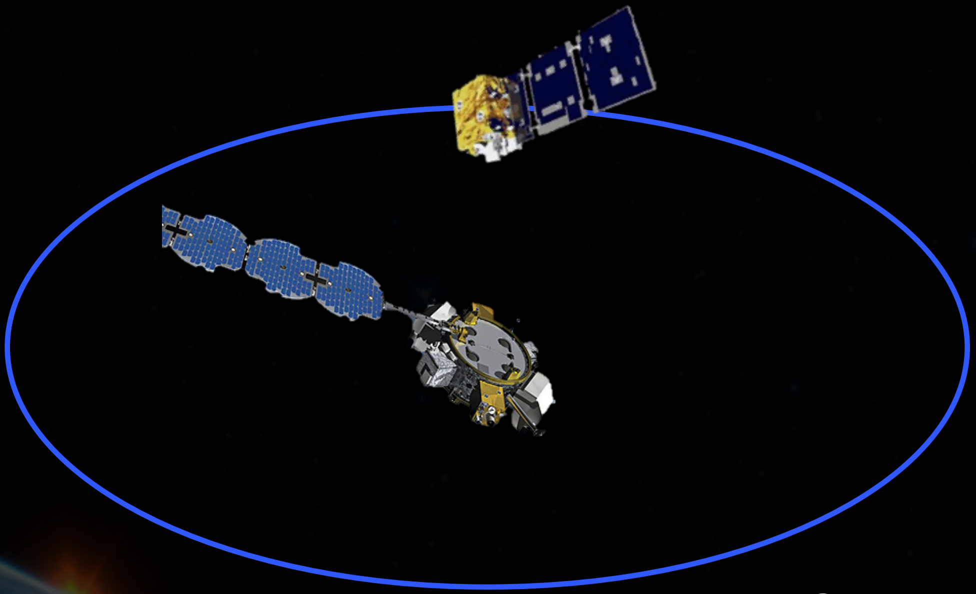
Resilient Autonomous Satellite Mode Switching for Space Weather



Dr. Christopher "Chrispy" Petersen

Assistant Professor

Mr. Faraz Abed Azad

PhD Student

In Collaboration with:

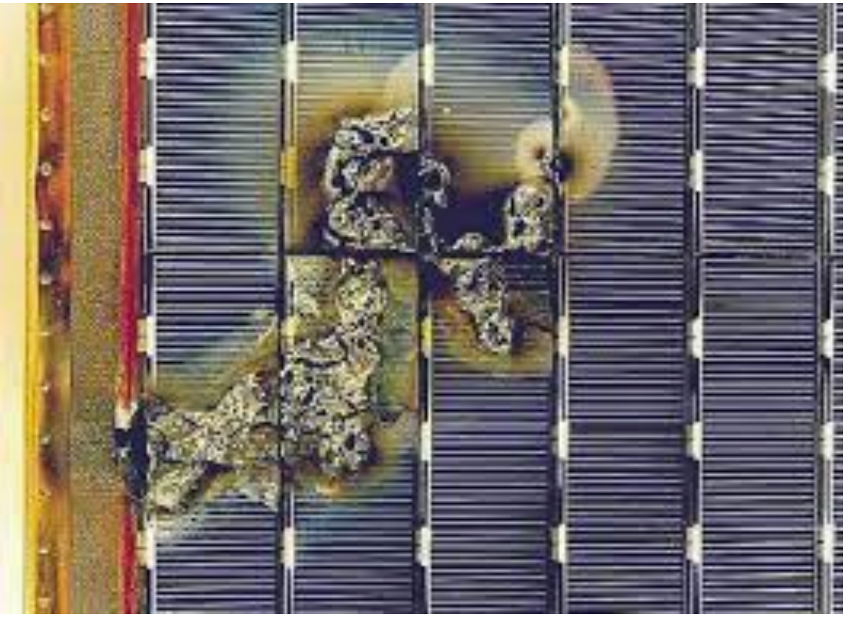
Dr. Alicia K. Petersen

Assistant Professor

UF Herbert Wertheim College of Engineering
UNIVERSITY *of* FLORIDA

Outline

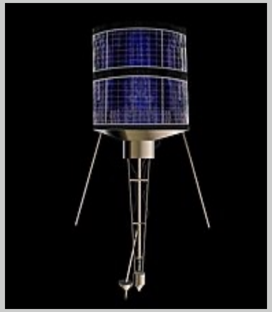
- Causes of Failure
- Types of Failure
- Space Environment Failures
- Resiliency via Prevention, Degradation, and Recovery
- Satellite Modes
- Connections to Assured Autonomy
- Real-time Considerations



Solar Panel Damage Due to Arching

Causes of Failure

Iridium 33 and Kosmos 2251 Collision



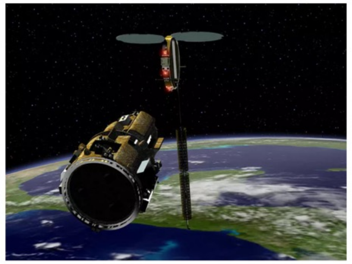
Effects of Worst Satellite Breakups in History Still Felt Today

By Leonard David published January 28, 2013

SPACE.com

Human Decision Error

Demonstration of Autonomous Rendezvous Technology (DART) Failure



An artist's concept of the DART spacecraft as it bears down on its target satellite in orbit. (Image credit: Orbital Sciences.)

Multiple Errors Caused DART Rendezvous Mission Mishap

SPACENEWS

by SpaceNews Editor — June 29, 2004

Navigation/Guidance SW Error

SpaceX Satellite Loss

Aurora viewed from North Dakota on Feb. 2. (Alex Resel)

How a rather mundane space storm knocked out 40 SpaceX satellites

The Washington Post

Democracy Dies in Darkness

Environment Unknowns

Things will go wrong in space, the questions become (a) how to best prepare and (b) how to act in-situ **calmly & responsibly, &** (c) how **autonomy can assist in these situations**

Types of Failure

Some people classify failures differently, be aware of lexicons

Ultimately spacecraft failures manifest in different ways

- Electrical – Solar panel damage, electric circuits, battery issues
- Mechanical – Imagers damaged, leaky thrusters
- Software – Memory leaks, bit flips, bad coding
- Unknown – (?)

Most People Blame Space Weather for Something They Do Not Understand!

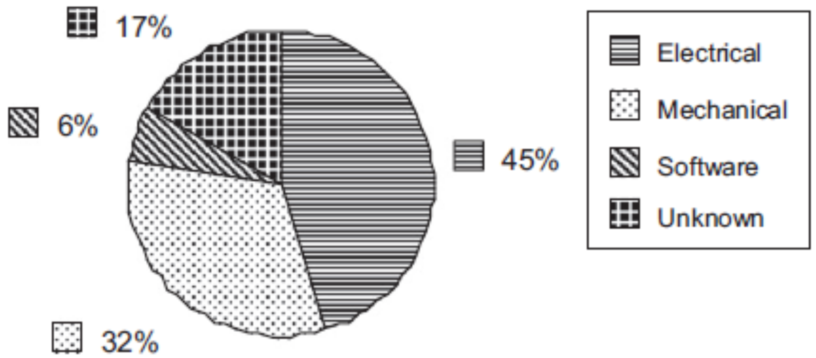


Fig. 2. Spacecraft failure type.

