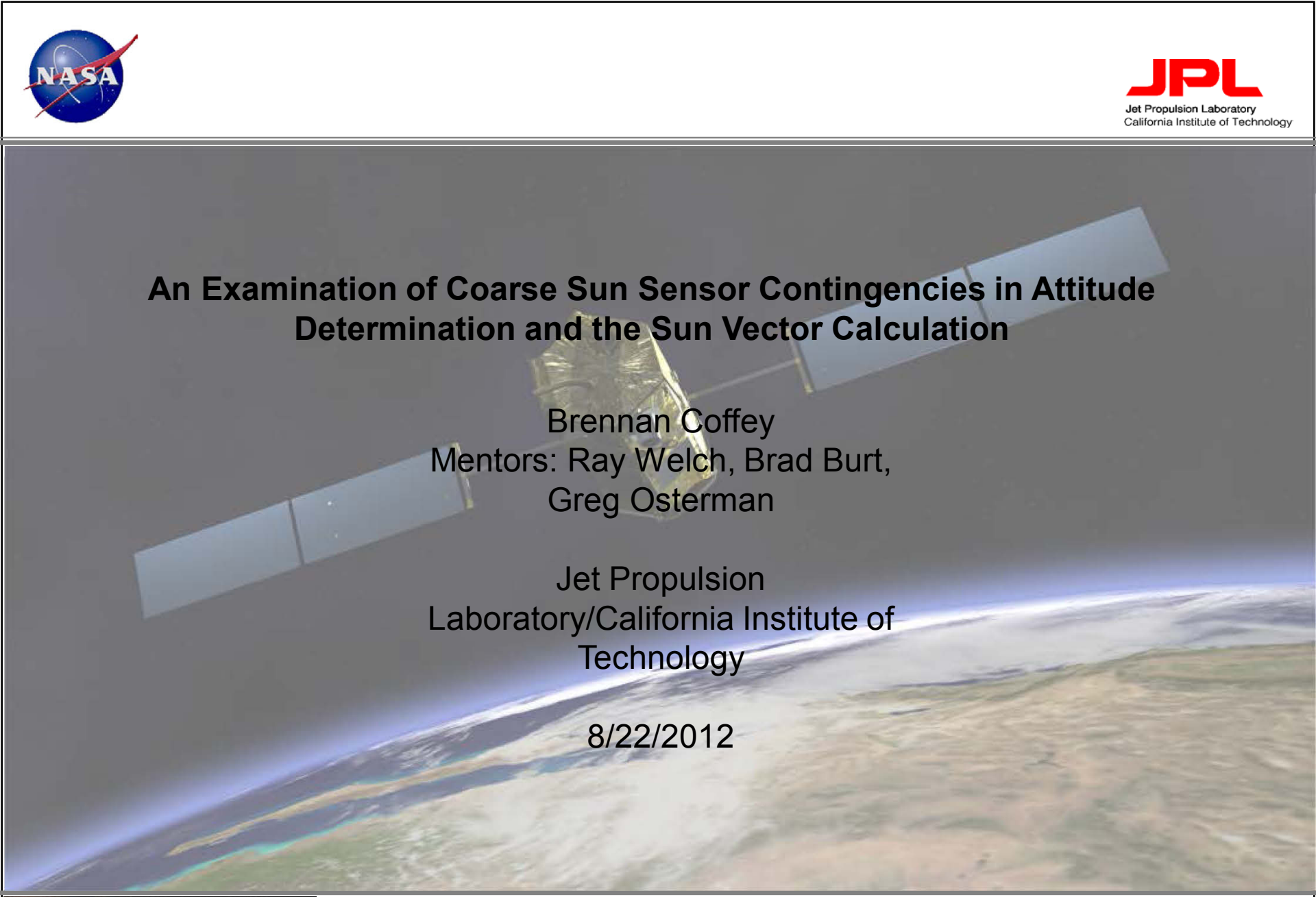


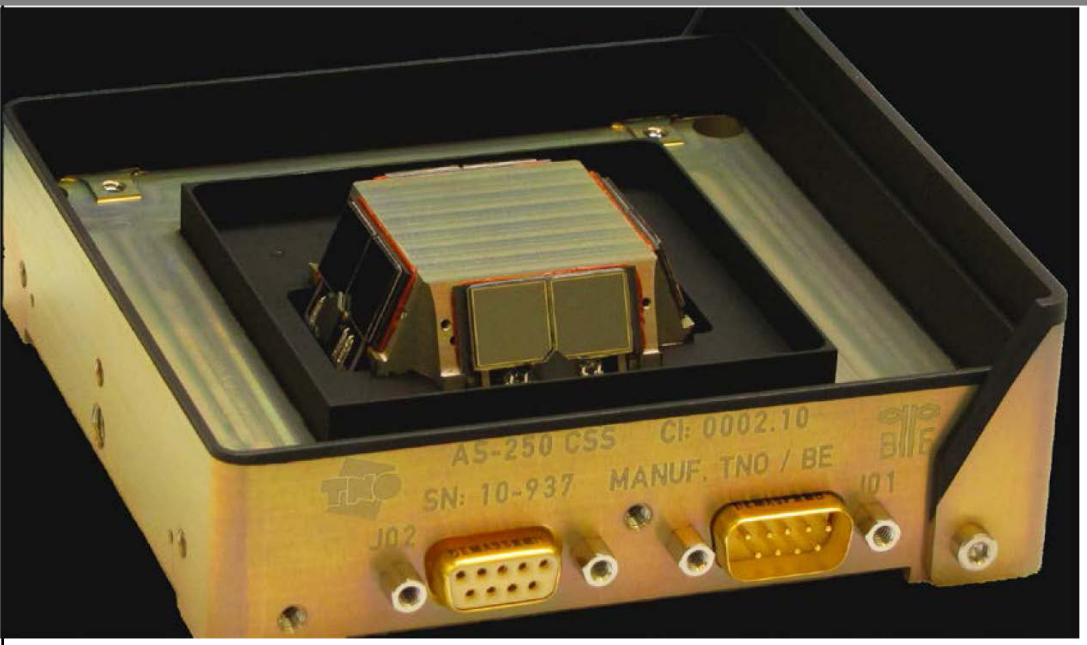
An Examination of Coarse Sun Sensor Contingencies in Attitude Determination and the Sun Vector Calculation

