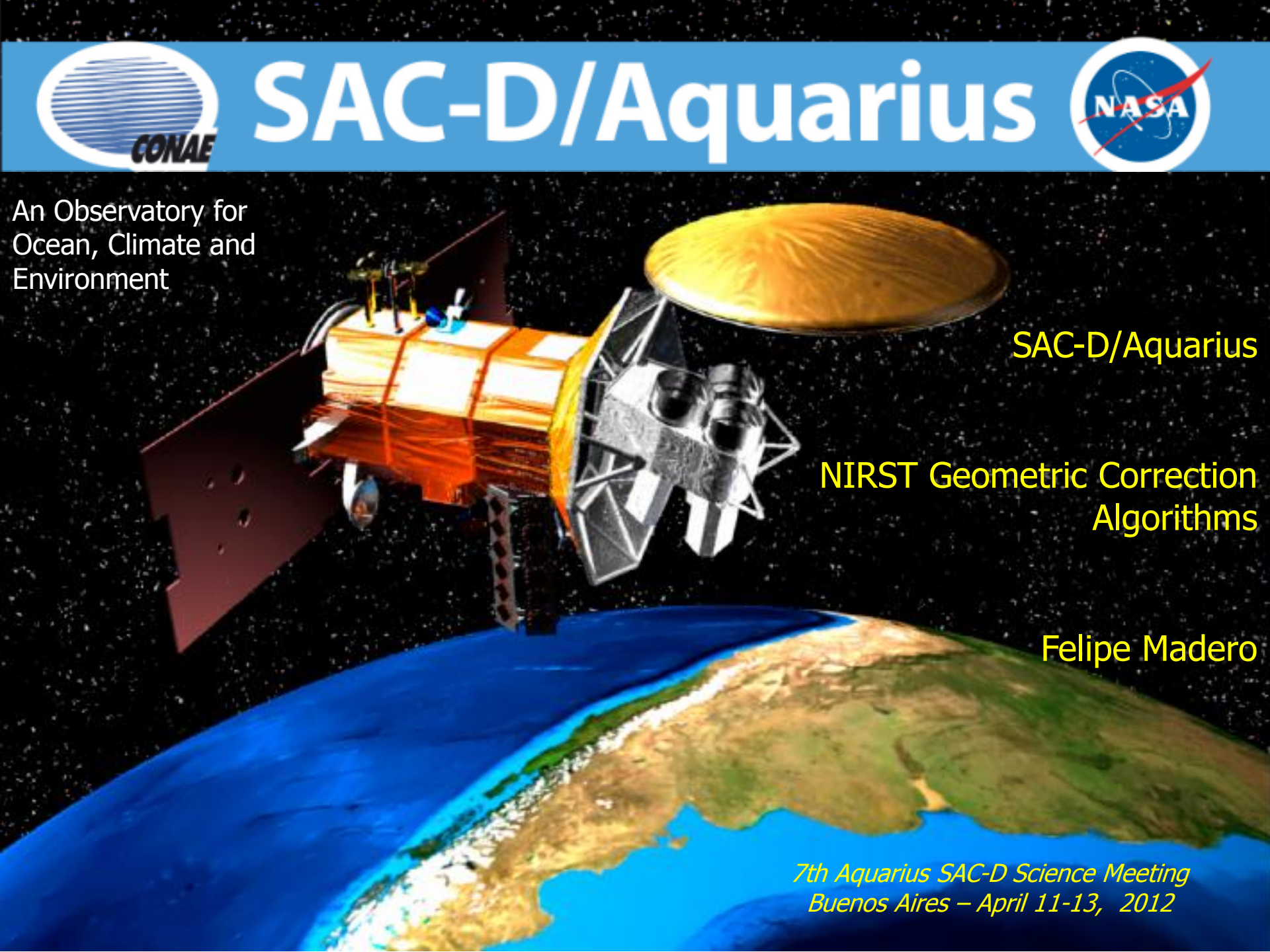
SAC-D/Aquarius

An Observatory for Ocean, Climate and Environment

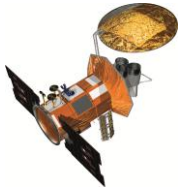


SAC-D/Aquarius

NIRST Geometric Correction Algorithms

Felipe Madero

7th Aquarius SAC-D Science Meeting
Buenos Aires – April 11-13, 2012

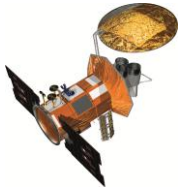


Geometric Processing Glossary

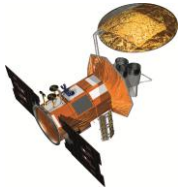
- Geometric Calibration: Activities done in order to tune the algorithms and models so as to obtain the expected geolocation results

