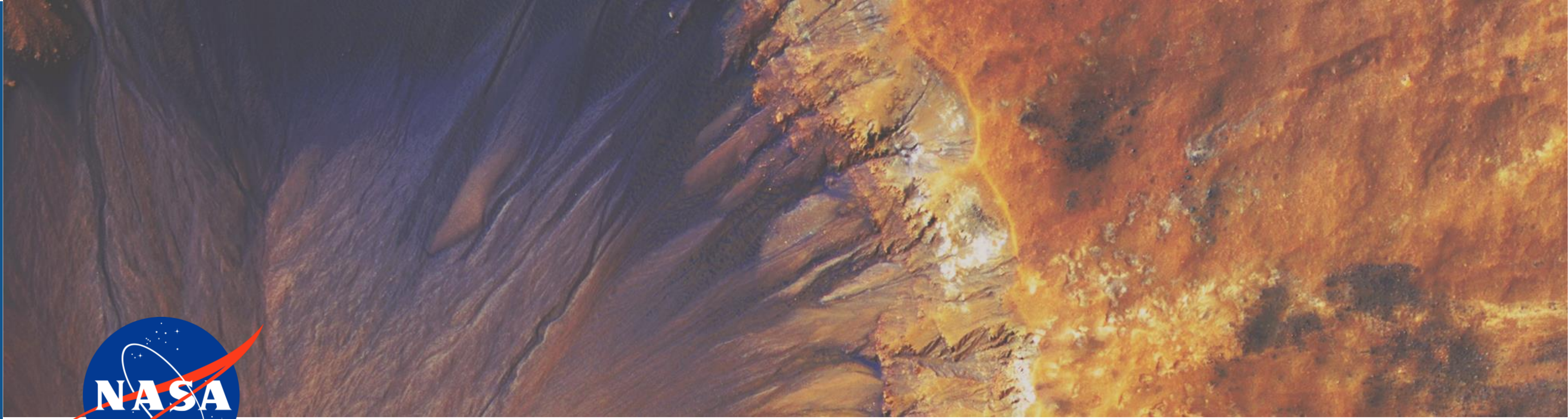


AI-enabled Autonomous Systems:
Space Power Applications

IAPG | Electrical Systems Working Group

10-12 March 2024

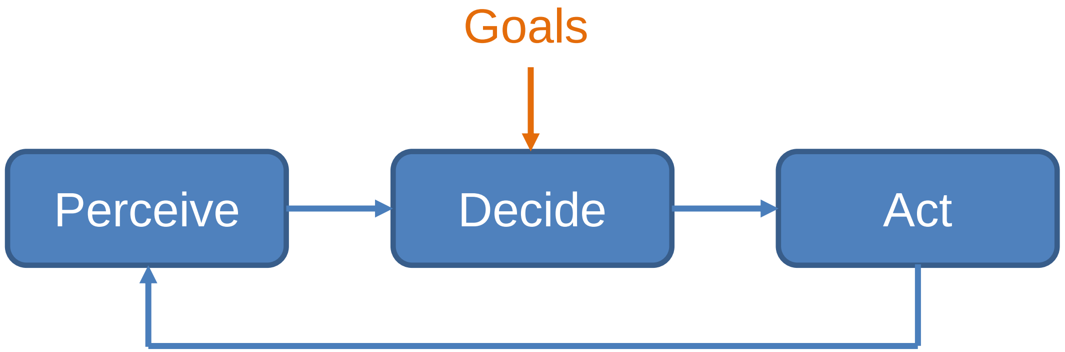
Marc Carbone | NASA GRC | Autonomous Power Control Lead

Earth Independent Operations

- Earth Independent Operations (EIO) for Mars will **deliver technologies** to provide **capabilities** to **mitigate risks** associated with reduced ground support **in time** to **meet Mars elements** and mission **needs**.

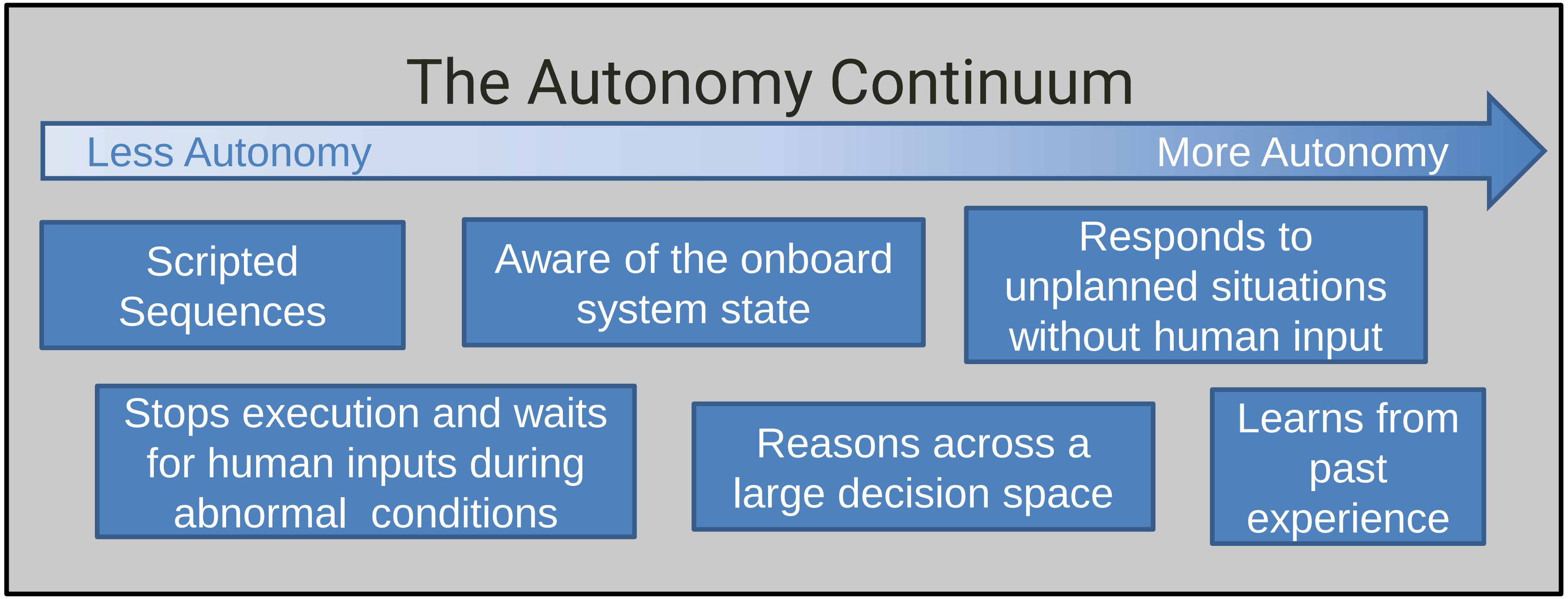
- Human Rating requirements state that Crew and Vehicle systems must be able to perform certain critical operations independent of support from ground, **<u>autonomously</u>**

What defines *Autonomy*?