Brennan Coffey
Mentors: Ray Welch, Brad Burt, Greg Osterman

Jet Propulsion Laboratory/California Institute of Technology

8/22/2012

Background

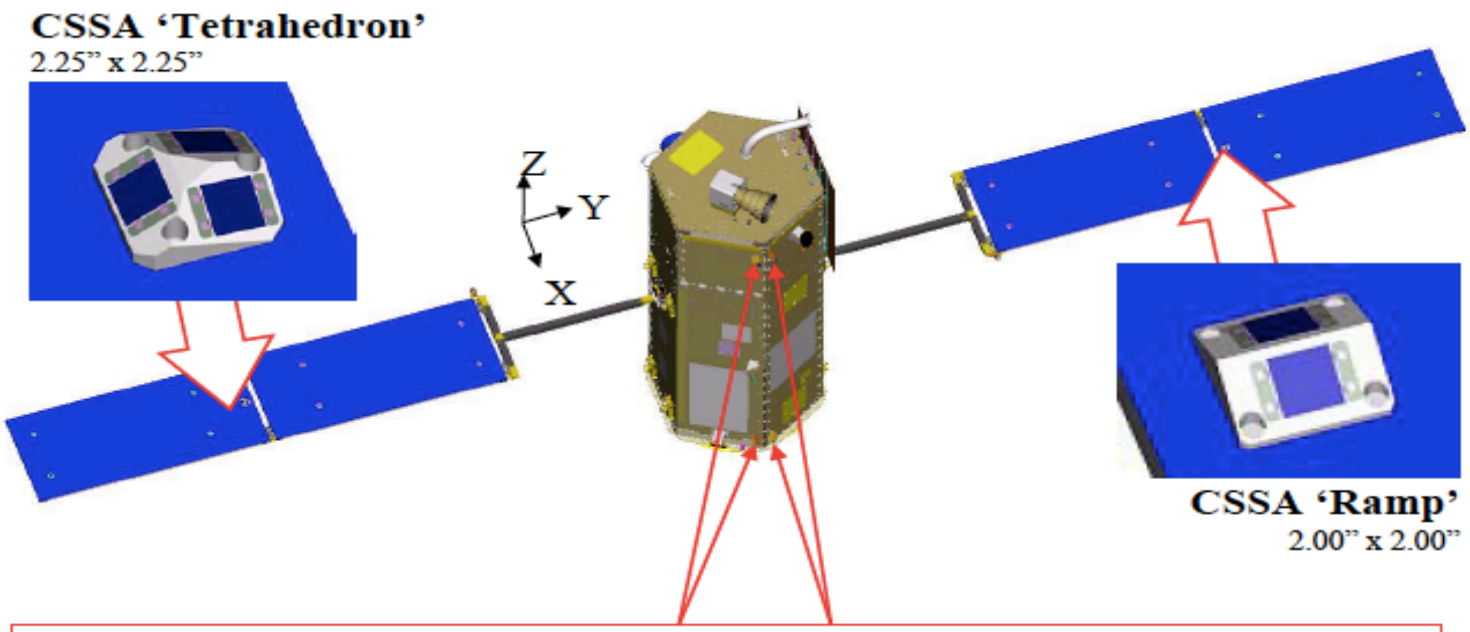


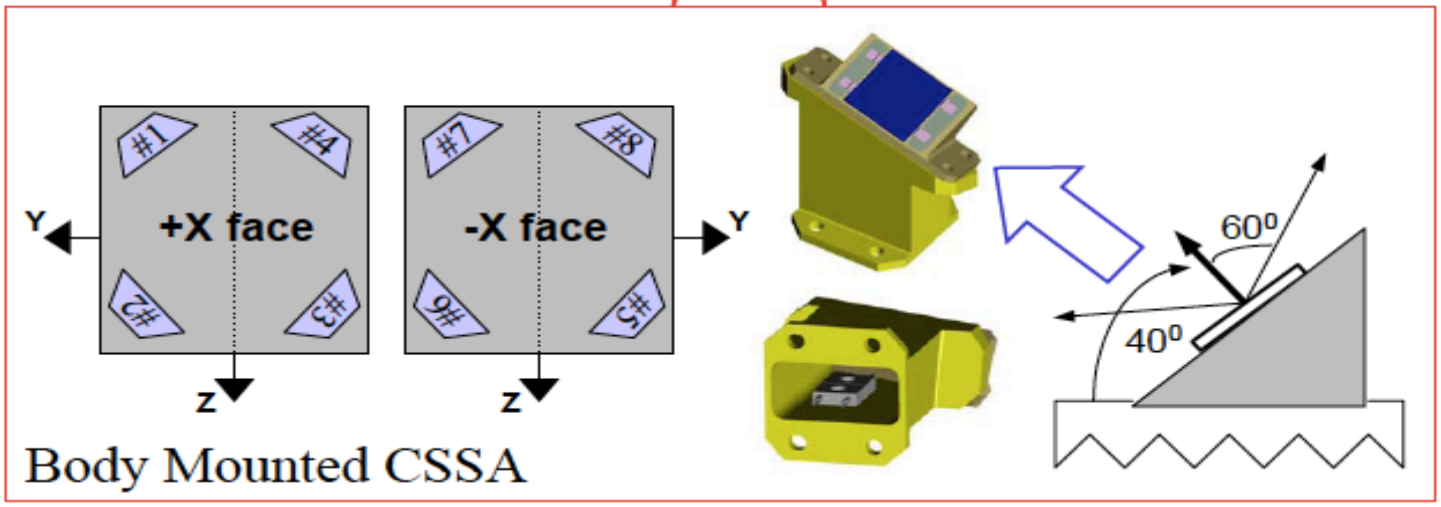
- In attitude determination, the goal is to determine the satellite's orientation for pointing in space
  - Inertial reference frame
  - Body reference frame
  - (for this problem) Array reference frame
- Coarse Sun Sensors (CSS) measure the current produced when the sun is partially normal to their surface

The Main Goal:

- Through measurements from the CSS we can determine the sun's position
- With the sun's position known, the arrays can track the sun and ensure that max power is received at all times in an orbit

Background





- The Coarse Sun Sensor Assembly (CSSA) is one of the most robust of the hardware systems on a satellite
- Recent manufacturing issues have been called into question the stability of the CSSA's

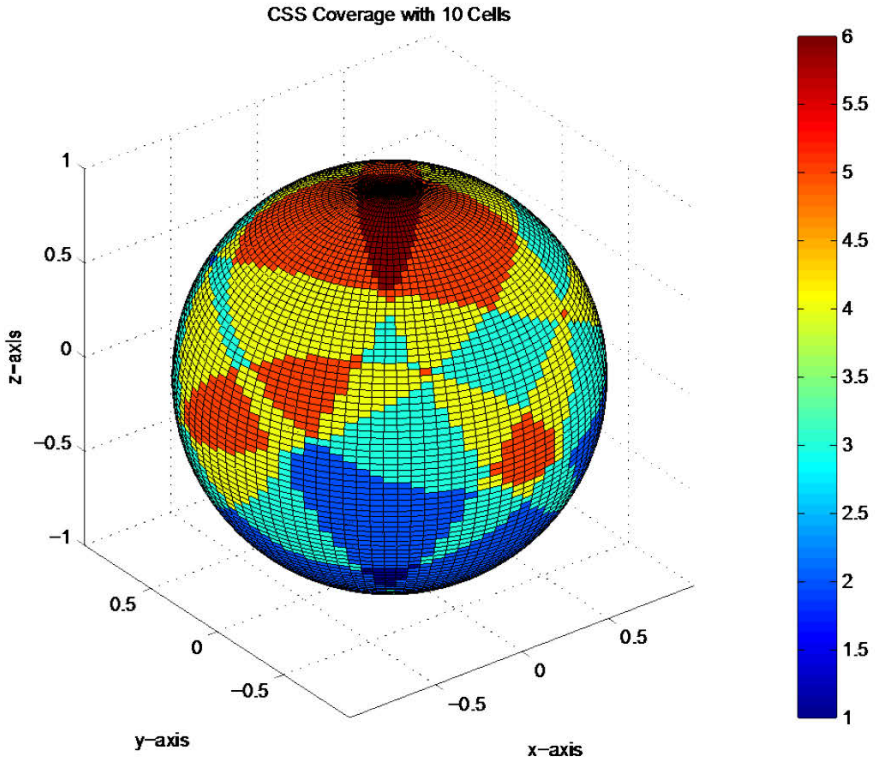
Questions to Answer

- With failed CSS, what type of holes do we see in the celestial sphere?
- What is the max sustainable loss for no holes in the celestial sphere?
- With failed CSS, what is the best way to reacquire the sun's position?
- How does the algorithm handle these failure scenarios and how robust is it?
  - Could mitigation include using extra cells for input?
- With the manufacturing issue, what type of failure scenarios can be seen?
  - How many failures are sustainable?