- Geometric corrections: (Systematic) Processes applied to the sensor data, in order to:
  - Obtain knowledge of the geographic location of the measurements.
  - Solve geometric distortions introduced in the acquisition
  - Map the data to useful geographic reference systems

- Geolocation: (Systematic) Process applied to the sensor data in order to obtain knowledge of the geographic coordinates of the measurements



Geolocation Problem



Main reason for having good geolocation accuracy for this sensor

- Provision of data of good quality to the end users

- Radiometric calibration partially based in good geolocation

Geometric requirements for NIRST

- Related to the expected location and extension of the swath, and to the geometric resolution of the measurements:
  - Pixel size of 351 mt
  - Swath width of 182 km
  - Pointing +/- 30 degrees

- Related to the expected band corregistration (LWIR/MWIR):
- Inter band corregistration: 1 pixel

- Related to the expected geolocation accuracy:
- Pointing knowledge <= 0.03 degrees

Geolocation outputs

Geographic coordinates (geodetic latitude/longitude) for the center of the 512 measurements, for the 3 selected bands of the acquisition.

Auxiliary geographic parameters associated to this positions (range to spacecraft, and zenith/azimuth angle to spacecraft, moon, and sun)

Pre Launch

Pre launch measurements

- Focal plane, focal point, optics parameters measurements, at INO

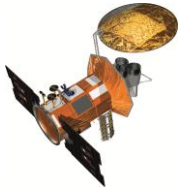
- FOV measurements, at INO

- Corregistration measurements, at INO

- Sensor mirror position measurements, at GEMA, UNLP

Pre launch data to processors input data

- Incorporating measured focal plane, focal point, and optics parameters, into a sensor model in order to generate initial set of line of sights for each detector.



Geolocation Algorithms

Geolocation Algorithms development

- ATBD/prototype library developed at the same time

- Geolocation results cross checked with standard geolocation libraries

- Software replacement: library first developed and validated using Python language. Replacement of bottlenecks using c++ code. Last version expected to be mainly c++ code, with python as control code.

Developed Geolocation Libraries

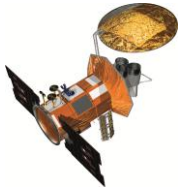
- Time system transformations
- Coordinates systems transformations
- Generic interpolation and smoothing
- Attitude data validation and processing
- State vector data validation and processing
- Intersection of line of sight with earth models
- Generation of auxiliary geolocation parameters

Geolocation Grid

- In order to reduce processing time due to geolocation, a geolocation grid is defined, which is a subsampling of the array of data obtained from acquisition.

- A good trade off between grid geolocation coordinates interpolation error, and time devoted to geolocation, was found by taking one sample per 20 detectors along the 512 detectors line, and taking all lines.

- Grid interpolation error, by using this parameters, is negligible.



Geolocation Steps: Input data from acquisition

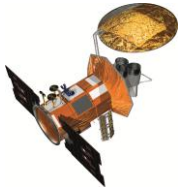
- Validate and filter time data. Fix time measurement errors by fitting the data to a first order polynomial (sample number to time).

- Validate and filter attitude data. No fixing of attitude measurement errors. Quaternions Interpolation.

- Validate and filter state vector data. Fix state vector error measurements by fitting the data to a suitable order polynomial (time to state vector).

Geolocation Steps: Pointing angle

- Get sensor pointing angle from telemetry (currently not working), from work order (database of known pointing angles obtained by sensor Engineering), or estimate pointing angle (more on this later). Generate a new SENSOR2PLAT rotation matrix by using this information.



Geolocation Steps: Get intersection parameters

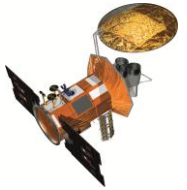
- For each sample of the geolocation grid:
  - Get spacecraft position in ECEF, by using time and state vector polynomial (J2000), and J2000 to ECEF transformation.
  - Get SENSOR2ECEF rotation matrix, by using calibrated SENSOR2PLAT rotation matrix, PLAT2J2000 rotation matrix from interpolated quaternions, and J2000 to ECEF transformation
  - Apply SENSOR2ECEF rotation matrix to the line of sight of the center of each measurement included in the sample

Geolocation Steps: Intersect line of sight with earth

- The spacecraft position in ECEF and the line of sight in ECEF defines a vector pointing to the earth, which is intersected to an earth model (WGS84), by solving a quadratic equation

- The resulting intersection point in ECEF is converted to geodetic latitude/longitude (elevation=0 due to using earth model)

- A planned further improvement is to intersect the line of sight with a digital elevation model, by an iteration procedure.