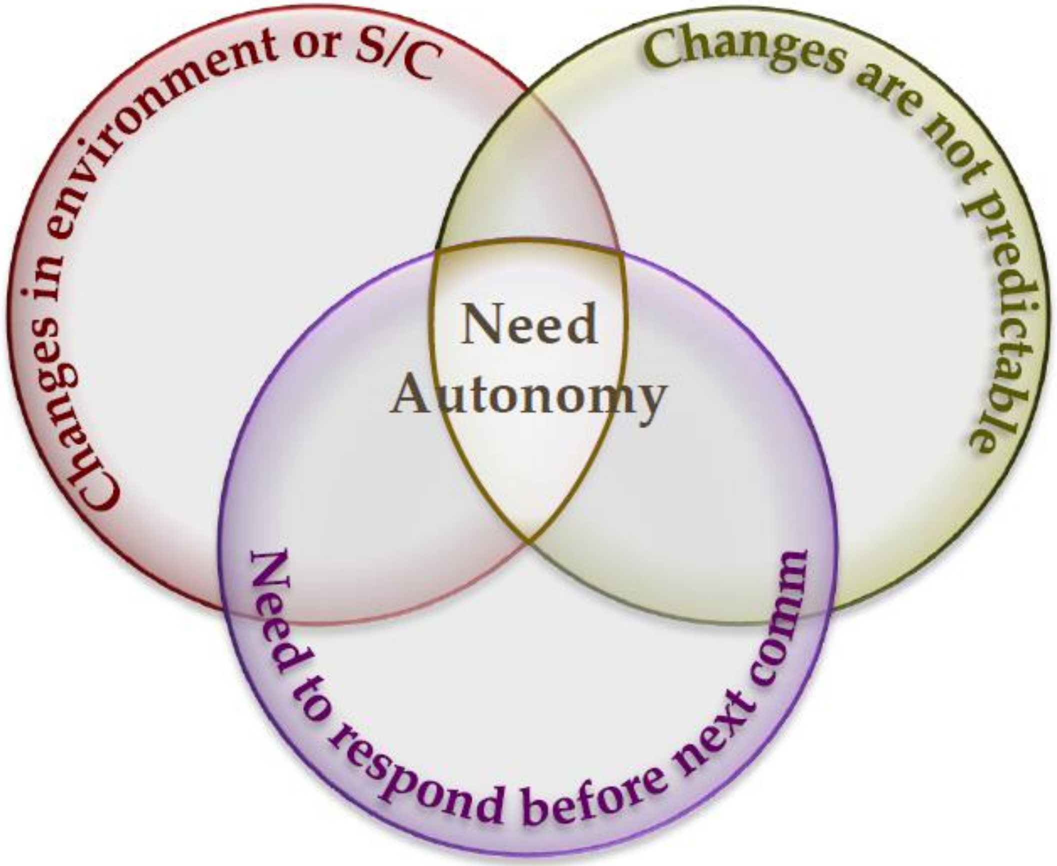


"The ability of a system to achieve goals while operating independently of external control."

– NASA System Engineering Handbook, 2018

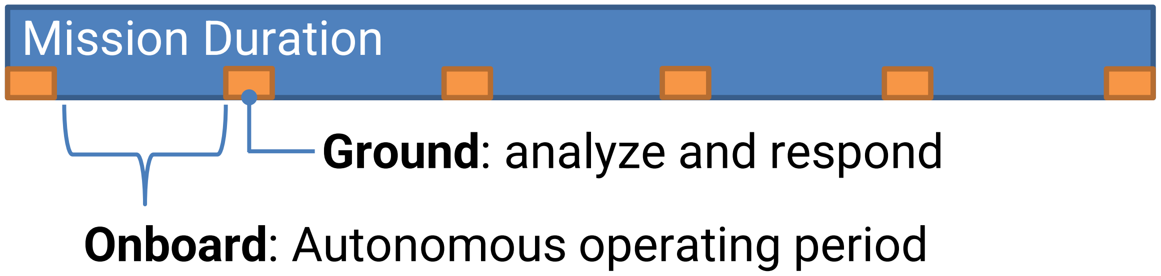


Bigger Question: *When* do we *need* autonomy?



Autonomy enables greater mission complexity and the reduction of pre-scripted controls.

Even with communication delays and outages, it is reasonable to assume that ground control will have limited supervision of any spacecraft, introducing the need for ***semi-autonomous*** operations over a given (planned or unplanned) period



Increasing Autonomy in Space

- As mission complexity increases onboard autonomous systems must perform difficult, complicated, and consequential tasks.

- In unpredictable environments they may make non-deterministic, dynamic responses to unexpected changes.

- Artificial-intelligence (AI)-enabled technologies have enabled autonomy in industrial, and academic applications.

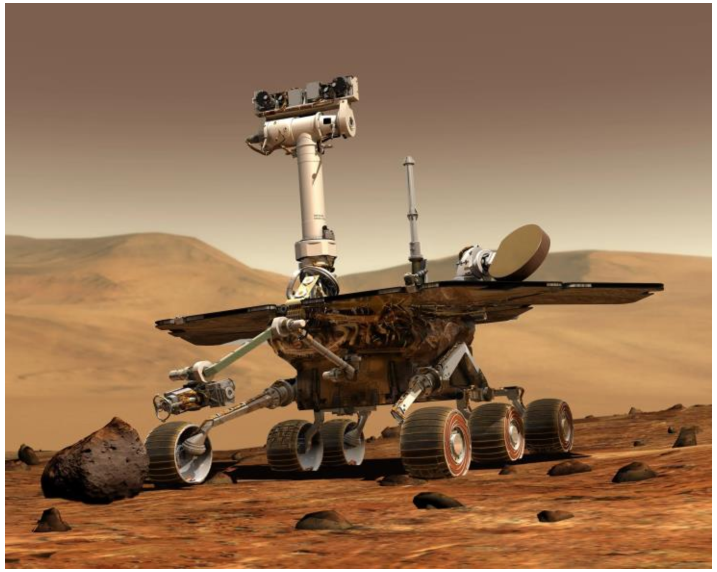
- AI may take over some decisions or perform tasks traditionally made by ground operators.

AI-enabled Autonomous Systems:
Challenges

Current State of AI & NASA Flight Software

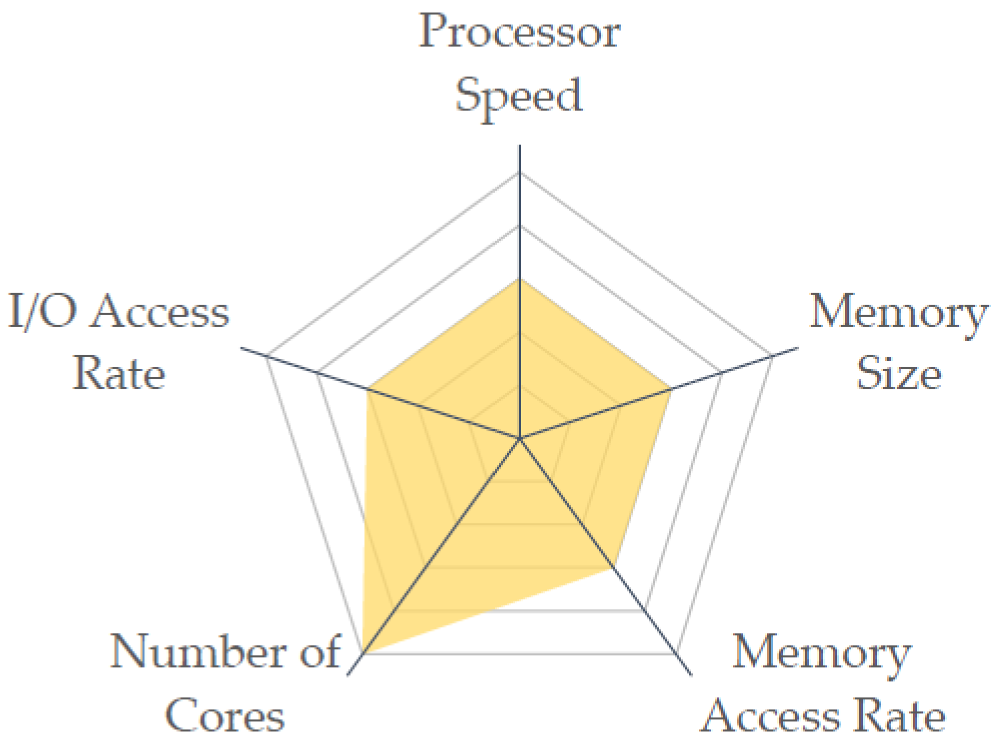
- AI/ML-based methods have been implemented on some unmanned missions
  - Self-driving on Mars rovers Opportunity and Curiosity
  - Planetary Spectrum Generator
  - SpaceX Falcon 9 landing software

- Currently, no processes exist to test and verify AI algorithms in flight software.



1. Complexity

- Complicated unpredictable environments results in a large decision-making space i.e. “state-space explosion”
- Flight computers have reduced computational abilities compared to terrestrial machines.
- AI-based systems must be carefully designed to function within the constraints of a flight computer and acquisition system



Example computational cost of a machine learning algorithm

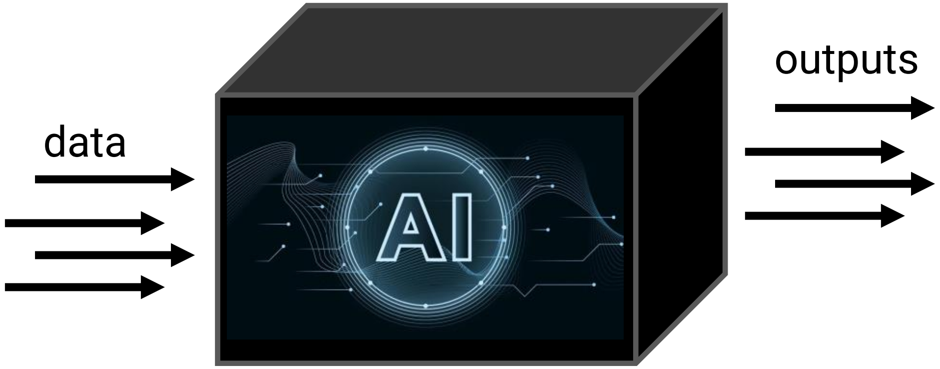
2. Lack of Tools and Standards for Testing

- Flight software requires metrics and standards for testing that support verification and validation (V&V).