Key

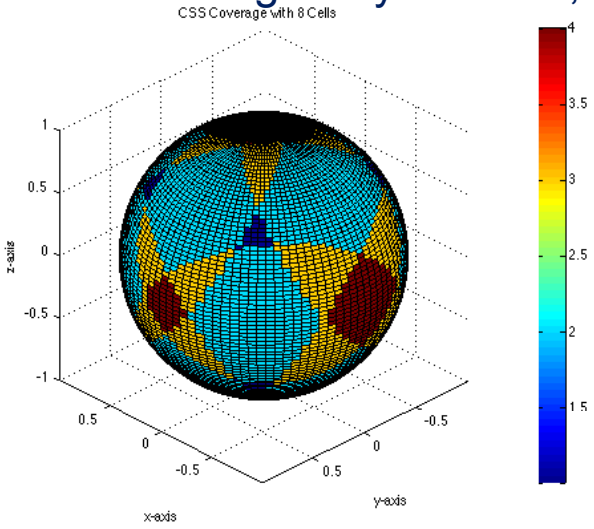
- Graphs are of the Celestial sphere with a minimum of 0 CSS viewing, and a max of 9
- All output is generated by OCO_FOV_v1_3.m
- All cells are considered except for the dual array sensors
- The array h = [0 1 1 0 1 1 0 1 0 1 0 1 0] shows the cells
  - 1 denotes a working array
  - 0 denotes a failed array

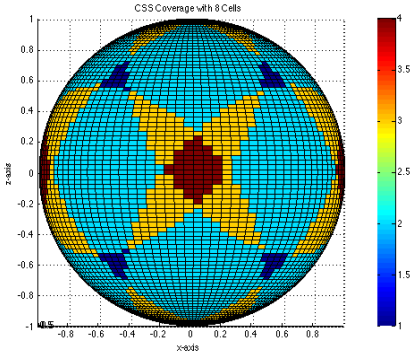
- Sensors 1,4,7,8 create +Z
- Sensors 2,3,5,6 create –Z
- Sensors 9,10,11 are the three array mounted CSS

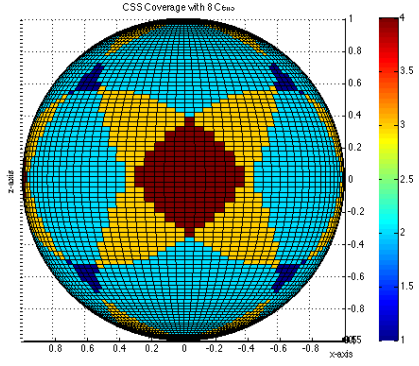


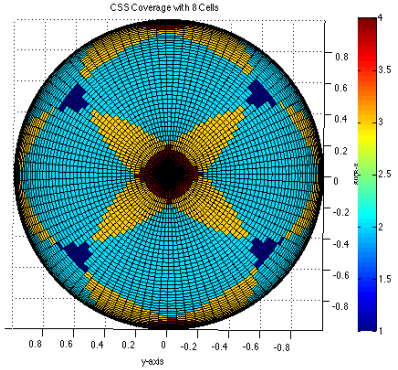
CSS Coverage Holes

- First adapted a script from SMAP
- Added functionality to view the OCO-2 configuration
- Examined static case:
  - Body sensors alone, the mission can sustain 2 failed sensors (with a rotation)
  - Including array sensors, maximum sustainable failed sensors is 3



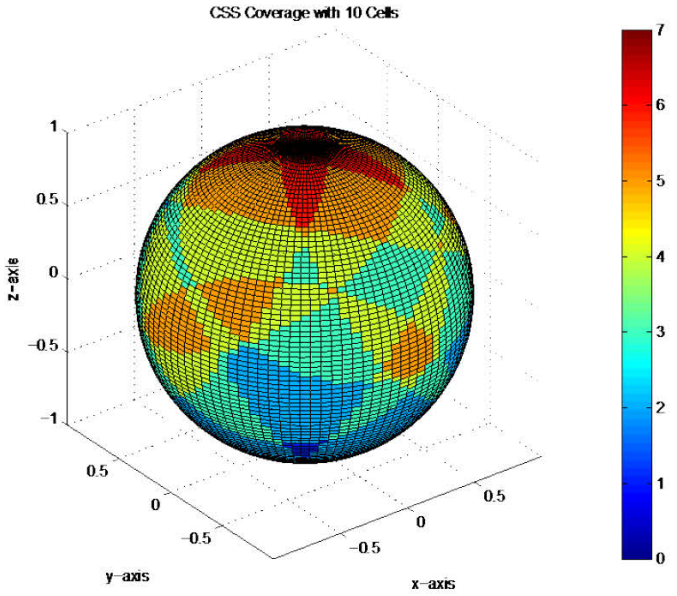






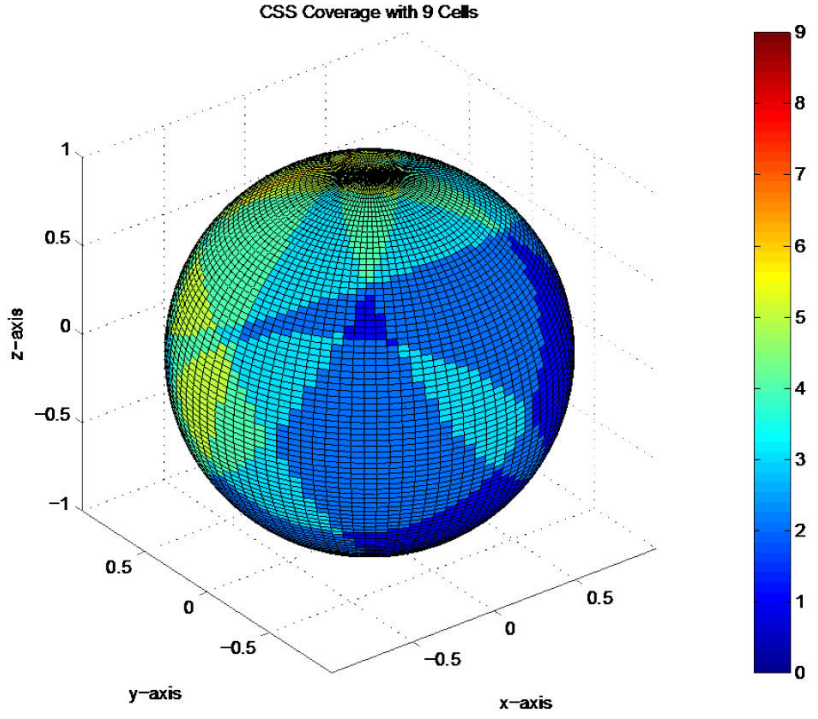
1 CSS Lost

- With arrays pointed just deployed (Sun vector at location [0 0 1]) the largest percentage of sky seen by zero sensors is 0.2%
  - Percentage seen by zero: 0.0% to 0.2%
  - Percentage seen by one : 0.87% to 14.15%
  - Percentage seen by two : 23.15% to 32.40%
  - Percentage seen by three: 21.36% to 26.02%



2 CSS Lost

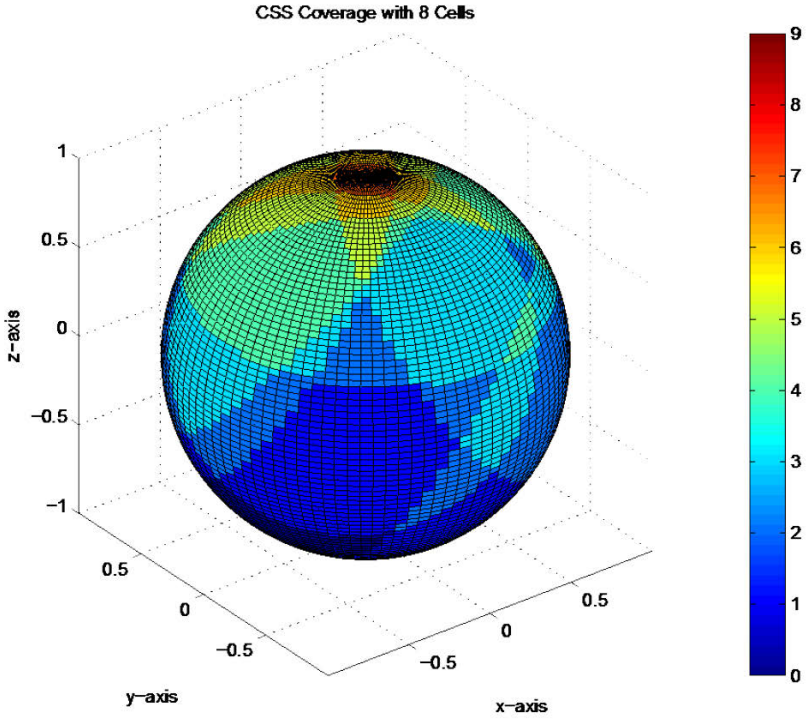
- Percentage seen by zero: 0.0% to 5.09%
- Percentage seen by one : 1.92% to 27.81%
- Percentage seen by two : 20.40% to 40.19%
- Percentage seen by three: 10.94% to 32.96%