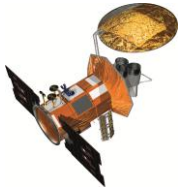


Geolocation Steps: Get auxiliary geolocation parameters

- Obtain moon and sun position in ECEF, by using standard tables, and J2000 to ECEF transformation

- Generate azimuth and elevation angles, between intersection point and spacecraft, sun, and moon, by using acquisition geometry

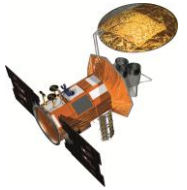
Corregistration



Product levels in relation to geometric correction

- L1A: geolocation parameters are included. Bands are corregistered.

- L1B2: resampling to map projection (grid) by using standard techniques, based in L1A, with recalculated geolocation parameters.



Current Corregistration

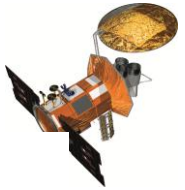
- Corregistering for L1A is currently not implemented.

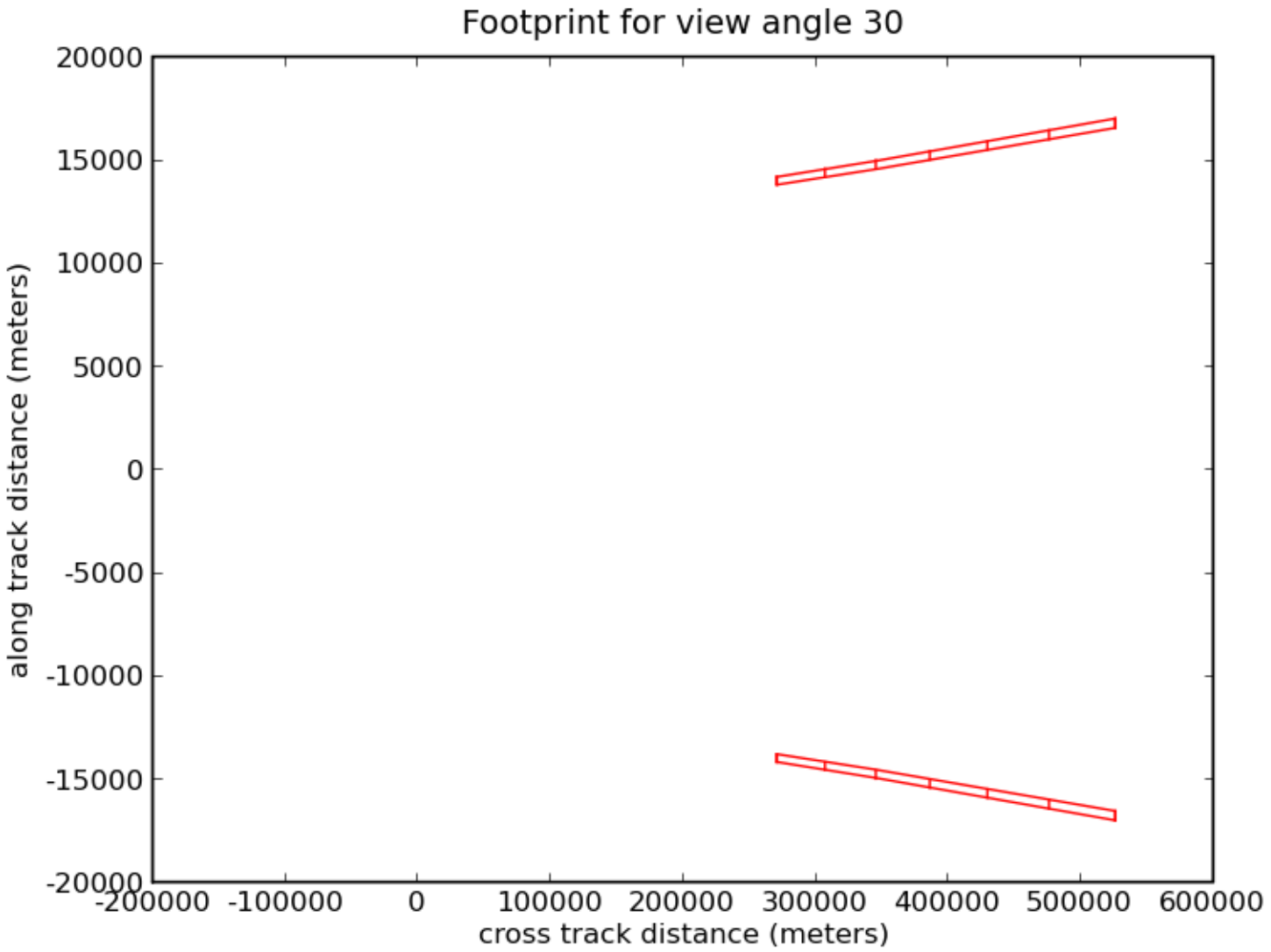
- A new version with includes corregistration based on geolocation is already implemented but is under revision.