- Can't test how the AI would react to every possible situation

- No way to quantify confidence in some techniques (non-deterministic)

- How to test unsupervised methods that will change ("learn") over time



3. Safety and Security Issues

- Need to ensure that the decision making is safe for astronauts onboard
- Currently no safety standards for AI on manned-flight missions
- Need to evaluate potential cybersecurity threats to AI methods

4. Integration with Humans

- Manned-missions will require some interactions between the autonomous system and the onboard crew
- Trust will need to be appropriately calibrated to ensure the crew will know when and to what extent they can rely on the autonomous system
- Methods for human-machine-teaming interfaces must be developed to increase awareness of interactions between the two

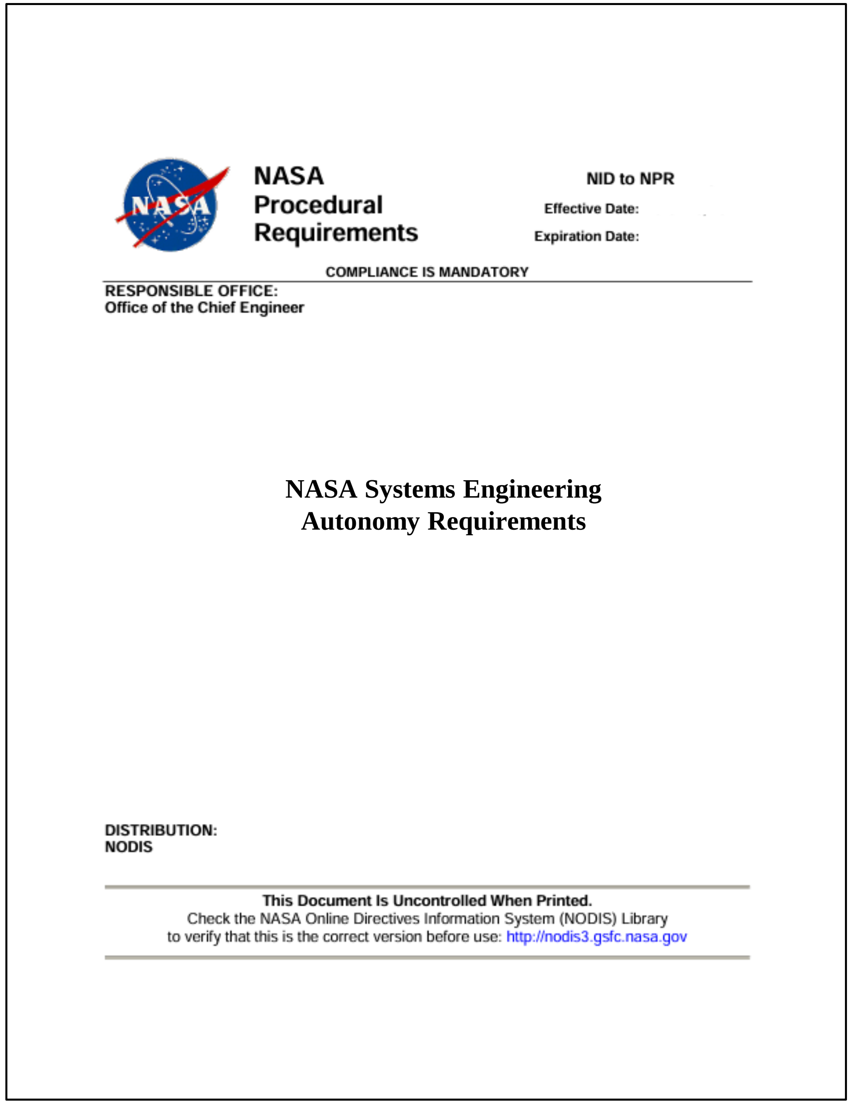
AI-enabled Autonomous Systems: Recommendations

1. Need better processes for requirements

- Designing requirements for autonomous systems can be challenging since their operational success cannot always be quantified

- Requirements for human-machine interactions need to be defined
  - Strict guidelines stating what the AI communicates the crew

- May need to transition to more *holistic* requirements rather than *technical/quantitative*



NASA Procedural Requirements

NID to NPR

Effective Date:

Expiration Date:

COMPLIANCE IS MANDATORY

RESPONSIBLE OFFICE:
Office of the Chief Engineer

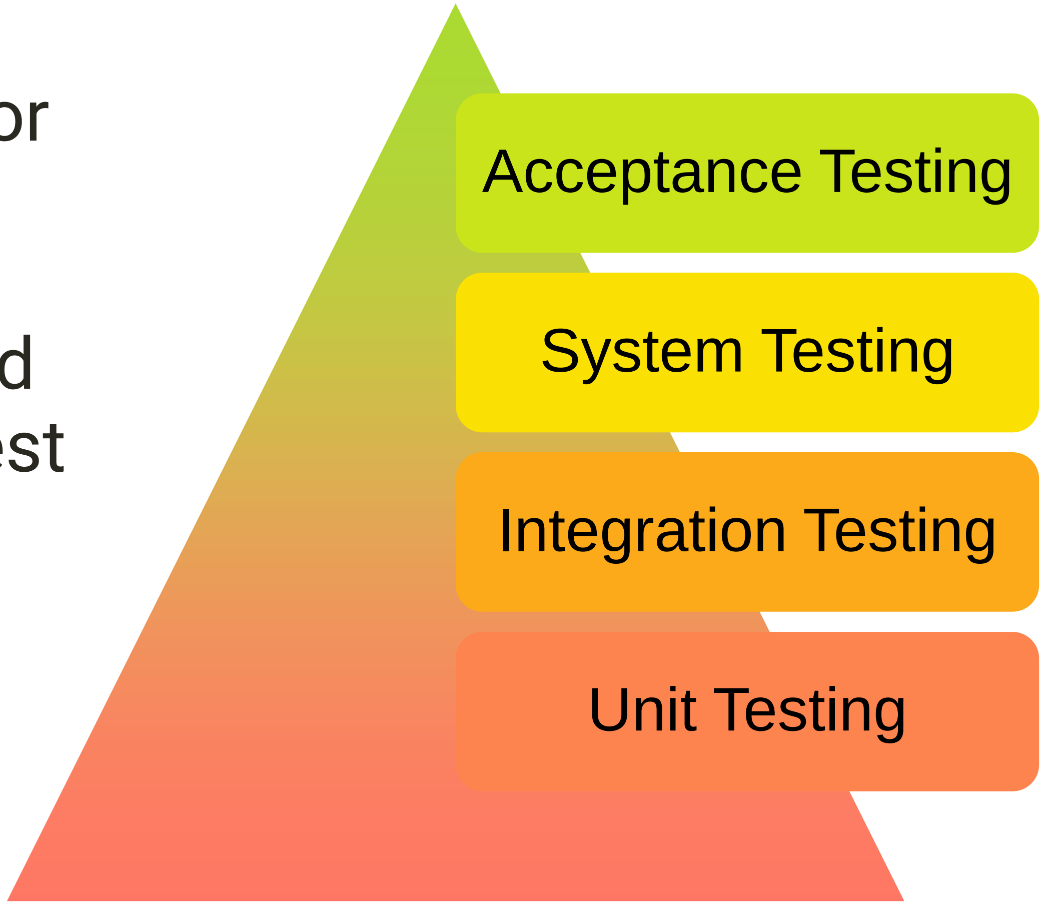
NASA Systems Engineering
Autonomy Requirements

DISTRIBUTION:
NODIS

This Document Is Uncontrolled When Printed.
Check the NASA Online Directives Information System (NODIS) Library
to verify that this is the correct version before use: http://nodis3.gsfc.nasa.gov