M. Tafazoli/ActaAstronautica64(2009)195–205

Space weather can causes all of these types of failures

Impacts of Space Environment Failure

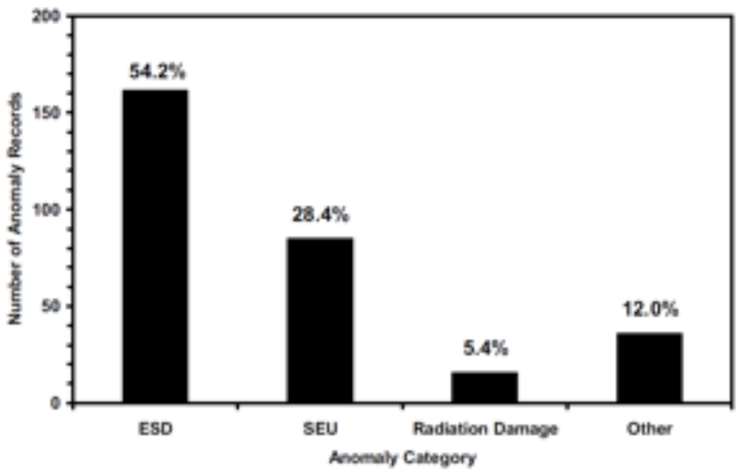
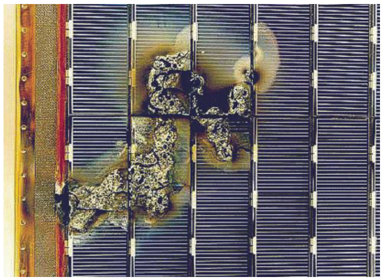


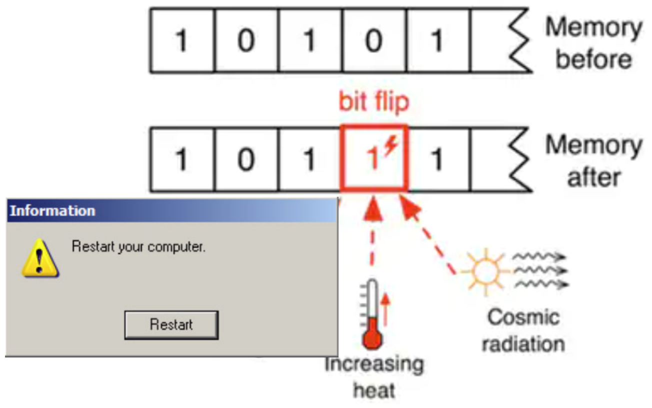
Figure 1. Spacecraft anomalies as a function of the space environment effect, where ESD is electrostatic discharge, SEU is single event upsets, Radiation Damage is total ionizing or non-ionizing dose, and Other represents other causes or unknown sources, from Koons et al., Aerospace Technical Report, 1999.

Effects of Space Weather on Technology Infrastructure

- (ESD) Electrostatic Discharge – causes "physical damage"
- (SEU) Single Event Upset – causes "soft damage"
- Radiation – causes "physical damage"
- Other – Everything else



ESD Arching Event



SEU Software Event

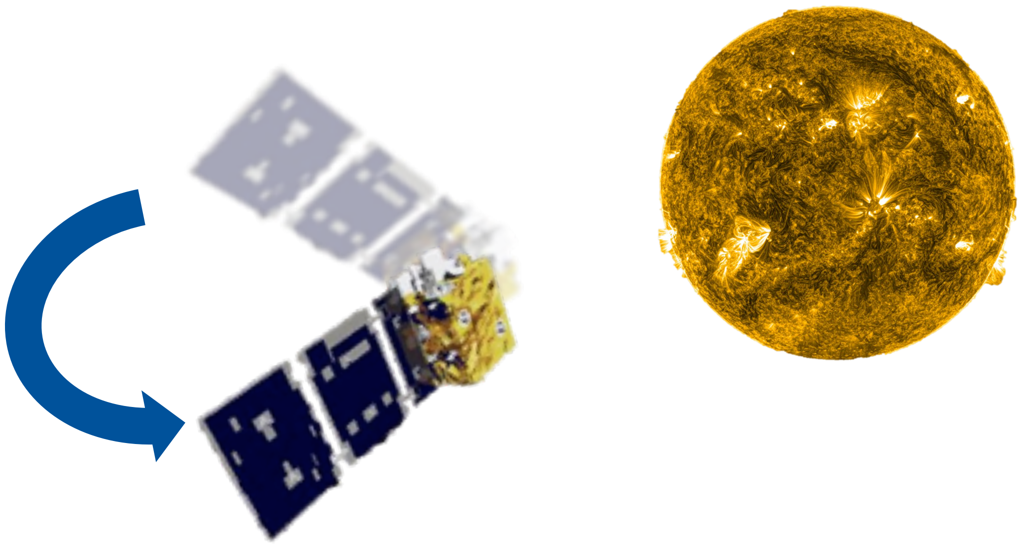
Impacts vary from "hard failures" to "soft recoverable failure"