The worst case, with 5.09% seen by 0 and 15.7% seen by 1

3 CSS Lost

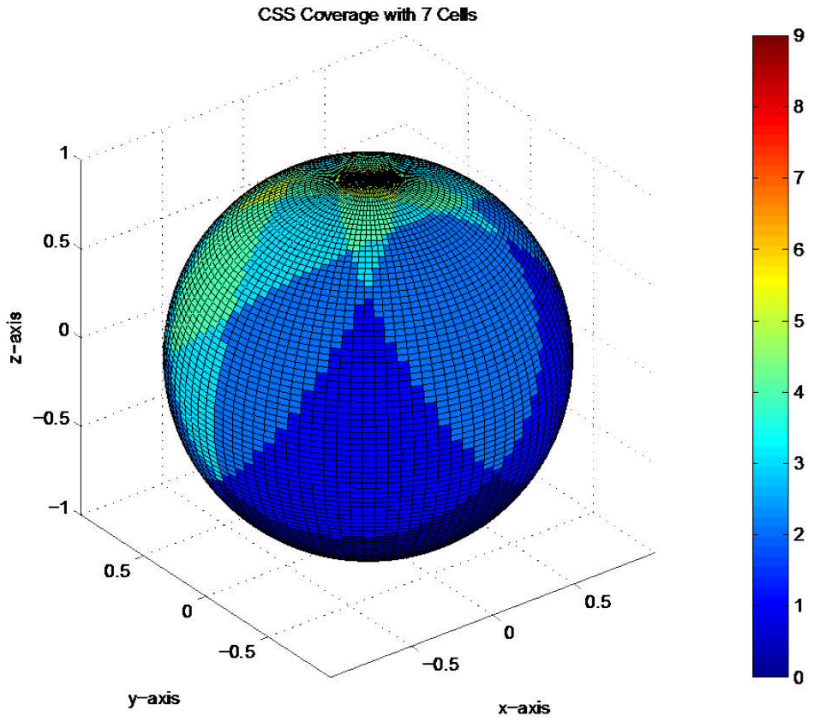
- Percentage seen by zero: 0.0% to 12.76%
- Percentage seen by one : 7.48% to 30.79%
- Percentage seen by two : 9.66% to 47.89%
- Percentage seen by three: 10.90% to 44.78%



The worst case with 12.75% seen by 0 and 27.29% seen by 1

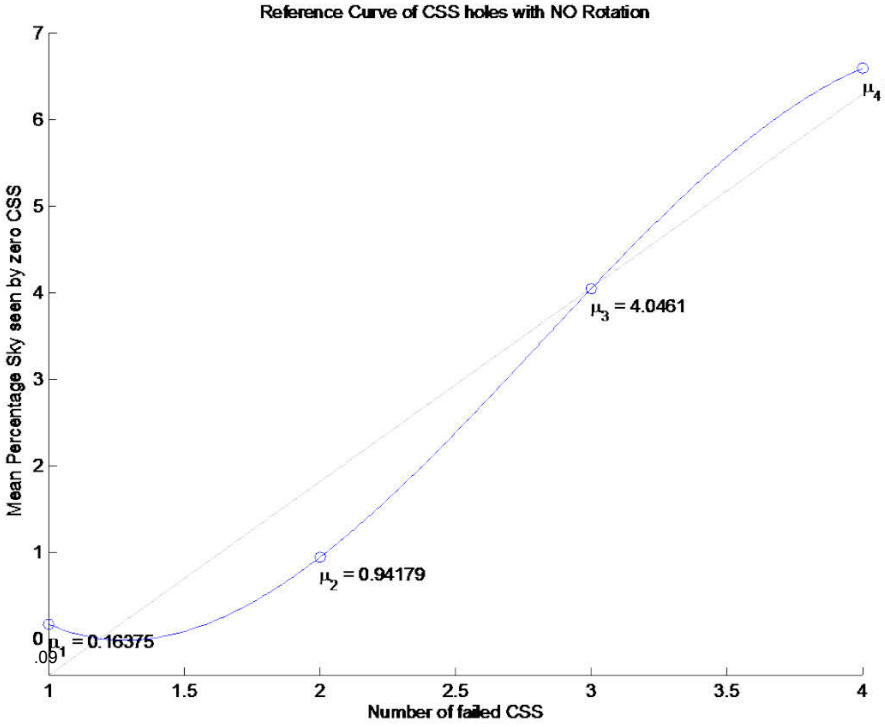
4 CSS Lost

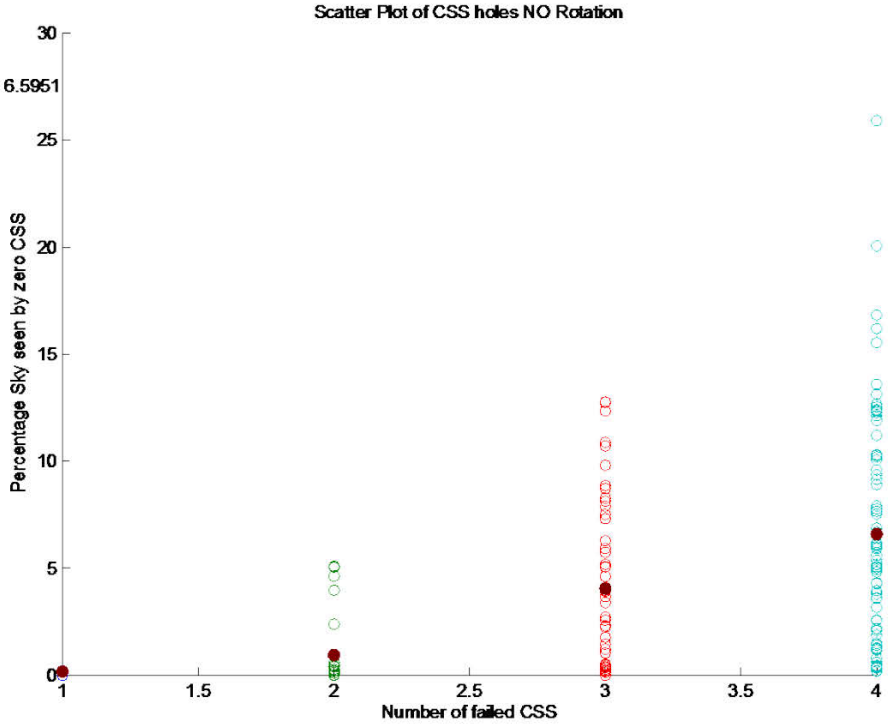
- Percentage seen by zero: 0.19% to 25.92%
- Percentage seen by one : 15.96% to 41.47%
- Percentage seen by two : 10.63% to 48.61%
- Percentage seen by three: 10.42% to 35.33%



