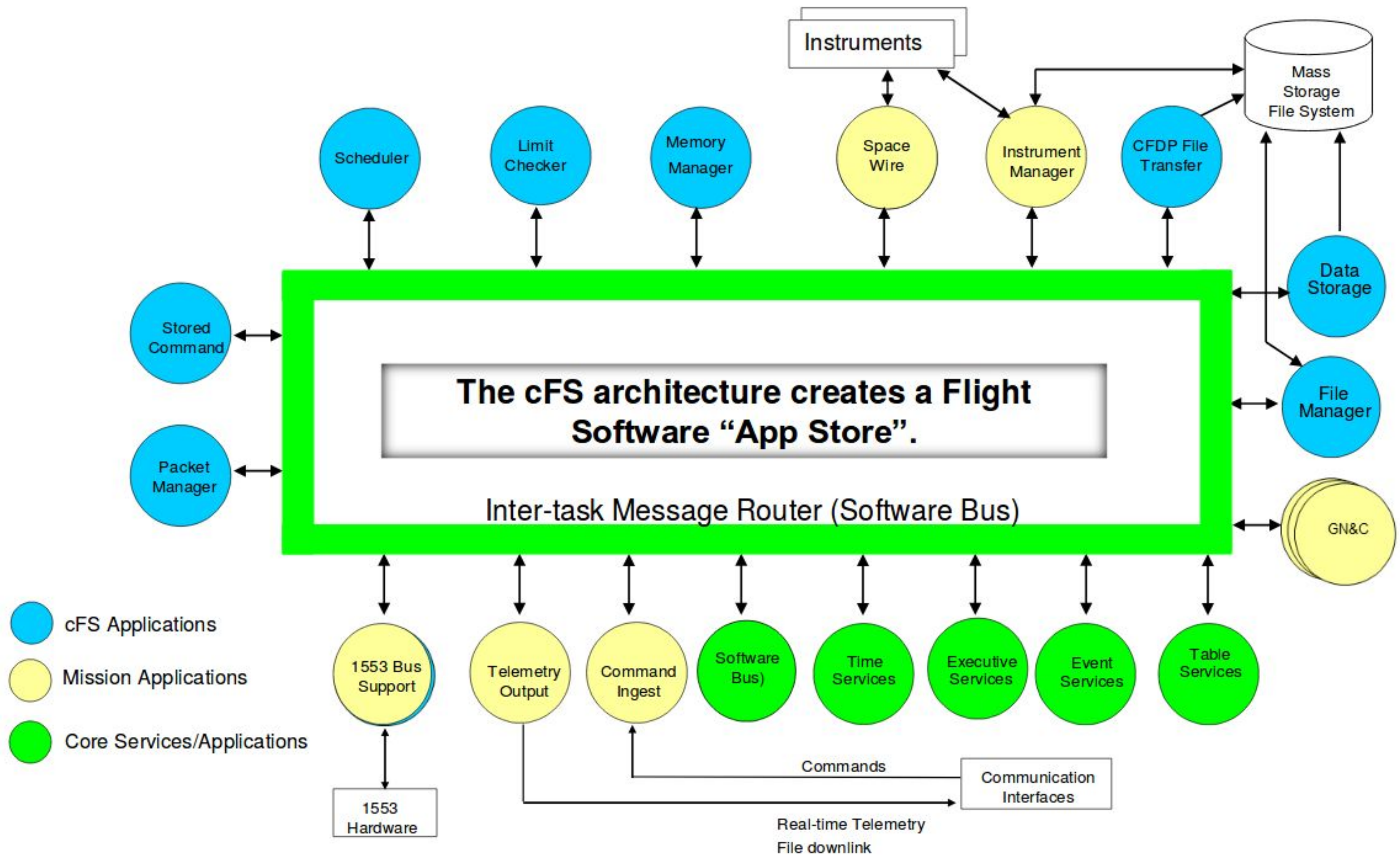
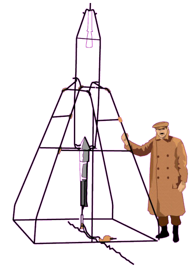


Image credit: NASA GSFC https://cfs.gsfc.nasa.gov/Features.html

cFS for AI Development



Does cFS have what we need for AI Development?