UF UNIVERSITY of FLORIDA
Duke UNIVERSITY
TEXAS The University of Texas at Austin
UC SANTA CRUZ

Resiliency via Prevention, Degradation & Recovery Set-up (PDR)

- Safe mode will take you off mission
  - Safe but not resilient
  - Drifting can take satellites tremendously off mission

- When preparing or experiencing a fault, what is the best decision to make for
  - Preventing – Ensuring an impending impact does not hurt the satellite in the future
  - Safe degradation – Fail so that the mission can be achieved back to X% > 0%
  - Recovery Set-up – Fail so that you can get back on mission easier



Example – Changing satellite rotation and reducing on-board computations to reduce spacecraft potential.

Can save majority of the mission, even if unknowns are known or otherwise

UF UNIVERSITY of FLORIDA · Duke UNIVERSITY · TEXAS The University of Texas at Austin · UC SANTA CRUZ

Categorize Satellite Modes for Possible Actions

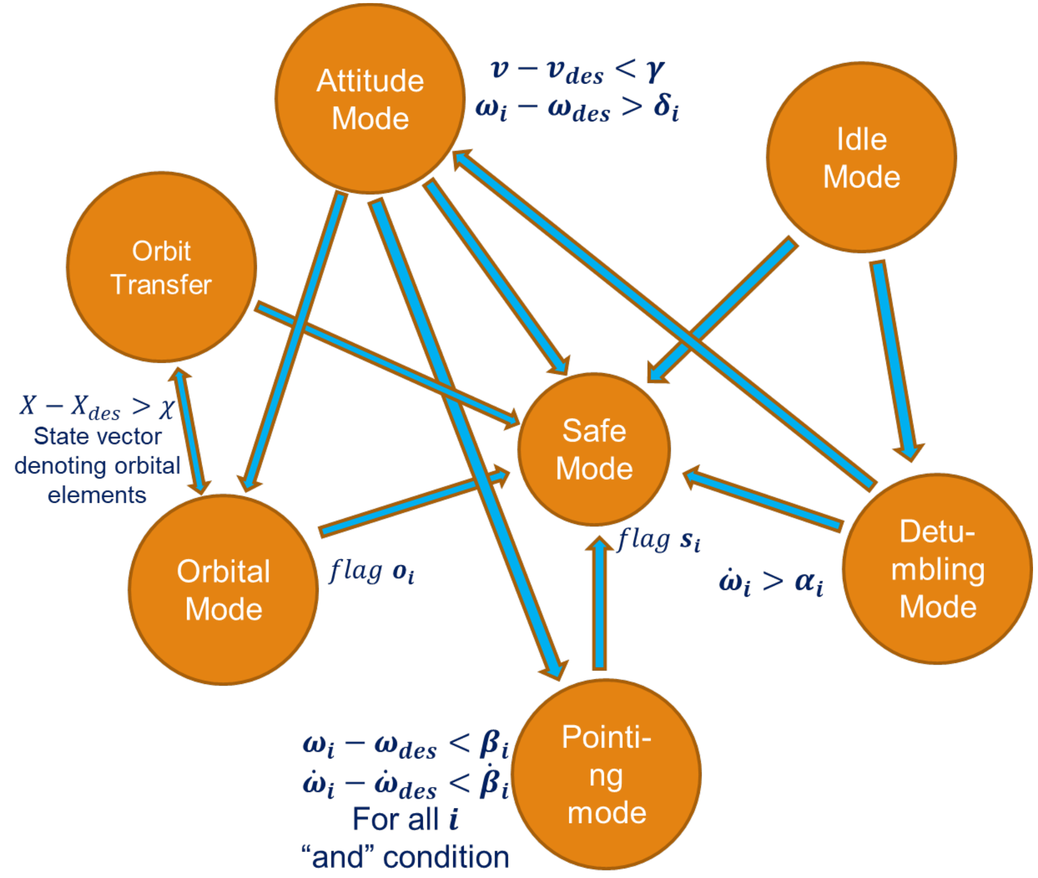
- Satellite control/nav algorithms can have an infinite action space
  - However these infinite possibilities in a computer are "decision making" nodes with a finite number of tasks