The worst case with 25.92% seen by 0 and 15.96% seen by 1

Graphical Interpretation





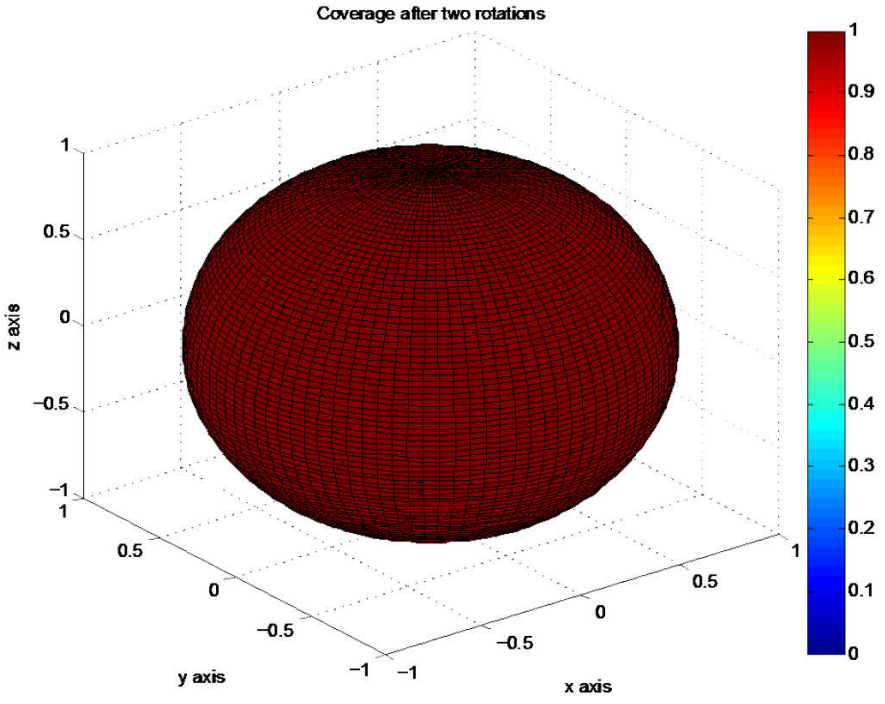
Summary: Static Case

- In general, sensors that have a higher percentage of unseen sky also have a high percentage of only one sensor viewing a portion of sky
- With one sensor lost, 0.2% of the sky is unseen
  - One rotation about YZ will correct for loss
- Average sun not seen is relatively low at the 2 CSS lost range, higher at the 3 -4 range

	Number of CSS Lost							
	1		2		3		4	
Percent Sky Seen	Min	Max	Min	Max	Min	Max	Min	Max
0	0.00	0.20	0.00	6.28	0.00	12.76	0.00	34.32
1	2.29	13.02	3.86	27.79	6.23	31.25	9.65	35.65
2	24.28	30.37	22.45	37.02	10.04	43.10	8.81	49.57
3	20.71	26.22	10.51	29.17	9.84	29.06	9.81	34.47
4	15.13	22.69	12.57	24.23	11.43	25.29	6.31	18.98
5	11.54	13.01	9.52	14.20	0.00	14.20	0.00	14.19
6	6.42	7.93	0.00	7.93	0.00	7.93	0.00	7.92
7	0.00	6.30	0.00	6.30	0.00	6.30	0.00	6.30

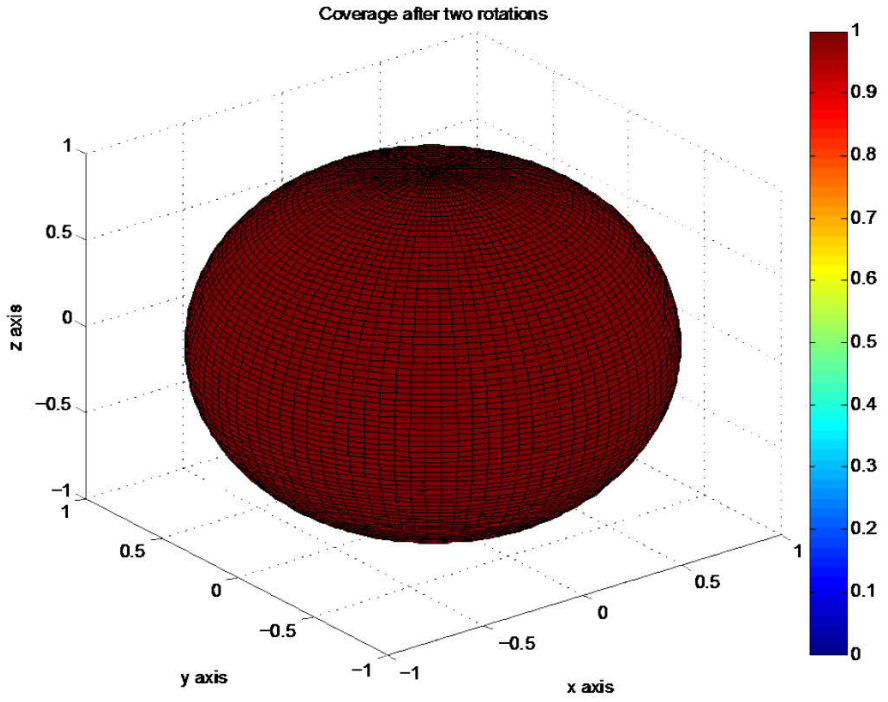
Mitigation by Rotation: YZ

- The rotation cases are observed independent of each other meaning that if one rotation does not see the entire sky, that does not necessarily mean that the other does not as well
- In fact the whole celestial sphere is covered in every failure scenario (1-4 CSS failed)



Mitigation by Rotation XZ

- Number of scenarios where two rotations did not cover the whole sky
- 0 of 162
- Highest Percent of sky not covered after two rotations
- 0.00
- The same percentages were seen for YZ rotation as well
- Both rotation techniques covered the entire celestial sphere



Summary: Mitigation by Rotation

- YZ and XZ Rotation covered the entire sky up to 4 failed CSS
  - One rotation about each axis was the pattern investigated
- Only after 5 failed sensors would we begin seeing this number drop
- Using the array sensors is being investigated

Sun Vector: No Failures

- When the sun is incident on a sensor a current is produced
- This current can be used to calculate a sun vector
  - Intensity of sunlight is modeled by the cosine of the angle difference in normal vectors
- Once the current is known, a pseudo inverse matrix is constructed, and the sun vector is calculated

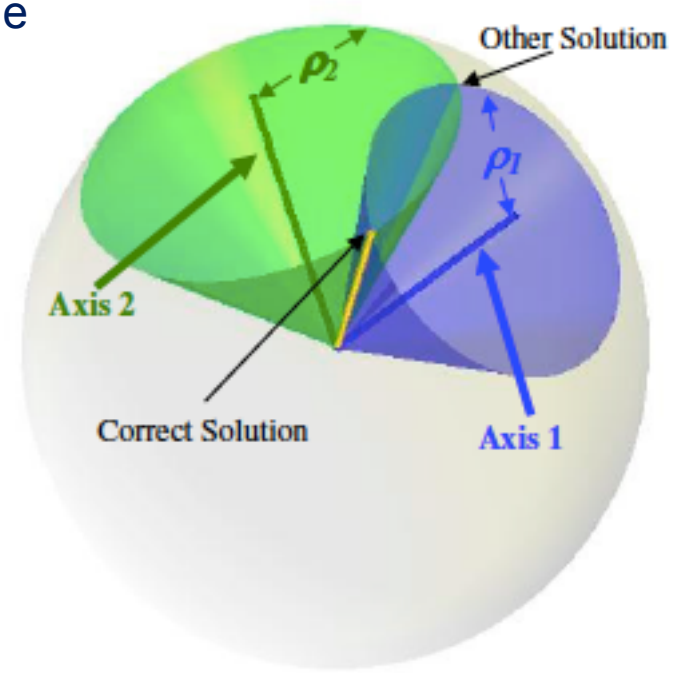


Figure 1. Illustration of Cones Method With Measured Angles ρ_1 and ρ_2 About 2 Axes