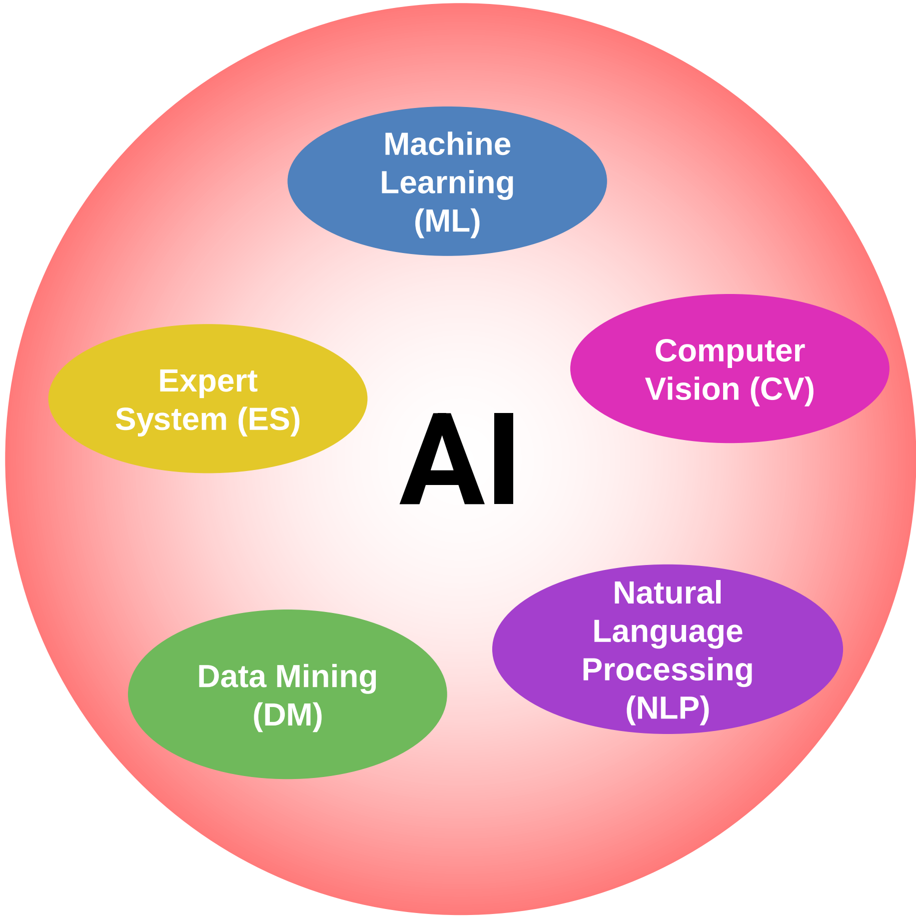
2. Use specific test frameworks

- Leverage existing testing frameworks for complex software applications
- Allow automated test tools to select and execute test points for more efficient test coverage (Monte Carlo Markov Chain)
- Leverage simulation for testing when possible



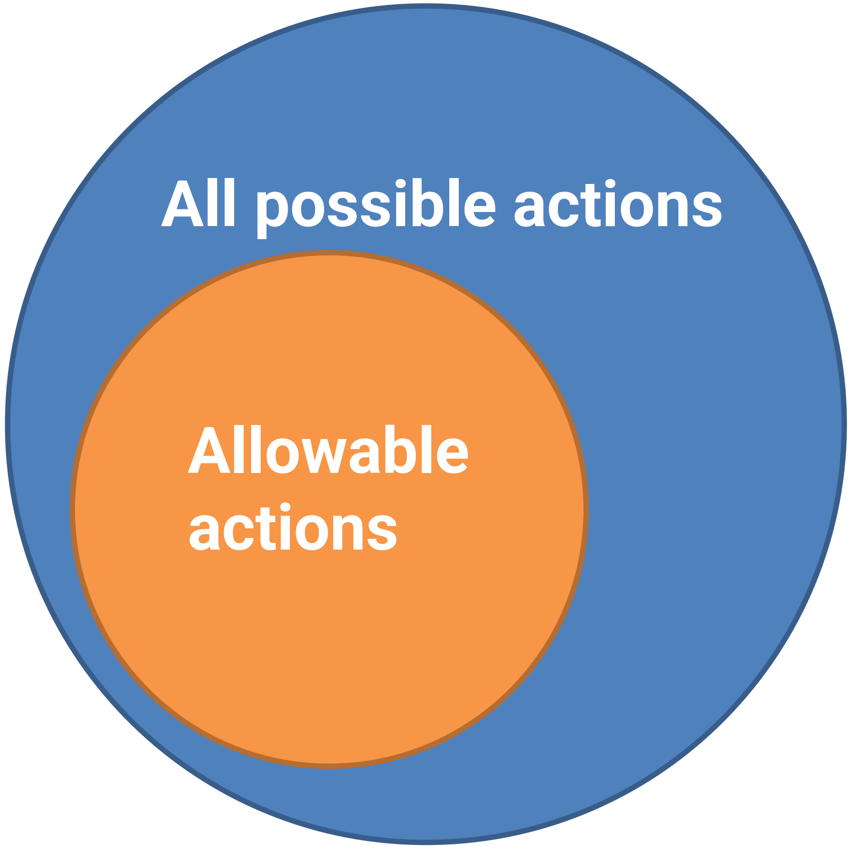
3. Invest in specific methods/tools

- Leverage **existing techniques** rather than developing new methods from scratch, when possible
- Modify existing methods to solve the **specific problems** unique to space-based automation
- Need methods for *gathering*, *storing*, and *deploying* large data sets for training
- **Partnerships** can enable common methods and tools



4. Constrain the set of decision making

- Limit the state-space problem by constraining the set of allowable decision making of the autonomous system
- Reduce testing by limiting the set of control actions (i.e. outputs) to a smaller sufficient set
- Removes uncertainty from the autonomous system



Use Case:
Autonomous Power Control

Autonomous Power Controller (APC)

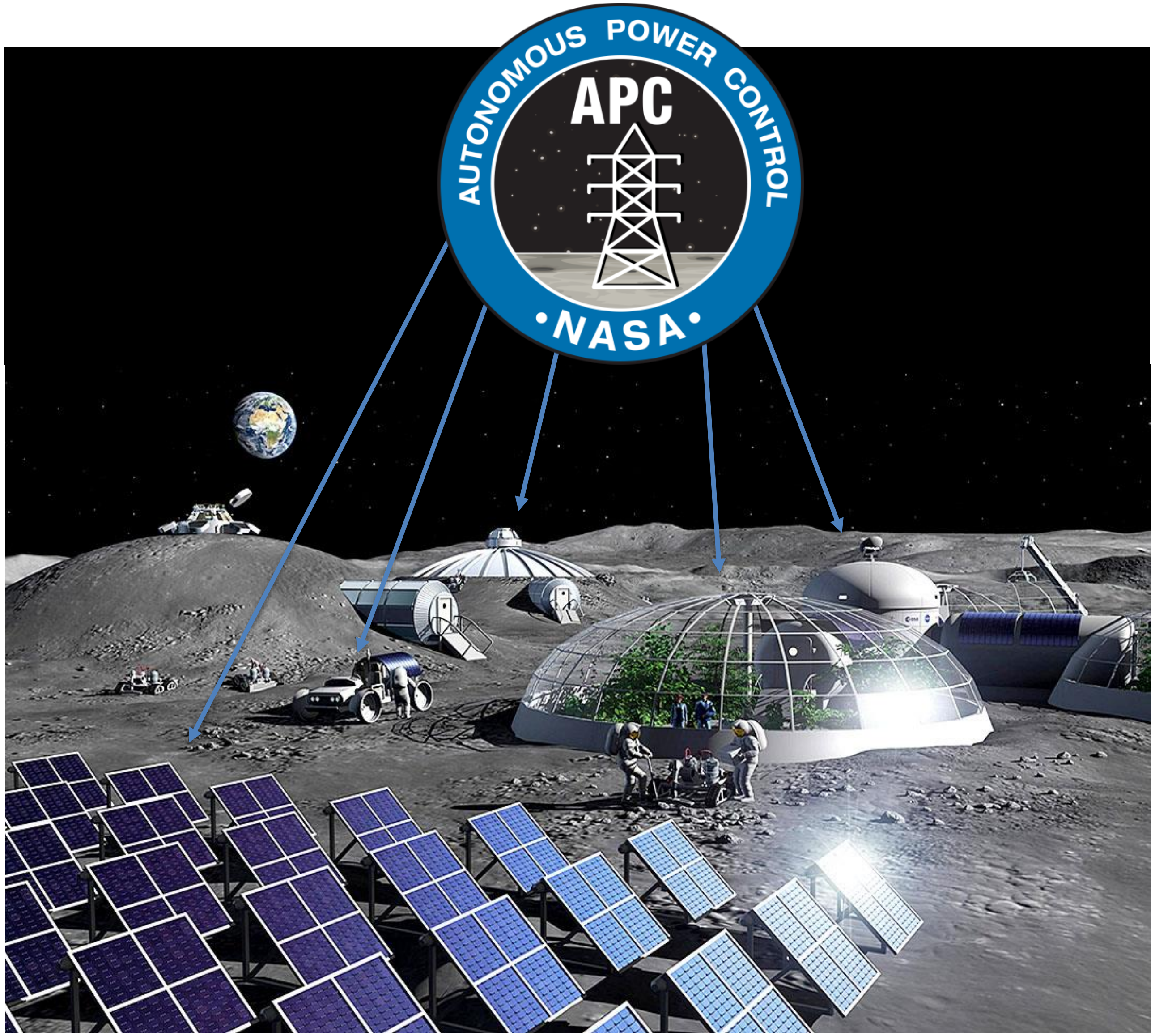
What is it?