Requirement	Provided by cFS
Live telemetry, science data	Yes!
High-level languages	C, C++... No Python
Latest libraries	No
Process isolation	No
Powerful platforms	Yes!

The On-board Artificial Intelligence Research Platform

The On-board AI Research (OnAIR) Platform

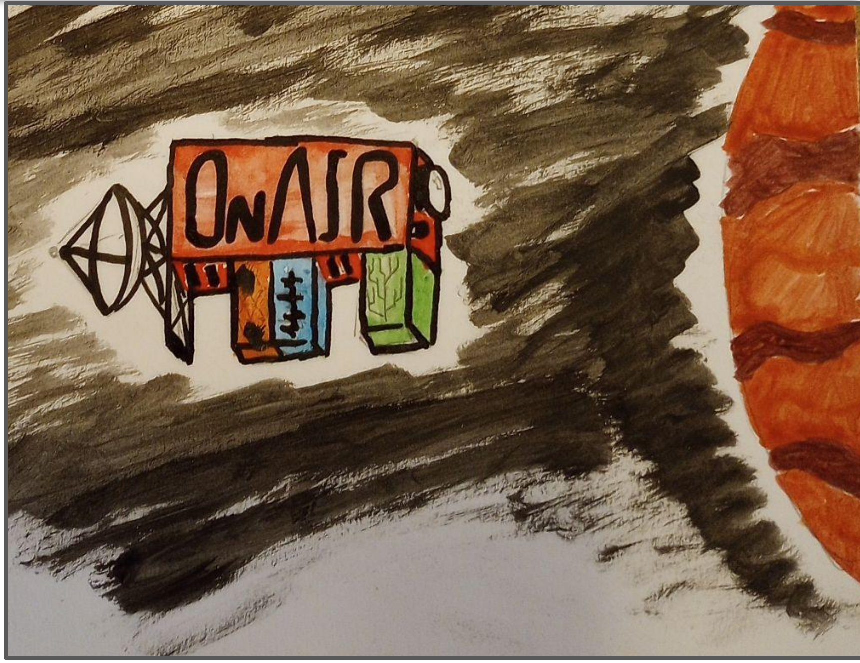
Extends cFS with multi-process support[1]

Supports Python

In process of Open Sourcing

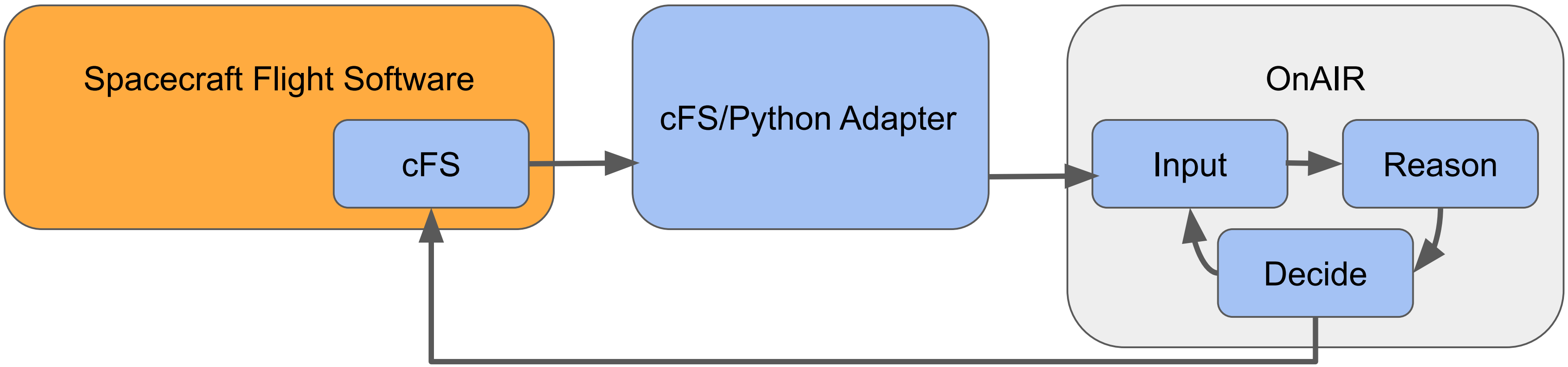
Target hardware[3]:

- SpaceCube 3 Mini (~100 MHz softcores)
- Mini-Z (2 667MHz ARM Cortex-A9)



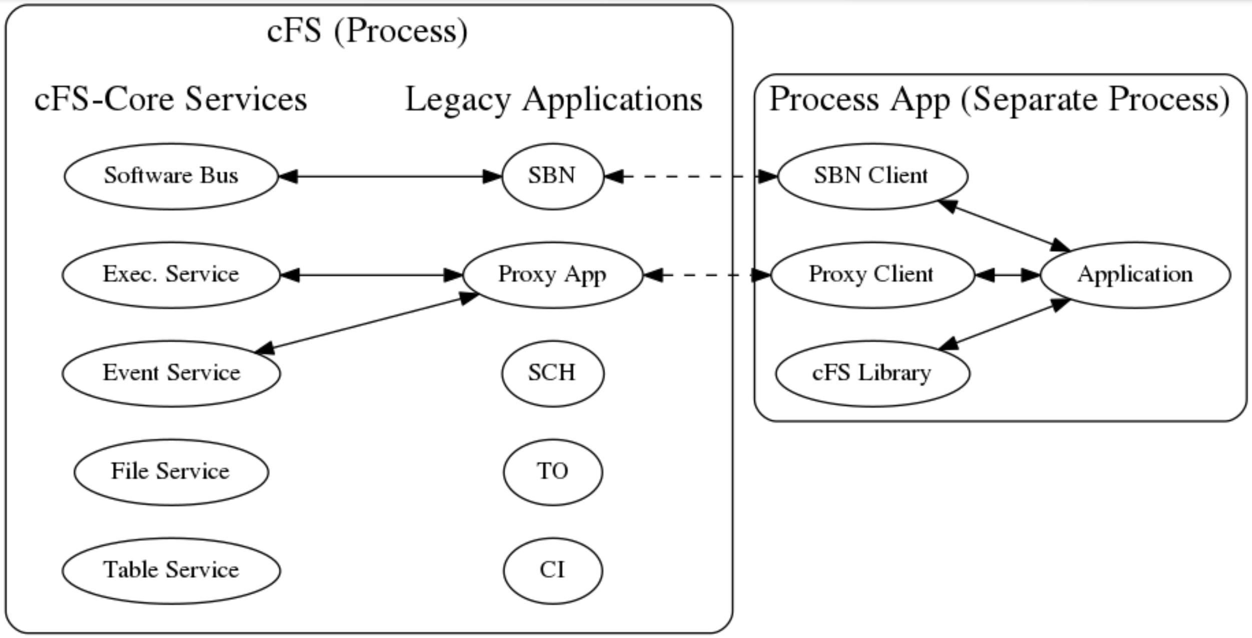
This is our logo until Evana gets one from the graphics folks. Credit: James Marshall

OnAIR Architecture



- Ingests data from live spacecraft telemetry
- Reasoning and decision making will use user-supplied plugins
- Expanding to support multi-agent systems
- Demonstrated working with cFS in a simulated environment (NOS3 and 42)

OnAIR Architecture: cFS Adapter



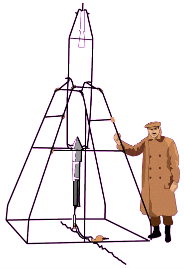
A cFS application launches a separate process for OnAIR

The Software Bus is accessed through the Software Bus Network and client

Other cFS services are accessed through remote procedure calls

cFS is also provided as a library

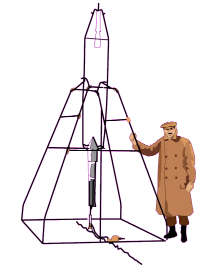
cFS with OnAIR



cFS with OnAIR provides the fundamentals for AI development

Requirement	Provided by cFS	Provided by OnAIR
Live telemetry, science data	Yes	
High-level languages	C, C++	Python
Latest libraries		Yes!
Process isolation		Yes!
Powerful platforms	Yes!	