Sun Vector

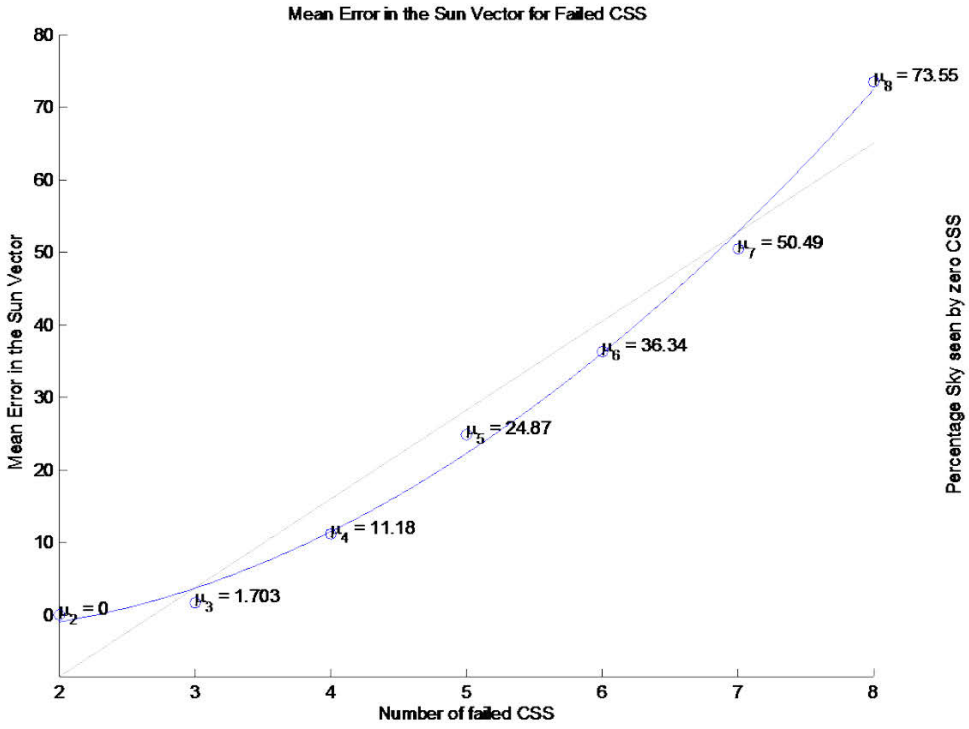
- 2 different approximations are made
  - Body Frame Sun Vector
  - Array Frame Sun Vector
- To check the accuracy, these two must be within reason
- The flight algorithm calculates the sun vector with up to 3 failed sensors with some dependencies
  - 3 body sensors
  - 3 array sensors
  - 1 array and 2 body
  - 2 array and 1 body

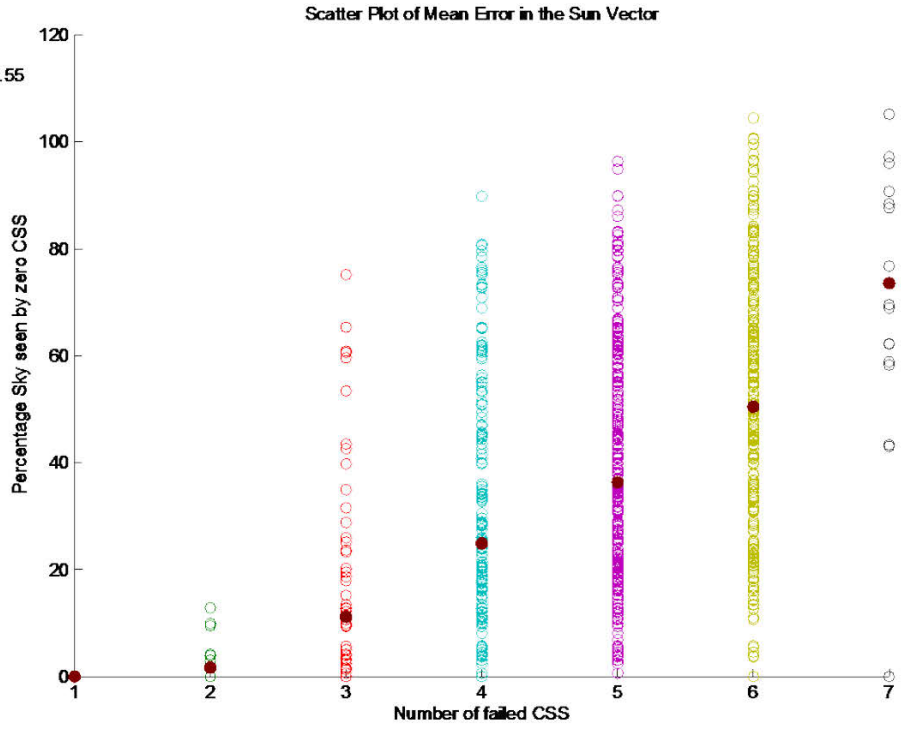
- The algorithm is biased towards the array sensors

Sun Vector: Error with 3 Array Sensors

Jet Propulsion Laboratory
California Institute of Technology

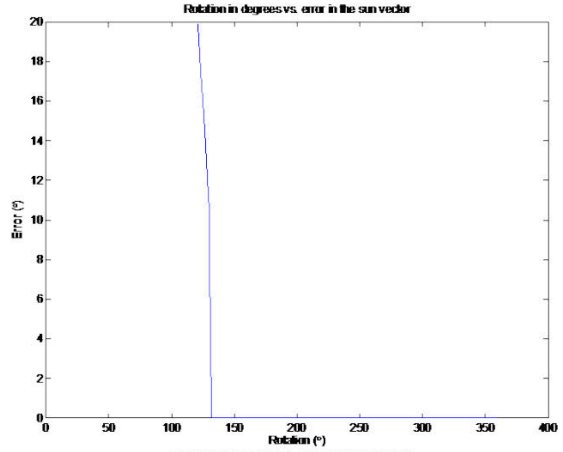
- Currently only the tetrahedron inputs into the sun vector





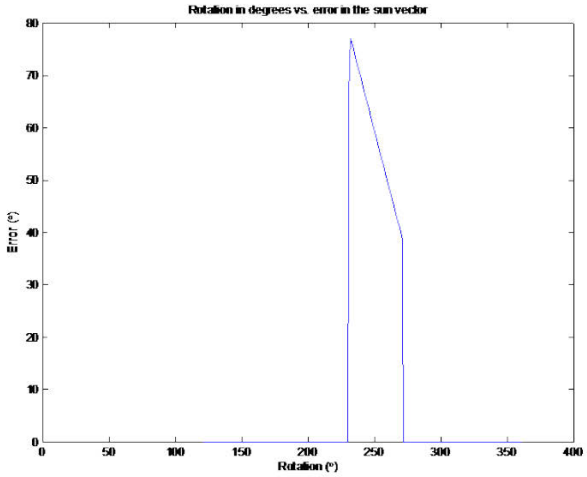
Sun Vector: Error with 3 Array Sensors

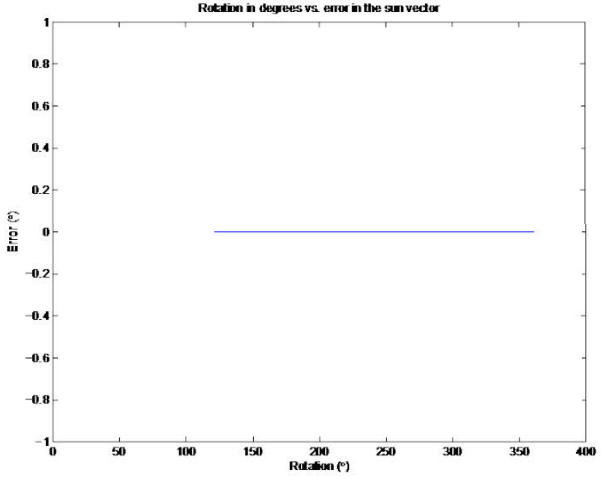
Jet Propulsion Laboratory
California Institute of Technology