The APC is a comprehensive Energy Management System designed to minimize the need for human interaction and oversight of electric power systems in space.

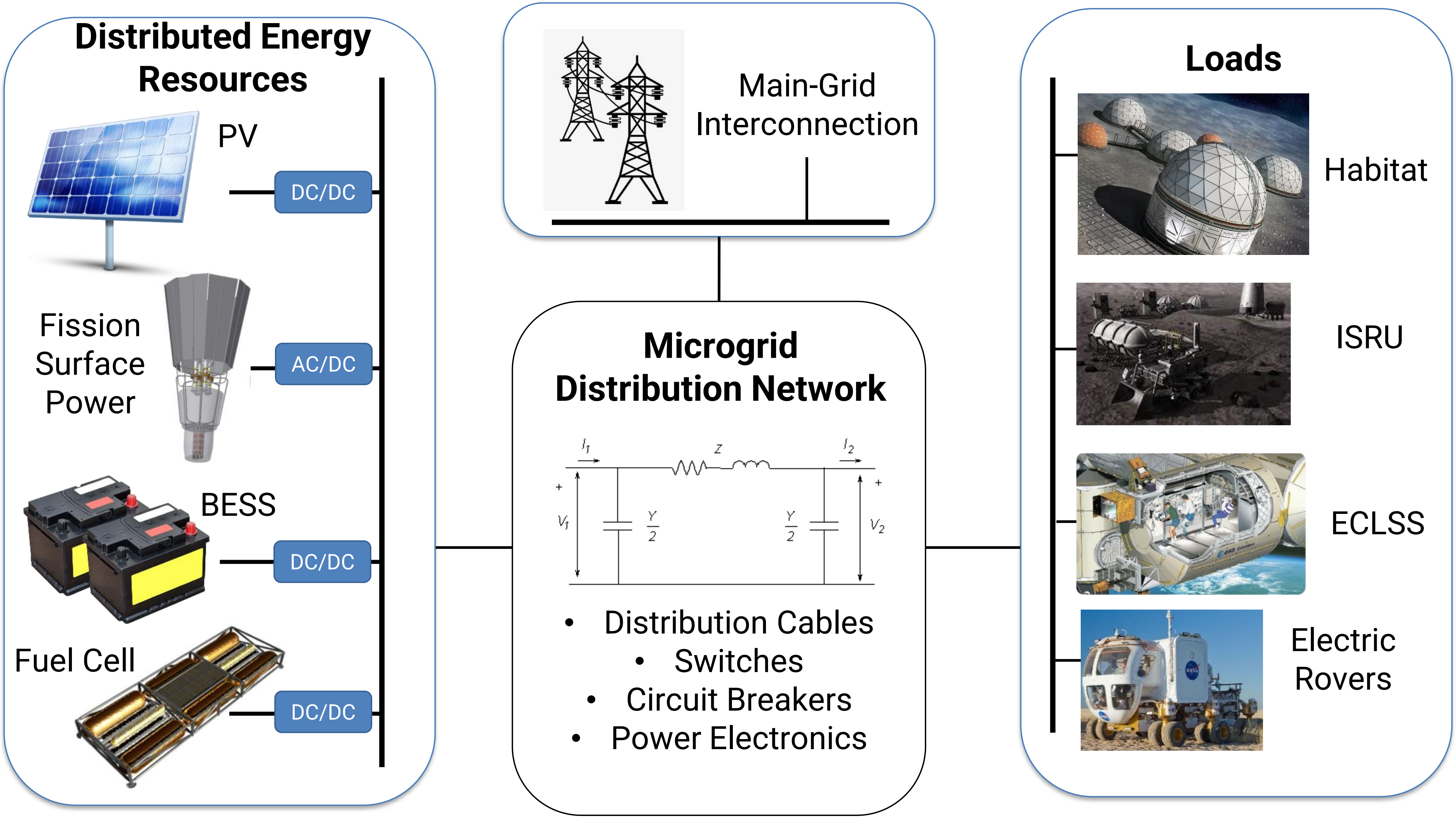
- Increase power availability, and resilience
  - Provide autonomous response to unexpected events
  - Prioritize mission critical loads

- Develop control strategies to achieve autonomy
  - Reduce the need for operator intervention
  - Quickly react to unplanned outages & failures

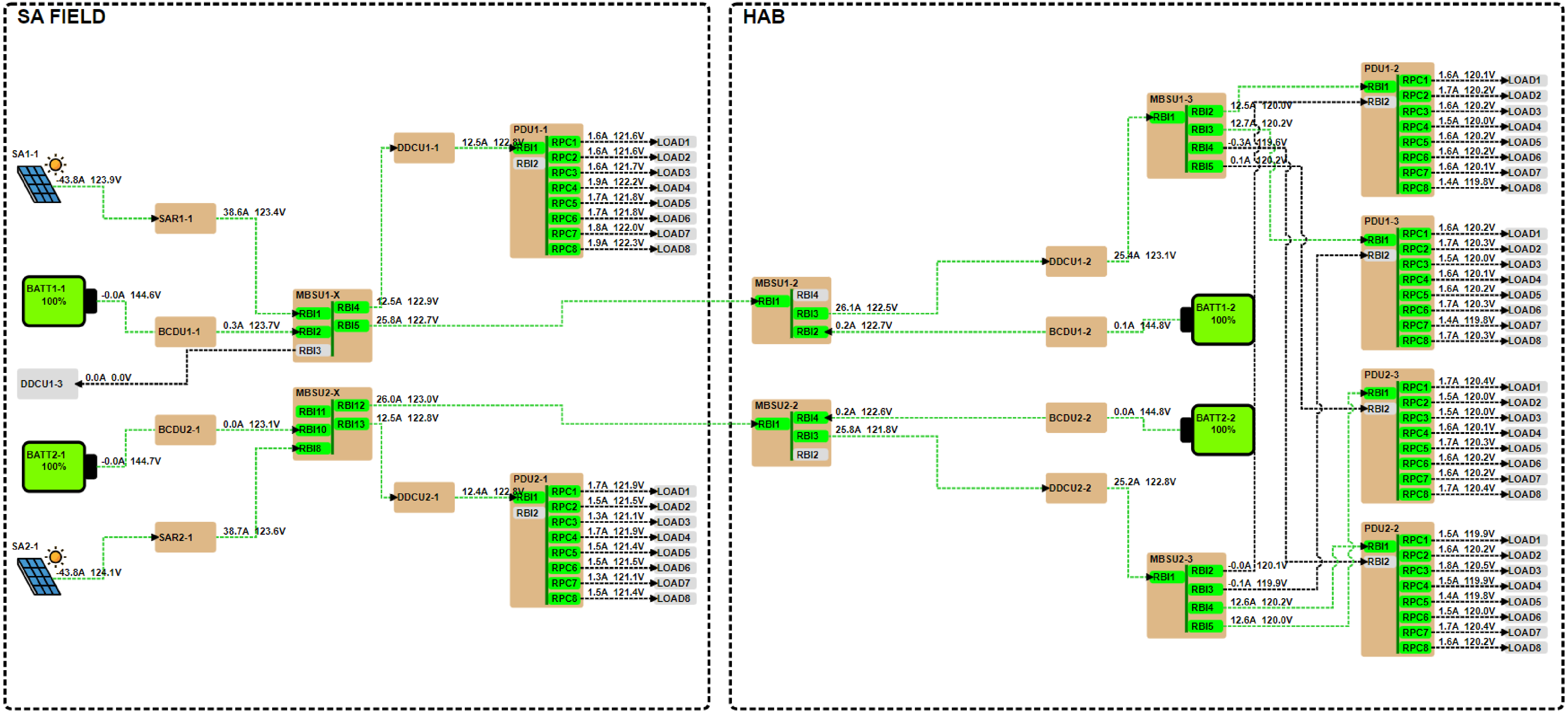
- Increase interoperability
  - System agnostic controller
  - Enable power system growth over time
  - Introduce plug-n-play components



