

A Appendix: Supplementary Information

A.1 Results: Supplementary Data and Analysis

This section provides deeper, disaggregated data to support the summary results and interpretations in the main text.

A.1.1 Performance Data Distributions

To provide a more nuanced view than the summary statistics in the main text, Figure A1 visualizes the full distributions of the execution time data for the APB and IPB.

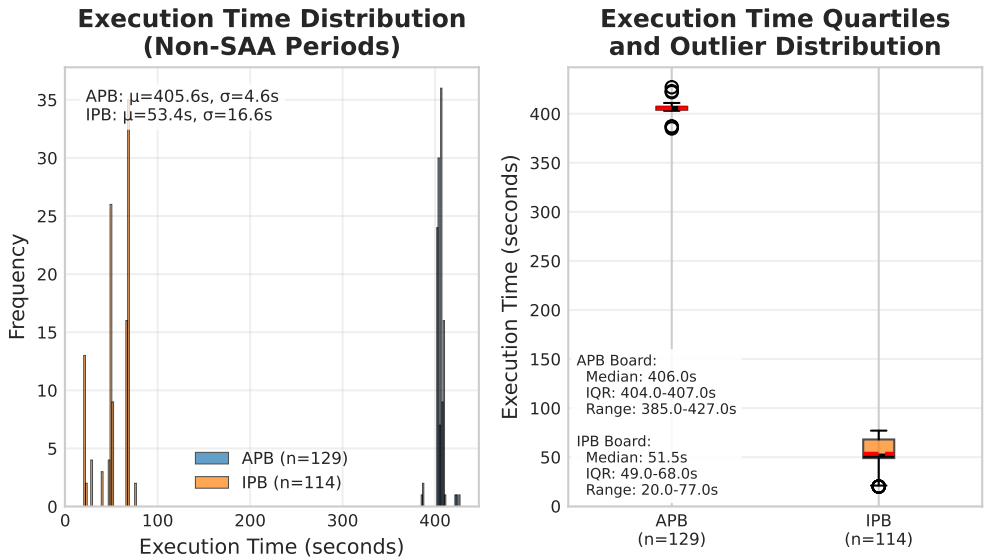


Fig. A1 Execution Time Distributions for APB and IPB (non-SAA). (Left) Histograms showing the frequency distribution of execution times, highlighting the narrow, consistent performance of the APB versus the wider variance of the IPB. (Right) Box-and-whisker plots comparing the quartiles and outlier ranges for both boards.

A.2 Environmental and Onboard Context

To provide context for the environmental analysis, Figure A2 shows the satellite's ground track relative to the South Atlantic Anomaly. To investigate potential confounding factors in the IPB's failure, Figure A3 correlates its reset count with onboard temperature data over the mission lifetime.

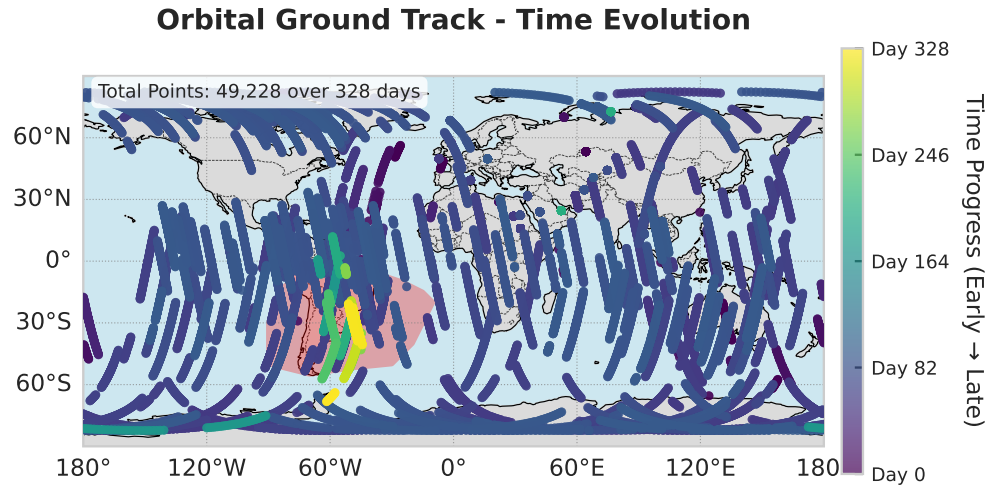


Fig. A2 Orbital Ground Track and South Atlantic Anomaly (SAA). This map shows representative orbital paths of the spacecraft with the SAA region overlaid, illustrating the frequency of passes through the high-radiation zone.