4 Failed CSS

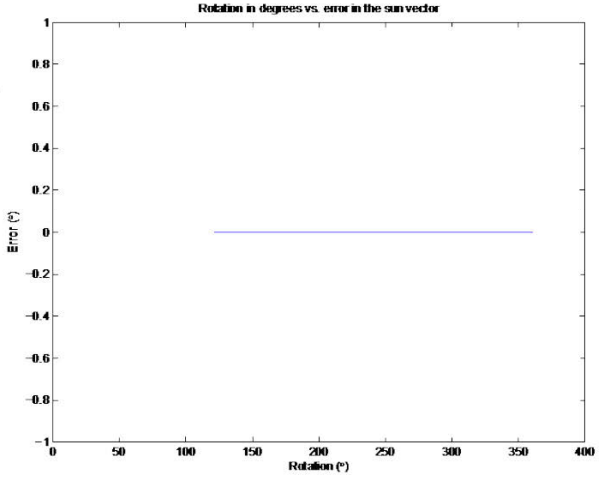
3 Failed CSS





1 Failed CSS

2 Failed CSS

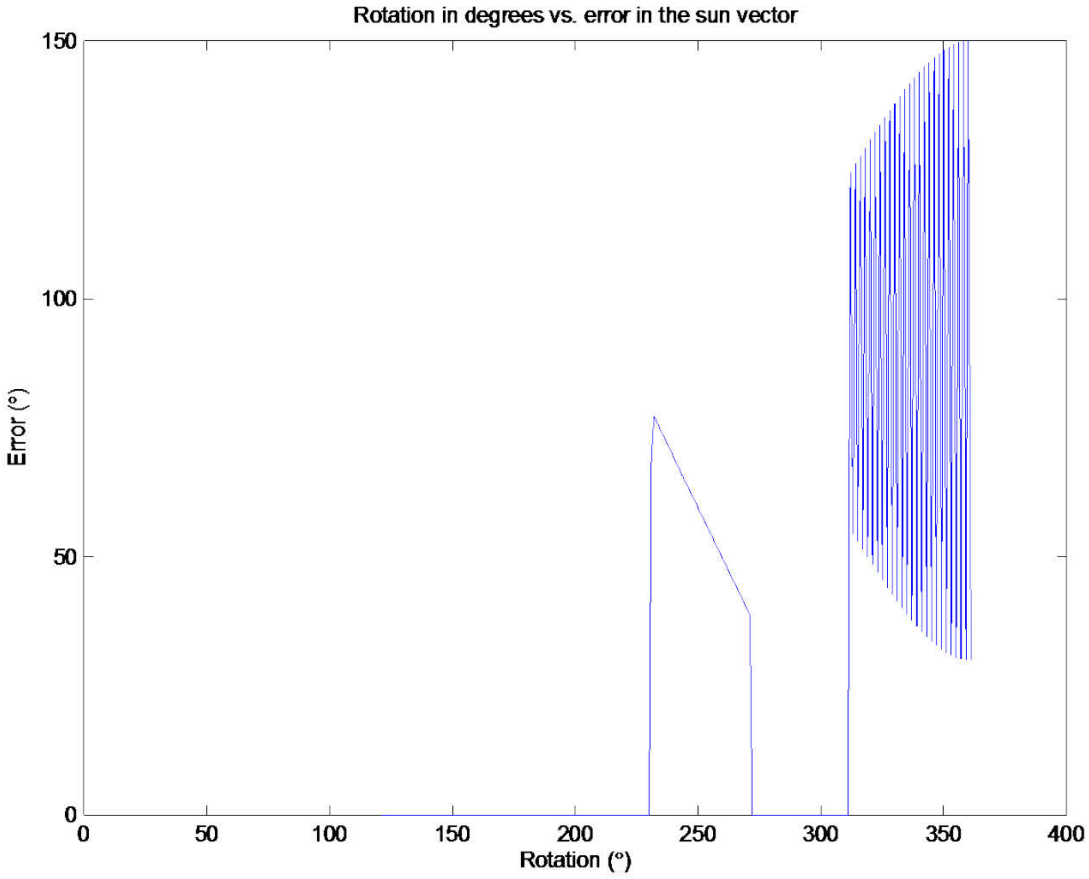


Most of the 2 failed CSS cases look like this, except for one…

Sun Vector: Error with 3 Array Sensors

Jet Propulsion Laboratory
California Institute of Technology

Death-Grip!



- This “fan-like” pattern begins to show itself at 2 failed sensors Algorithm is calculating a plane for the sun vector

Sun Vector: Error with 3 Array Sensors Summary

Jet Propulsion Laboratory
California Institute of Technology

- Overall it seems that the average errors in the sun vector are minimal for the description of the mission
- 2 max failed sun sensors is the limit
  - Dependent on the configuration of the 2 failed sensors where the result is a fan pattern
- With one failed CSS average error is 0
  - Either all body or all array are lit, no dependencies
- With two failed CSS average error is 1.7 deg
  - Fan shape presents itself
- Any higher than 3 and the average error becomes not large enough to ruin the mission, depending on the configuration