Lunar Habitat Architecture



Lunar Habitat Mock Architecture



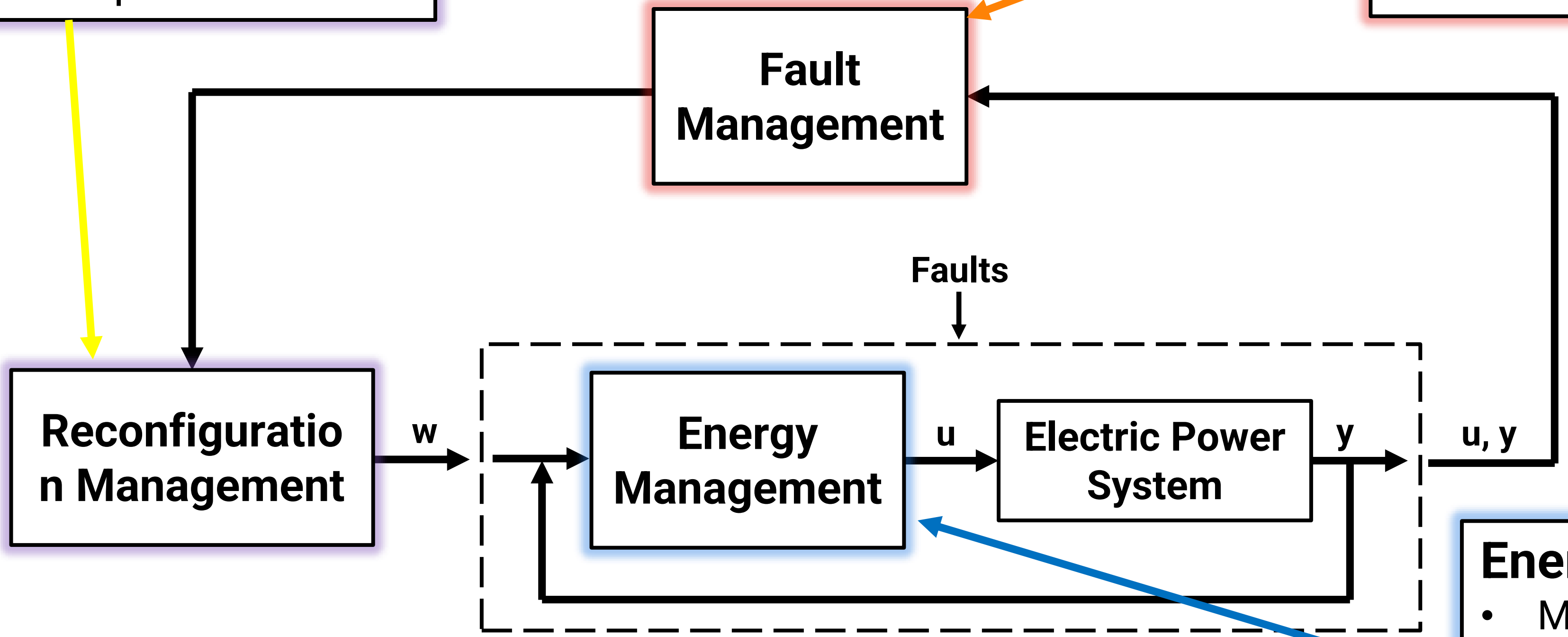
APC Block Diagram

Reconfiguration Management:

- Determine corrective actions after a fault
- Provide maintenance shutdown/startup services

Fault Management:

- Detect, isolate, and diagnose system faults
- Identify the fault type, location, and magnitude of the fault



Autonomous Power Controller Functional Block Diagram

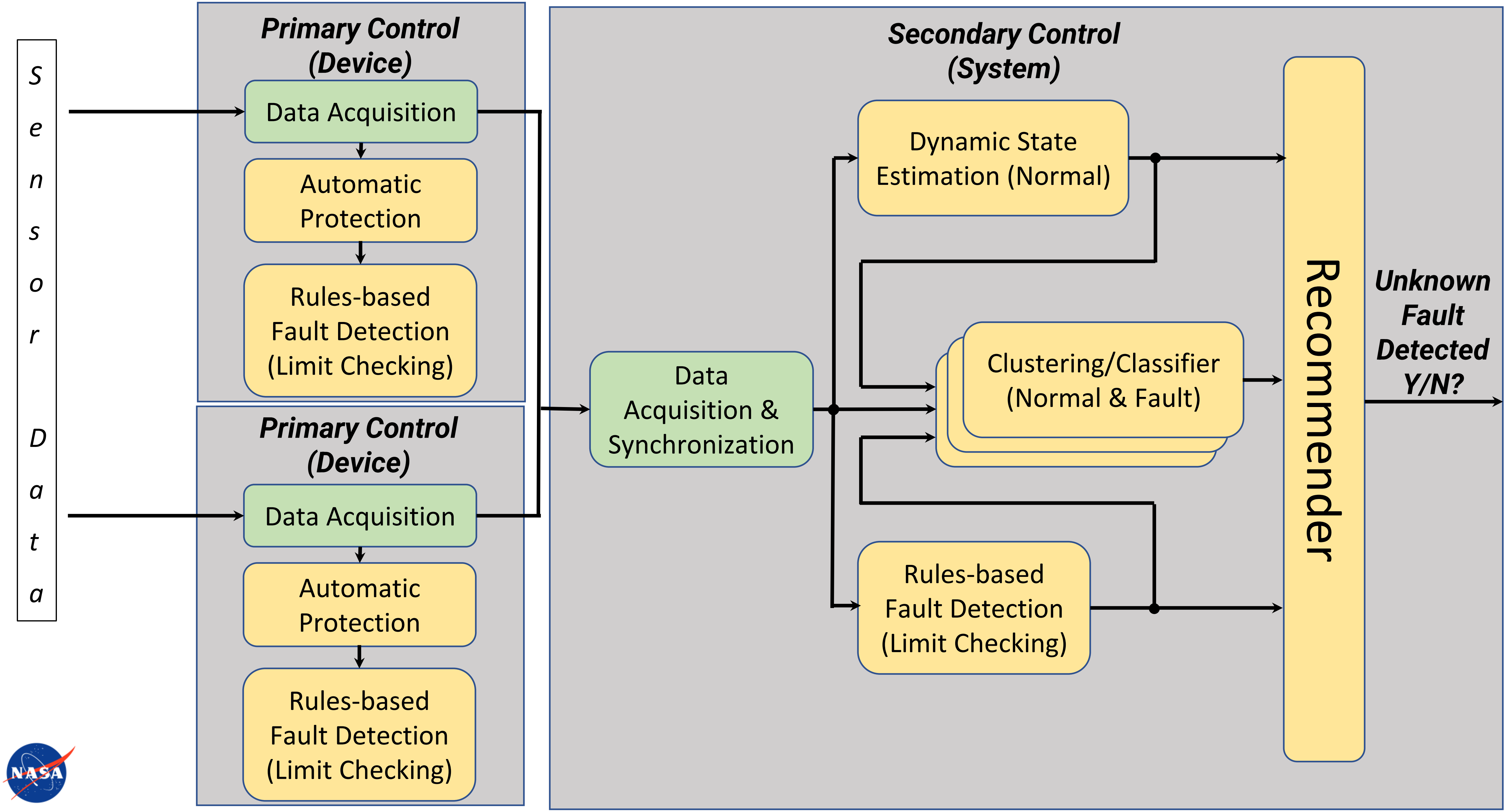
Energy Management:

- Manage distributed energy sources, loads and storage
- Maintain system stability
- Perform look-ahead analysis for power system planning

State Variance: Detecting Unknown Fault Types

- *Using machine learning models with limited pre-trained algorithms, demonstrate the ability to accurately identify a set of unknown or ambiguous faults.*

- *Show that the model is capable of accurately responding to faults it has not been trained on.*



Sensor Data
Primary Control (Device)
Data Acquisition
Automatic Protection
Rules-based Fault Detection (Limit Checking)
Primary Control (Device)
Data Acquisition
Automatic Protection
Rules-based Fault Detection (Limit Checking)
Secondary Control (System)
Data Acquisition & Synchronization
Dynamic State Estimation (Normal)
Clustering/Classifier (Normal & Fault)
Rules-based Fault Detection (Limit Checking)
Recommender
Unknown Fault Detected Y/N?