- Some nodes cannot be reached from other nodes
  - Safe mode is always "reachable"

Its also the "easy" solution

Understanding how to traverse this set as well as the node subtasks will enable on-mission resilience

Choosing an action is not limited to after event, but rather before, during, and after

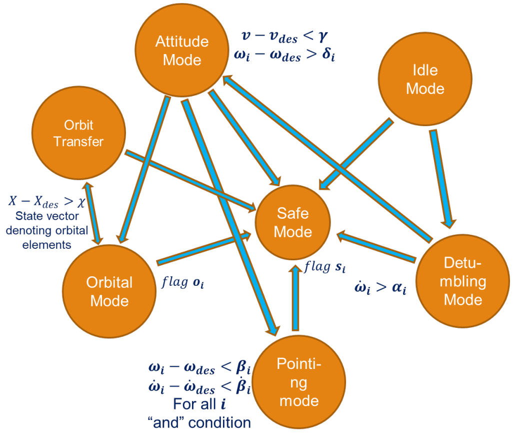


Node-Transition Graph for Satellites

Satellite Mode Details

Performed Work

- Initial set of nodes with transitions established
- Sub tasks (e.g. control-algorithms) identified



- **Now to mathematically define:**
  - Finite state machine
  - Hybrid system
  - Markov decision making process

Table of Spacecraft Modes

Modes (with #)	Sensors & Actuators (should extend to different columns)	Details	Objectives (extend?)	Admissible modes to switch to (Create a small chain diagram for each?)	The mode's objectives (across all of its sub-modes)	The mode's conditions (across all of its sub-modes)	Control Algorithm
1. Idle	Sensors: On (Attitude, velocities, orbit, etc.) Actuators: Off	The sensors monitor the states.	Standby for the launch (and shortly for mode transition?)	3, 2	Standby		Just monitor, no controls
2. Detumbling (damping)	Sensors: On Actuators: Reaction wheels and magnetic torquers (for damping)	Ensure redundancy (for safety) for DOF	Stabilizes the spacecraft	3,5,6	$\dot{\theta} < \dot{\alpha}_{\theta}$ $\dot{\varphi} < \dot{\alpha}_{\varphi}$ $\dot{\psi} < \dot{\alpha}_{\psi}$	$\dot{\theta} > \dot{\beta}_{\theta}$ $\dot{\varphi} > \dot{\beta}_{\varphi}$ $\dot{\psi} > \dot{\beta}_{\psi}$	Lyapunov control (Need to prevent actuators saturation as well, by solving an NLP (MPCs) or Barrier Lyapunov control methods)
3. Safe hold	Sensors: Monitor the systems' health and states. Actuators: Using the actuators, ensure robustness to disturbances (one approach can be generating small spins)	Initiated by the fault or anomaly detection system. Probably should have different levels (sub-modes)!?	Face the solar panels towards the sun for power generation, also, minimize power consumption. Ensure health of the satellite (like over-heating)	1,2	$\alpha < \frac{\hat{n}\,\vec{d}^T}{\lVert\vec{d}\rVert_2}$ • For $0 < \alpha < 1$ close to 1 • $\hat{n}$ denotes the unit vector normal to the solar panel. • $\vec{d}$ denotes the vector from center of the satellite's solar panel to the sun.	Avoid translation? (depends on the situation that we are in?!)	Fault Tolerant Control algorithms