Previous Work



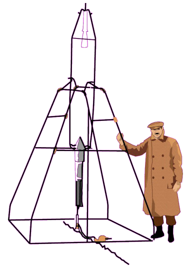
Previous Work: Fault Diagnosis

Fault detection: did a fault occur?

- Examples: current limit exceeded, hardware not responding

Fault diagnosis: why did a fault occur?

- Example: an electrical short caused a current and temperature increase



Previous Work: Fault Diagnosis

Fault detection: did a fault occur?

- Examples: current limit exceeded, hardware not responding

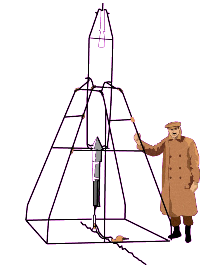
Fault diagnosis: why did a fault occur?

- Example: an electrical short caused a current and temperature increase

Initial work focused on diagnosis, assumed faults were detected by telemetry exceeding limits

Data Sets

Data	Missions	Frames	Mnemonics
Toy	11	1440	8
Simulated	10	650-2000	17
Real Sounding Rocket	1	6500	33



Data Sets

Data	Missions	Frames	Mnemonics
Toy	11	1440	8
Simulated	10	650-2000	17
Real Sounding Rocket	1	6500	33

Definitions:

- Data point: a single value such as 10.8
- Mnemonic: a named series of data points over time such as “CPU Temperature = 10.8, 12.3, 15”
- Frame: a data point for each mnemonic, represents a single time step.

    CPU Temperature = 10.8
    5 Volt Bus Current = 2.3
    12 Volt Bus Current = 1.2

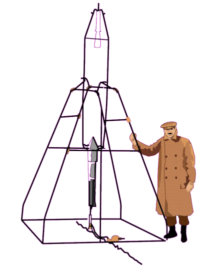
Problem Formulation

Given: the faulting mnemonic

The mnemonic that tripped the fault detection system.

Want: root cause mnemonic

The mnemonic most closely related to the cause of the fault.



Problem Formulation

Given: the faulting mnemonic

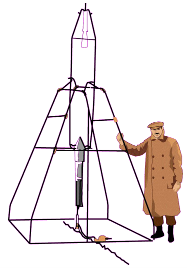
The mnemonic that tripped the fault detection system.

Want: root cause mnemonic

The mnemonic most closely related to the cause of the fault.

Possibilities:

Faulting == Root cause	Root cause is discoverable	Root cause is NOT discoverable



Strategy

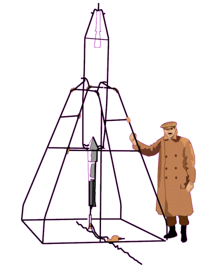
The data alone has information about the system

Apply algorithms to extract that information

Different algorithms will yield different insights

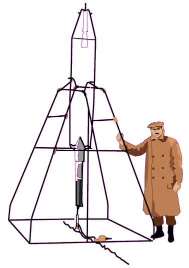
Hypothesis: We can combine these insights to diagnose a fault

Ensemble Algorithm[2]



Three algorithms and their insights:

1. Kalman - how a data point differs from its individual past
2. Autoencoder - how a mnemonic differs relative to other mnemonics
3. Causality - the relationships between all mnemonics to one another



Ensemble Algorithm[2]

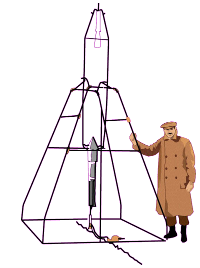
Three algorithms and their insights:

1. Kalman - how a data point differs from its individual past
2. Autoencoder - how a mnemonic differs relative to other mnemonics
3. Causality - the relationships between all mnemonics to one another

If there is a single faulting mnemonic AND that mnemonic contributes the most to the Autoencoder’s reconstruction error -> the root error is the faulting mnemonic itself.