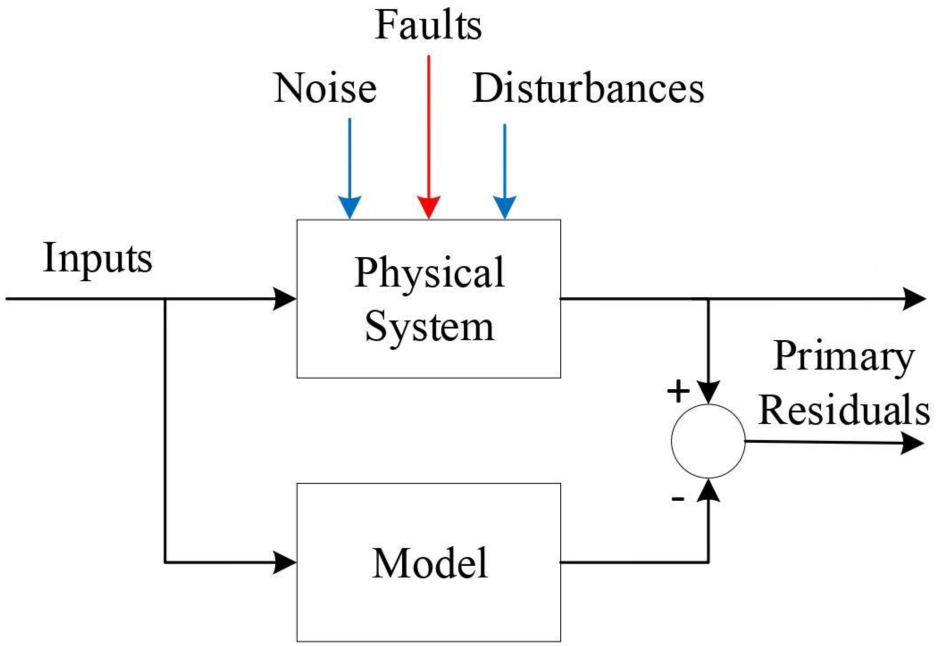
State Variance – State Estimation

<u>State Estimation provides ***analytical redundancy***</u>

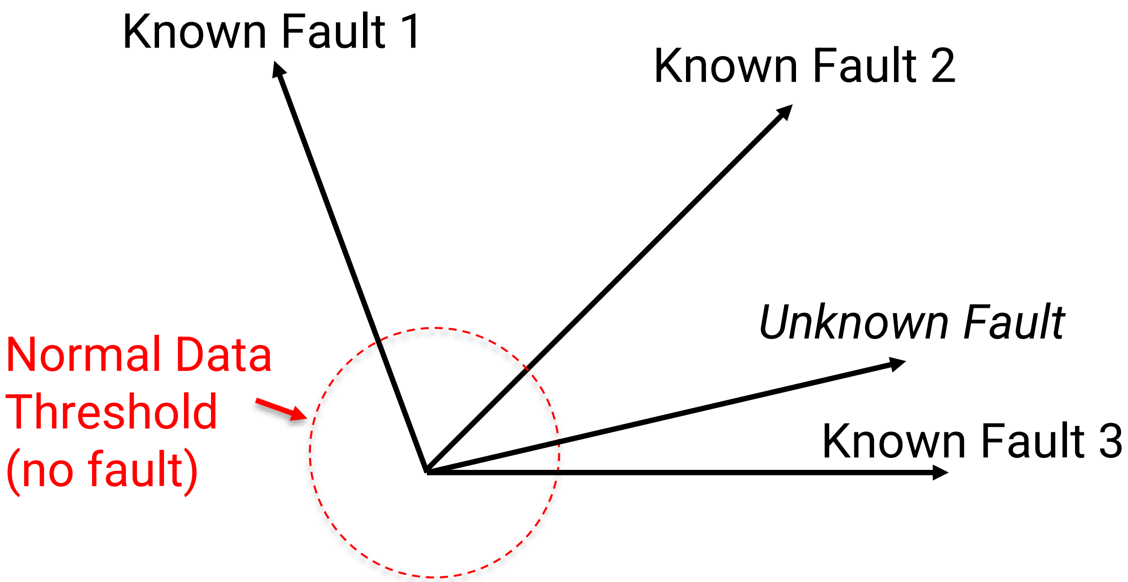
– A model based on physics first-principles of the system
– Estimates states that cannot be directly measured
– Provides reliable data in the face of noise, faults, and disturbances

<u>State Estimation application to State Variance</u>

– Can detect subtle differences between expected outputs and measured outputs called "residuals"
– The *magnitude* and *direction* of the **residual vector** may help determine the location and nature of the change, in particular, for unknown faults



Analytical Redundancy achieved through state estimation



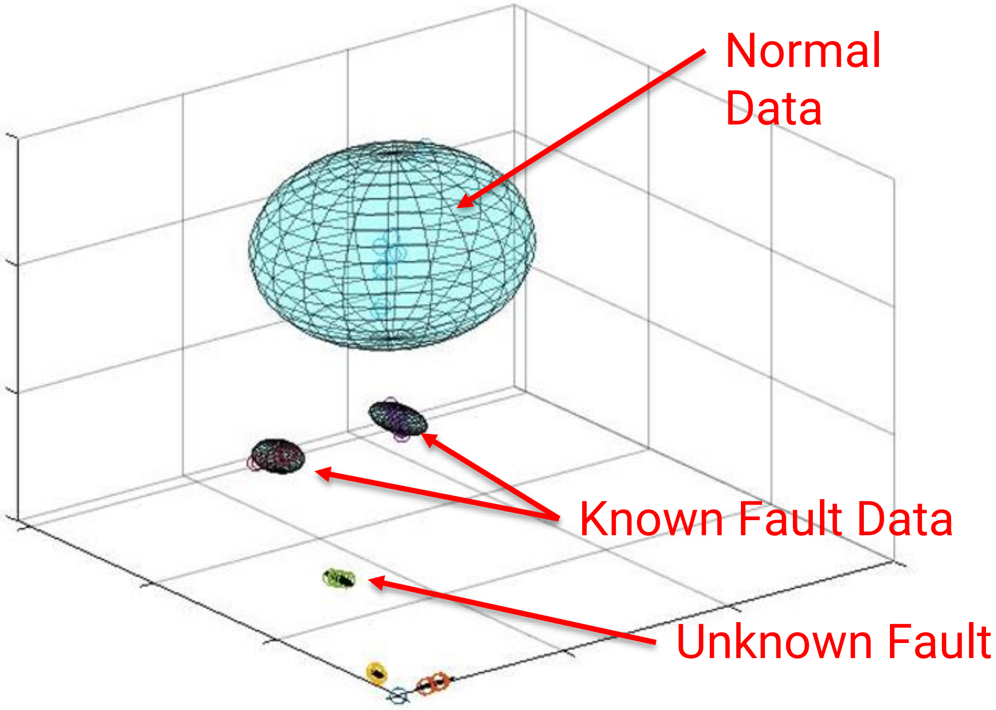
State Variance using Residuals

24

State Variance: Feature Analysis Using Clustering

Example Features

Single Stream	Statistics	$\text{mean}(w_t^i), \text{var}(w_t^i), \text{range}(w_t^i)$ $\text{median}(w_t^i), \text{entropy}(w_t^i), \text{hist}(w_t^i)$
	Difference	$u_t^i = \text{Diff}(x_t^i)$; Statistics
	Transformation	$\text{fft}(w_t^i), \text{wavelet}(w_t^i)$
Inter Stream	Deviation	$x^i - x^j \quad \forall i, \forall j \in \mathcal{N}(i)$
	Correlation	$corr(x^i, x^j) \quad \forall i, \forall j \in \mathcal{N}(i)$