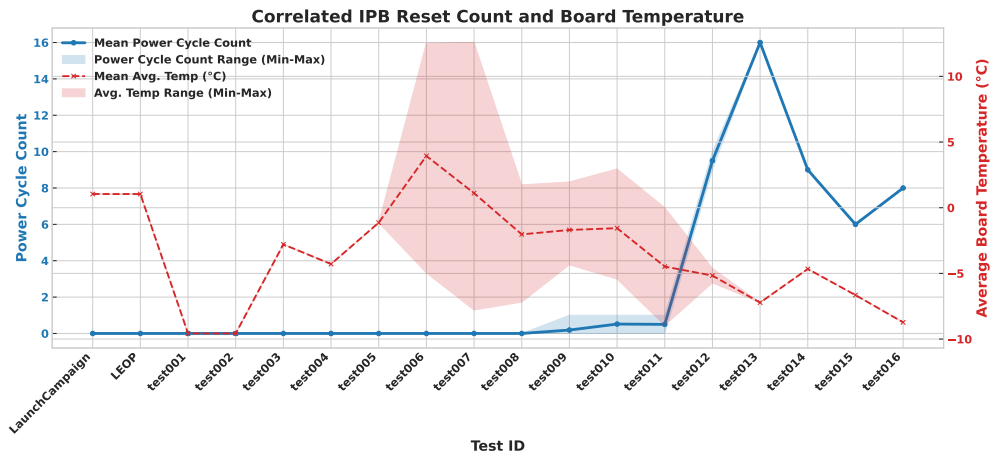


Fig. A3 Correlated IPB Reset Count and Board Temperature. This plot shows the IPB's power cycle count (left Y-axis, line) and its measured board temperature (right Y-axis, bar) for each test campaign. The lack of a strong correlation between temperature spikes and the escalating reset count from Test-012 onwards suggests that thermal stress was not the primary driver of the end-of-life failure cascade.

A.3 SAA and Anomaly Detection Methodology

To ensure the integrity of the in-orbit data and to correctly classify test runs based on their location relative to the South Atlantic Anomaly (SAA), a multi-stage data

processing pipeline was employed. This section details the algorithms used for SAA detection, position cross-validation, and anomaly detection.

A.3.1 SAA Detection and Position Cross-Validation

The primary algorithm for classifying each data point as inside or outside the SAA is detailed in Algorithm 4. This procedure iterates through the time-stamped ECEF (Earth-Centered, Earth-Fixed) position vectors received from the spacecraft’s GPS. For each point, it finds the closest Two-Line Element (TLE) set and uses the Skyfield propagation library to calculate the expected ECEF position. This serves as a cross-check, and the error between the onboard GPS position and the propagated TLE position is recorded. The onboard ECEF position is then converted to geodetic coordinates (latitude, longitude, altitude). A point is classified as within the SAA if its coordinates fall within a predefined SAA boundary polygon and its altitude is below 1000 km.

Algorithm 4 SAA Detection and Position Cross-Validation

```

1: procedure DETECTSAAANDVALIDATEPOSITION(onboard_positions,
   tle_data, saa_polygon)
2:   for all (timestamp, onboard_ecef) in onboard_positions do
3:     tle  $\leftarrow$  FINDCLOSESTTLE(tle_data, timestamp)
4:     propagated_ecef  $\leftarrow$  PROPAGATETLE(tle, timestamp)
5:     error_km  $\leftarrow$  NORM(onboard_ecef - propagated_ecef)
6:     (lat, lon, alt)  $\leftarrow$  ECEFTOGEODETTIC(onboard_ecef)
7:     if POINTINPOLYGON(lon, lat, saa_polygon) and alt < 1000 km then
8:       location_class  $\leftarrow$  “SAA”
9:     else
10:      location_class  $\leftarrow$  “non-SAA”
11:   STORERESULT(timestamp, lat, lon, alt, error_km, location_class)

```

A.3.2 Anomaly Detection

To identify potential issues with the onboard GPS data, two primary anomaly checks were performed, as shown in Algorithm 5. First, a Position Error Anomaly is flagged if the cross-check error between the onboard ECEF and the propagated TLE position exceeds a predefined threshold, indicating a significant deviation. Second, an ECEF Freeze Anomaly is flagged if the reported ECEF position remains static (changes by less than a small threshold) over a time gap that exceeds the expected update rate, which would suggest the GPS unit is frozen or telemetry is not being updated.

Algorithm 5 Anomaly Detection: ECEF Freeze and Large Position Error

```
1: procedure    DETECTANOMALIES(processed_positions,    error_thresh,
    freeze_thresh)
2:   last_ecef  $\leftarrow$  null
3:   last_timestamp  $\leftarrow$  null
4:   for all record in processed_positions do
5:     if record.error_km > error_thresh then
6:       FLAGANOMALY(record.timestamp, "Position Error")
7:     if last_ecef is not null then
8:       position_change  $\leftarrow$  NORM(record.onboard_ecef - last_ecef)
9:       if position_change < freeze_thresh then
10:        time_gap  $\leftarrow$  record.timestamp - last_timestamp
11:        if time_gap > max_allowed_gap then
12:          FLAGANOMALY(record.timestamp, "ECEF Freeze")
13:   last_ecef  $\leftarrow$  record.onboard_ecef
14:   last_timestamp  $\leftarrow$  record.timestamp
```