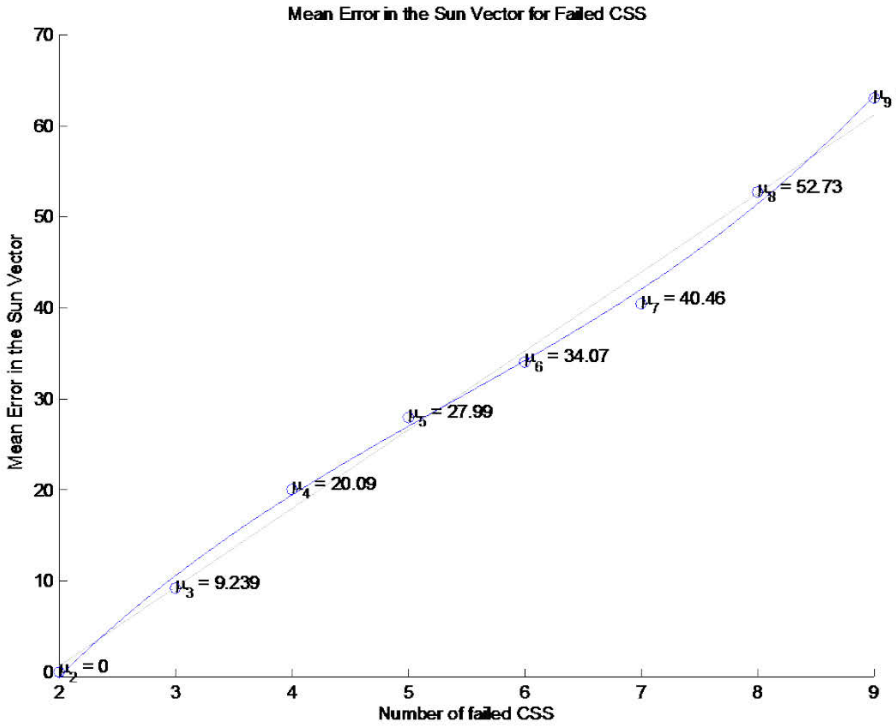
Sun Vector: Error with 5 Array Sensors

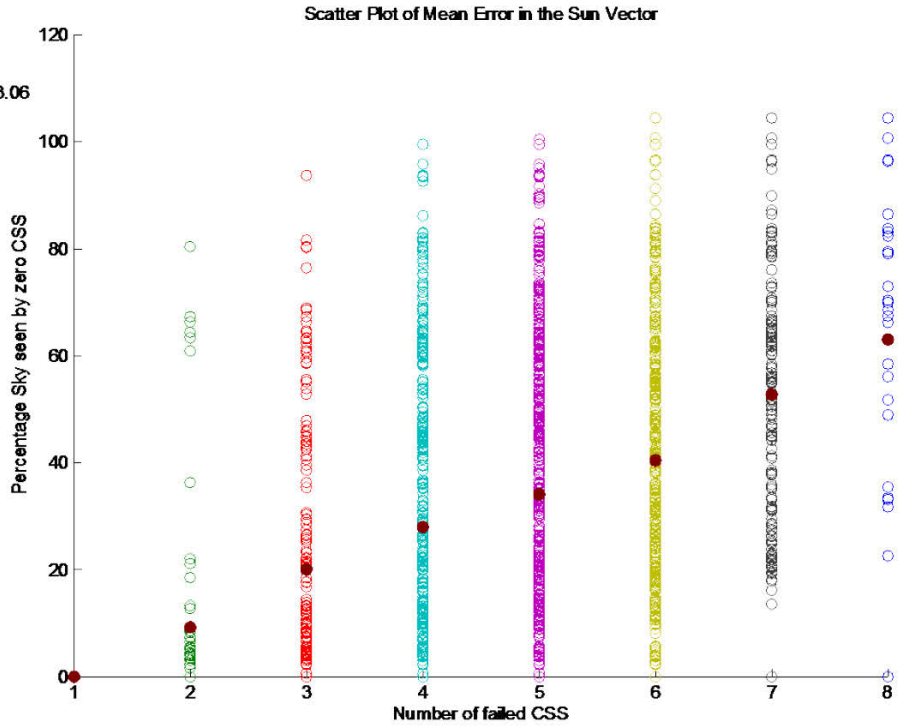
- What if 5 sensors are used to calculate the sun vector?
- This is a completely theoretical case!
- The method was interpolated from the logical flow of the algorithm

- Let's Go!

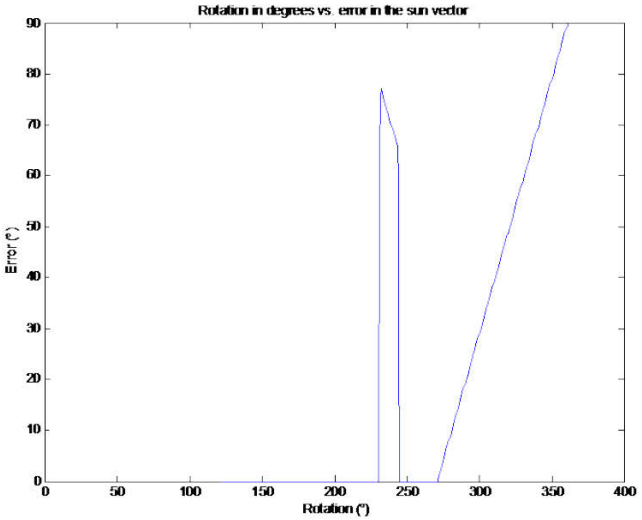
Sun Vector: Error with 5 Array Sensors

Jet Propulsion Laboratory
California Institute of Technology





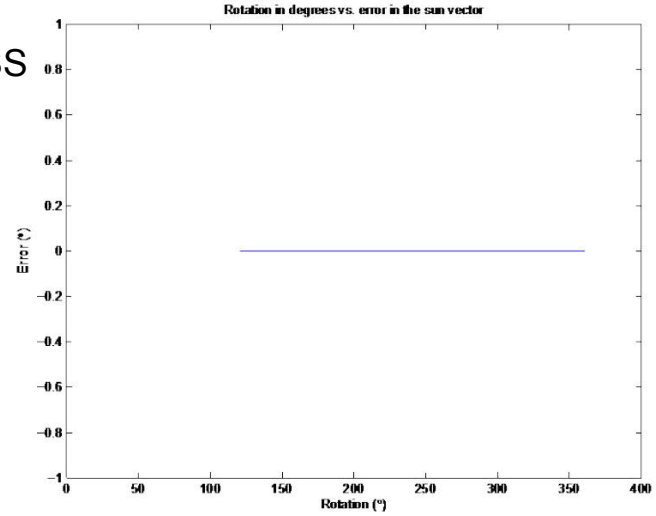
Sun Vector: Error with 5 Array Sensors

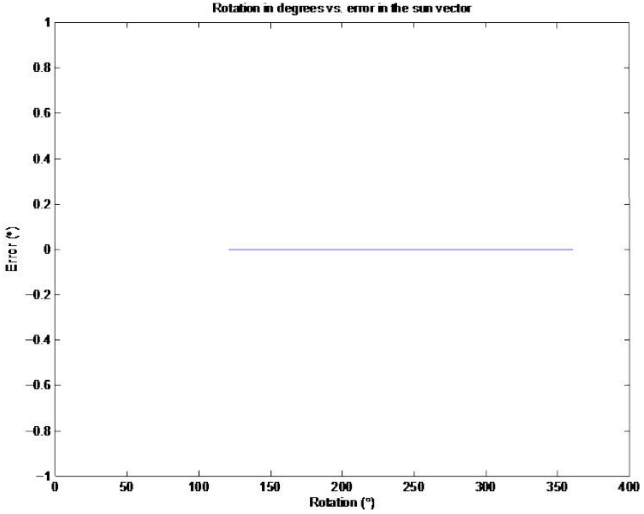


4 Failed CSS

Most of the 3 failed CSS cases look like this except for the fan pattern one..

3 Failed CSS

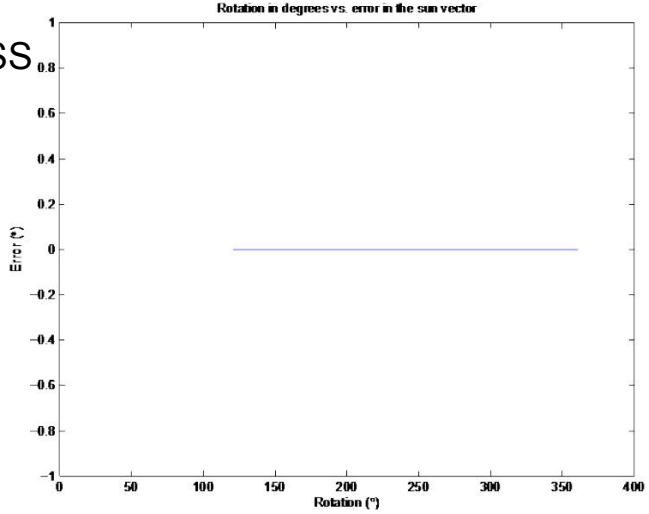




1 Failed CSS

Most of the 2 failed CSS cases look like this except for the fan pattern one..

2 Failed CSS



Sun Vector: Error with 5 Array Sensors

- The fan pattern can be created from any scenario where 2 CSS are failed
- The only benefit from using the extra sensors on the array comes in to play where the fan case is ignored
  - On average a gain of 3-4 degrees with the lower failure scenarios and 10 degrees at 7+ CSS failed
  - Not worth it to include them in the algorithm; the array provides enough redundancy as it is
- Important Note: The data is not complete for 5 array sensors!
  - Expensive computationally and only able to run a few scenarios
  - Higher number of failed CSS more accurate due to structure of Monte Carlo analysis

Sun Vector: Error with 5 Array Sensors Summary

- Not very much gain with using extra sun sensors in the sun vector estimation
  - Extra permutations need to be run
- Only error gain is seen at the 5+ failed CSS range
  - (knock on wood) That scenario is highly unlikely
- Bradford cells can be examined with these routines to determine if they will be better or worse than the OSC ones currently on the spacecraft

Acknowledgements

- Mentors: Brad Burt, Ray Welch

- This work was performed at the Jet Propulsion Laboratory of the California Institute of Technology under contract from the National Aeronautical and Space Administration

- Thank you!

Sources

- http://www.ai-solutions.com/Portals/0/AI%20Docs/Technical%20Paper%20Library%20pdf/2006/7.pdf