With nodes and transitions mathematically formalized, a optimization for resilience can be cast

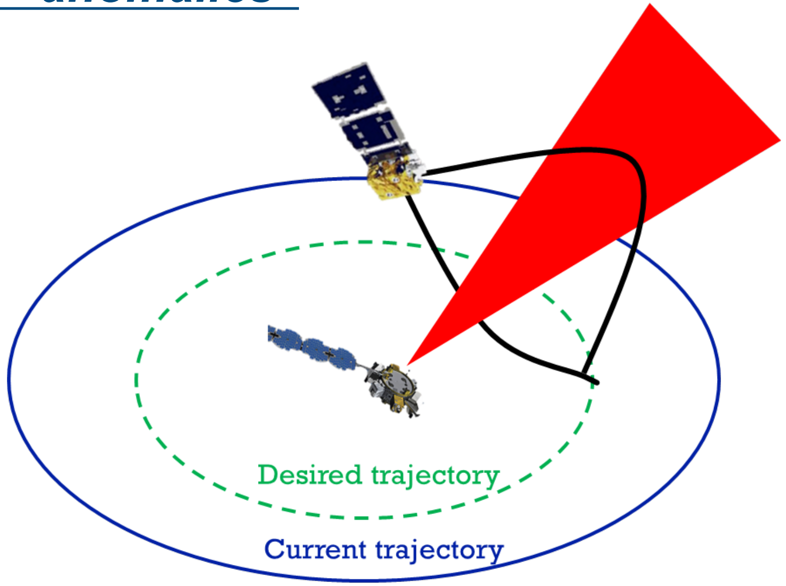
UF UNIVERSITY of FLORIDA · Duke UNIVERSITY · TEXAS The University of Texas at Austin · UC SANTA CRUZ

Connections with Safety and Assured Autonomy

Autonomy can be thought of as a known unknown

During testing it can lead to off-mission "anomalies"

- Space weather are unknown unknowns
- Autonomy is a known unknown
- Preparing the satellite to still stay on mission as much as possible to these external stimuli
  - Ensures satellites do a safe mission
  - Enables human "feelings" to accept risker decisions



Satellite can choose two ways around an exclusion cone, which way would you "feel" to take

Ensure that even under unknown actions, missions can be safe and accomplished partially

UF UNIVERSITY of FLORIDA | Duke UNIVERSITY | TEXAS The University of Texas at Austin | UC SANTA CRUZ

Aspects of Time for Assured Autonomy

- Real-time – The ability for a vehicle to make decisions with the allocated computational resources on time fames necessary to complete the mission
  - This Is mission and vehicle dependent
  - Does not imply sufficiently fast decisions at constant rate, real-time can imply decisions made asynchronously

Scale of Real-Time

1 2 3 4 5 6 7

Current State of Thinking → Trending Towards Development

Current State of Thinking	Trending Towards Development
Sufficient Fast	**Memory Efficient**
Exploit mathematical theory for rapid computation	Exploit numerical tricks for memory usage
Optimality	**Feasibility**
Take necessary time to get best solution	Take necessary time to ensure safe solution
Diverse	**Consistent**
Explore search space extensively (often randomly) to find non-trivial answers	Ensure that roughly same inputs provide roughly same outputs to ensure trust
Monolithic	**Reconfigurable**
Methods are self contained for easy deployment and execution	Methods have non-traditional parameters accessible for decision making

Ensure that this method can be used in real-time before, during, and after event

Questions