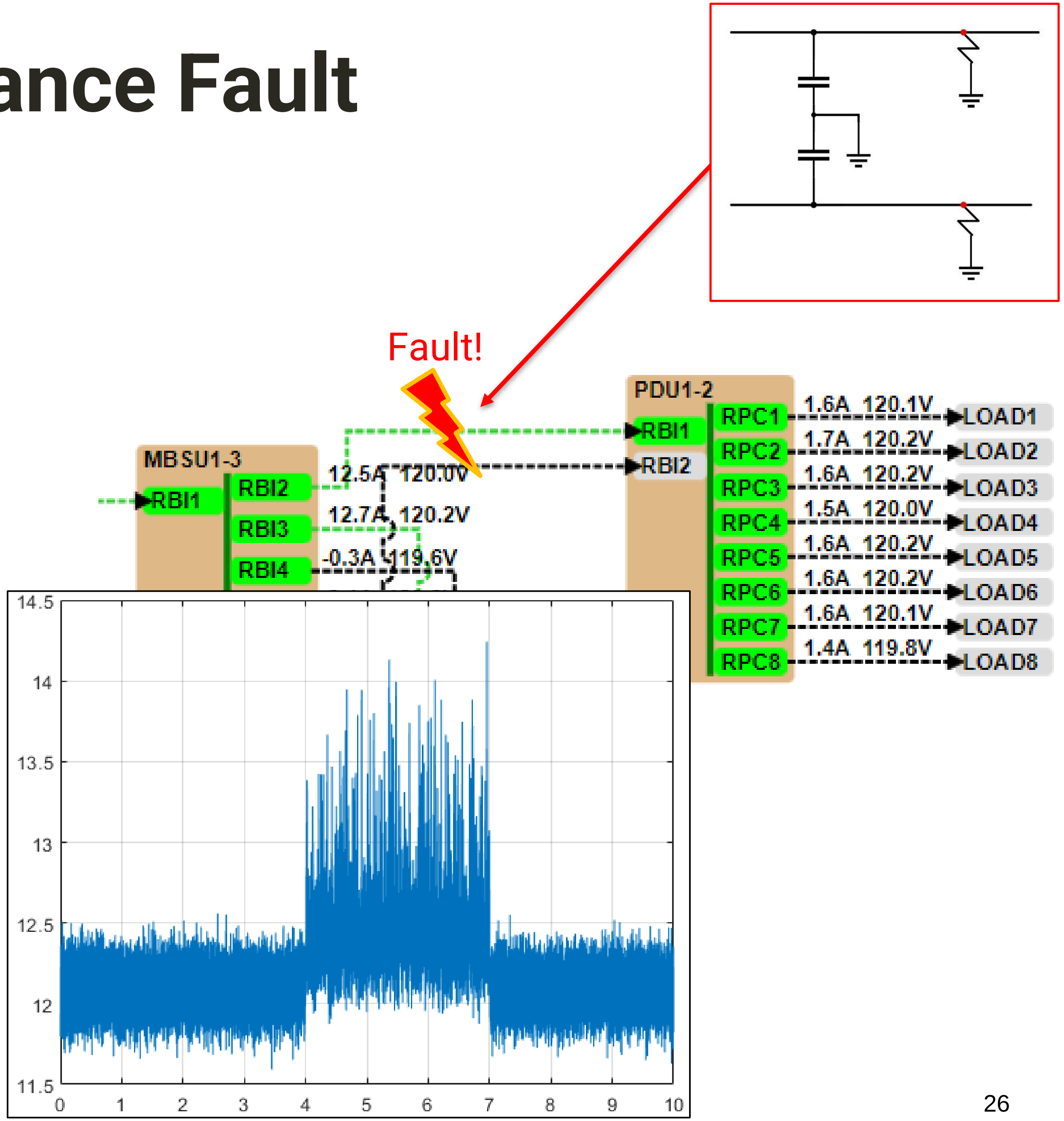


Clusters of faults based on feature set

- Extract large numbers of features from high-frequency data to characterize transient and steady-state data
- Use clustering techniques to find relationships between features, finding patterns in the data, producing a list of regular load performance characteristics
- Identify events outside of regular clusters, indicating abnormal/faulted device behavior
- Optionally include a system expert in the loop to validate machine learning results and strengthen accuracy

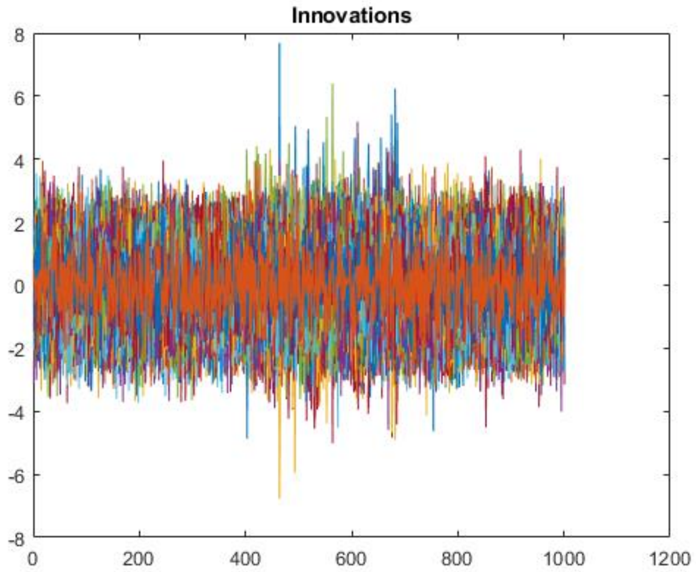
Use Case 1: High Impedance Fault

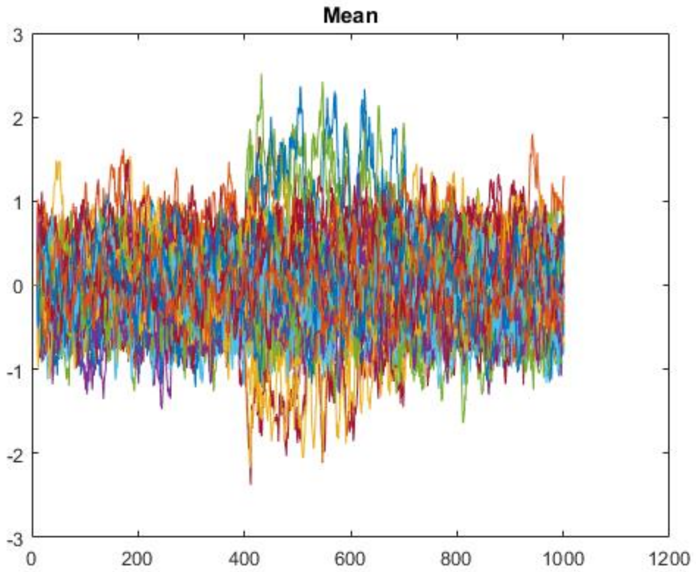
- Goal: use state estimation and AI techniques to detect an unknown fault (high impedance line-to-ground)
- Train model using normal and known fault data
- Goal: Detect change in state (nominal to off-nominal) for an unplanned high impedance fault

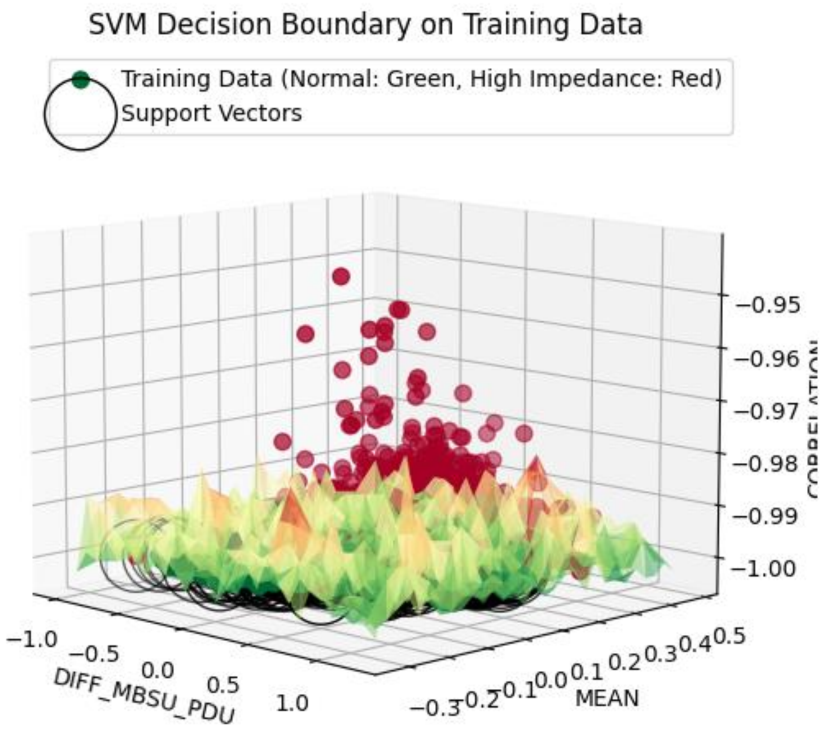


Use Case 1: Current Results

- Results of current model-based fault detection scheme (dynamic state estimation) and classification technique Support Vector Machine (SVM)







Conclusions

- NASA is exploring new tools to enable greater autonomy for their crewed missions to deep space
- There are many existing challenges preventing these technologies from being deployed in space flight software
- NASA needs to develop processes and standards to transition these technologies from research to application

Acknowledgements

[1] Porter, Daniel J., and John W. Dennis. "Test & evaluation of AI-enabled and autonomous systems: A literature review." (2020).

[2] Nesnas, Issa AD, Lorraine M. Fesq, and Richard A. Volpe. "Autonomy for space robots: Past, present, and future." *Current Robotics Reports* 2.3 (2021): 251-263.

[3] Carbone, M., and Loparo, K., “Fault Detection and Diagnosis in Spacecraft Electrical Power Systems,” AIAA Journal of Aerospace Information Systems, 2023

[4] Granger, Matthew G. *A Combined Framework for Control and Fault Monitoring of a DC Microgrid for Deep Space Applications*. Case Western Reserve University, 2021.