Example: a sensor breaks and sends faulty data

Ensemble Algorithm[2]



Three algorithms and their insights:

1. Kalman - how a data point differs from its individual past
2. Autoencoder - how a mnemonic differs relative to other mnemonics
3. Causality - the relationships between all mnemonics to one another

If the faulting mnemonic is not the root cause, it may be a symptom

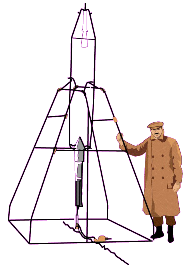
The causality algorithm gives us a list of the mnemonics most related to the symptom; if any of those are breaking their kalman filter then that is the root cause.

Example: a temperature on the skin of a sounding rocket surpasses its limit. The root cause is traced to the airspeed.

Results: Accuracy

How often was the root cause mnemonic discovered?

Multiple experiment runs per mission



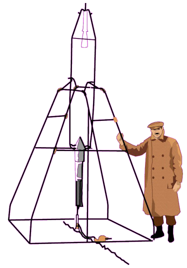
Results: Accuracy

How often was the root cause mnemonic discovered?

Multiple experiment runs per mission

Data	Missions	Accuracy
Toy	11	77%
Simulated	10	70%
Real Sounding Rocket	1	0%

Results: Accuracy

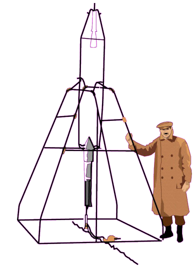


How often was the root cause mnemonic discovered?

Multiple experiment runs per mission

Data	Missions	Accuracy
Toy	11	77%
Simulated	10	70%
Real Sounding Rocket	1	0%

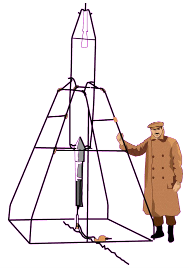
Sounding Rocket: consistently diagnosed another symptomatic mnemonic. The root cause mnemonic was behaving erratically.



Results: Comparison

Implemented other common algorithms:

- Proximal Policy Optimization (PPO)
- Standalone Autoencoder (AE)

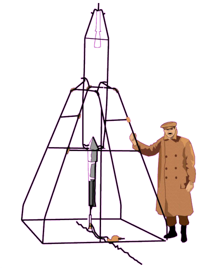


Results: Comparison

Implemented other common algorithms:

- Proximal Policy Optimization (PPO)
- Standalone Autoencoder (AE)

Data	Missions	Ensemble Acc.	PPO Acc.	AE Acc.
Toy	11	77%	4%	60%
Simulated	10	70%	0%	53%
Real Sounding Rocket	1	0%	N/A	N/A



Results: Comparison

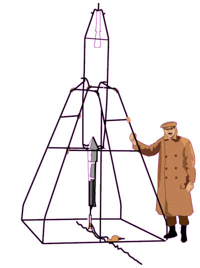
Implemented other common algorithms:

- Proximal Policy Optimization (PPO)
- Standalone Autoencoder (AE)

Data	Missions	Ensemble Acc.	PPO Acc.	AE Acc.
Toy	11	77%	4%	60%
Simulated	10	70%	0%	53%
Real Sounding Rocket	1	0%	N/A	N/A

PPO and AE accuracy show when the root cause mnemonic was in the top 2 results for the algorithm

Ongoing Work



Ongoing Work

Make a usable platform for student research:

- Open source
- Data samples
- Example algorithms

Continue in-house research:

- Prototyping for a distributed satellite mission concept
- Use existing hardware…

SCENIC

SpaceCube Edge Node Intelligent Collaboration (SCENIC) Experiment[4][5]

Launching soon

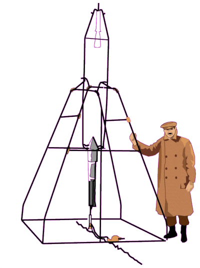
Outside of the International Space Station

Telemetry will be available summer 2023

Intend to test OnAIR after initial experiments are complete

Demonstration

Conclusion



Conclusion

Increased autonomy is necessary to increase resiliency and meet science goals

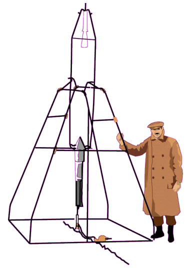
The Onboard Artificial Intelligence Research (OnAIR) Platform

- Expands the capabilities of existing flight software
- Demonstrates fault diagnosis
- Ensemble of algorithms achieves a higher accuracy

Future Plans

- Open Source
- Provide a plugin architecture for researchers
- Run experiments in space

Call to Action



NASA has a ton of open source software: https://github.com/nasa/cfs

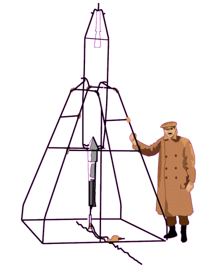
You can get started with a laptop

CubeSats are “affordable” and NASA has educational launch opportunities:

https://www.nasa.gov/directorates/heo/home/CubeSats_initiative

Email us! james.marshall-1@nasa.gov and evana.gizzi@nasa.gov

Acknowledgements and References



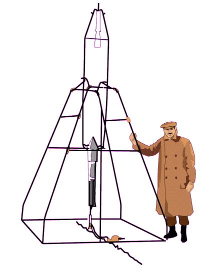
Acknowledgements

Smith College

- Smith College Lecture Committee
- Kevin Sheah
- Sarah Lanzoni
- Pablo Frank

OnAIR (formerly RAISR) Team

- Nathan Riolo (NASA WFF 589)
- Alan Gibson (NASA GSFC 587)
- Christopher Trombley
- Caroline Kuzio (NASA WFF 589)
- Ahmed Ghalib (NASA WFF 810)
- Nicholas Pellegrino
- Christopher Chapman
- Hayley Owen
- Jeffrey St. Jean
- Gabriel Raskin



References

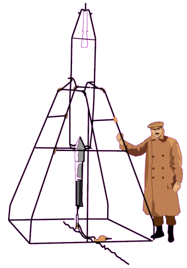
[1] Marshall, J. Multiple Process Support in cFS: Enabling Safe On-Board Processing *Flight Software Workshop*, 2021, Presentation only, online: https://www.youtube.com/watch?v=dMw9PGvBM2E

[2] Gizzi, E., Owens, H., Pellegrino, N., Trombley, C., Marshall, J., Sinapov, J. Autonomous System-Level Fault Diagnosis in Satellites Using Housekeeping Telemetry. *Proceedings of the AIAA/USU Conference on Small Satellites*, Weekend Session 1: Advanced Concepts - Research \& Academia I, SSC22-WKI-06. https://digitalcommons.usu.edu/smallsat/2022/all2022/44/

[3] Wilson, C. SpaceCube: A NASA Family of Reconfigurable Hybrid On-Board Science Data Processors *Future In-Space Operations (FISO) Working Group Seminar,* 2020 https://ntrs.nasa.gov/api/citations/20200000731/downloads/20200000731.pdf

[4] Prototype Design for NASA SpaceCube Intelligent Multi-Purpose System for Enabling Remote Sensing, Communication, and Navigation (SpaceCube IMPS): https://techport.nasa.gov/view/96769

[5] Goodwill, J., Crum, G., MacKinnon, J., Brewer, C., Monaghan, M., Wise, T., & Wilson, C. (2021, August). NASA SpaceCube Edge TPU SmallSat Card for Autonomous Operations and Onboard Science-Data Analysis. *Proceedings of the Small Satellite Conference* (No. SSC21-VII-08). AIAA.



Acronyms

Acronym	Definition
AE	Autoencoder
cFS	core Flight System
GSFC	Goddard Space Flight Center
MMS	Magnetospheric Multiscale
OnAIR	On-board Artificial Intelligence Research
PPO	Proximal Policy Optimization
RAISR	Research in Artificial Intelligence for Spacecraft Resilience
SCENIC	SpaceCube Edge Node Intelligent Collaboration
WFF	Wallops Flight Facility