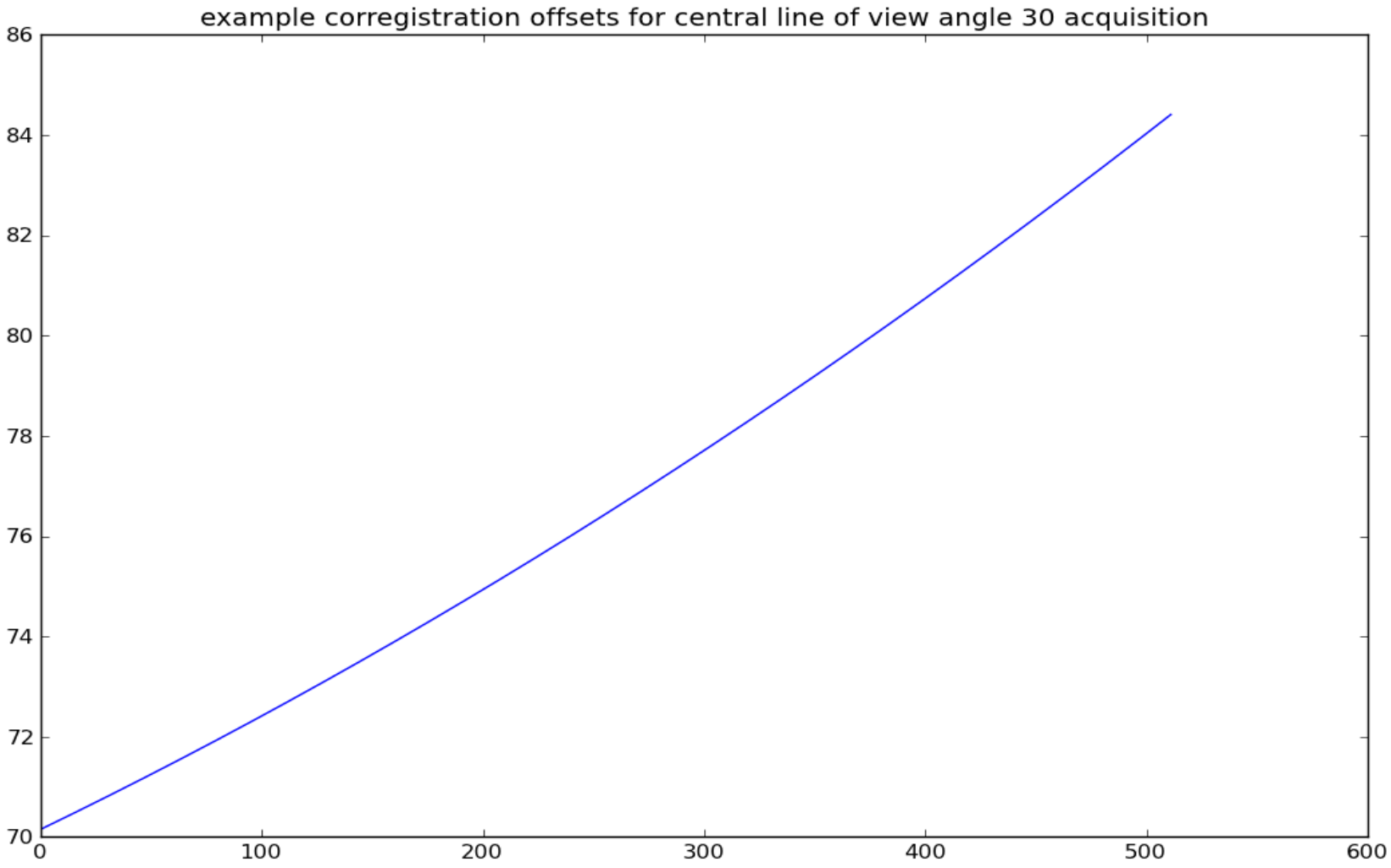
Corregistration problem

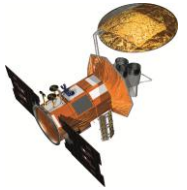
- Good corregistration of lines 1 to 3 is not an easy problem, due to:
  - Change of corregistration offset from detector 1 to 512
  - Change of corregistration offset from first line of acquisition to last line of acquisition
  - Difficulty to assess this corregistration changes by using only image correlation.

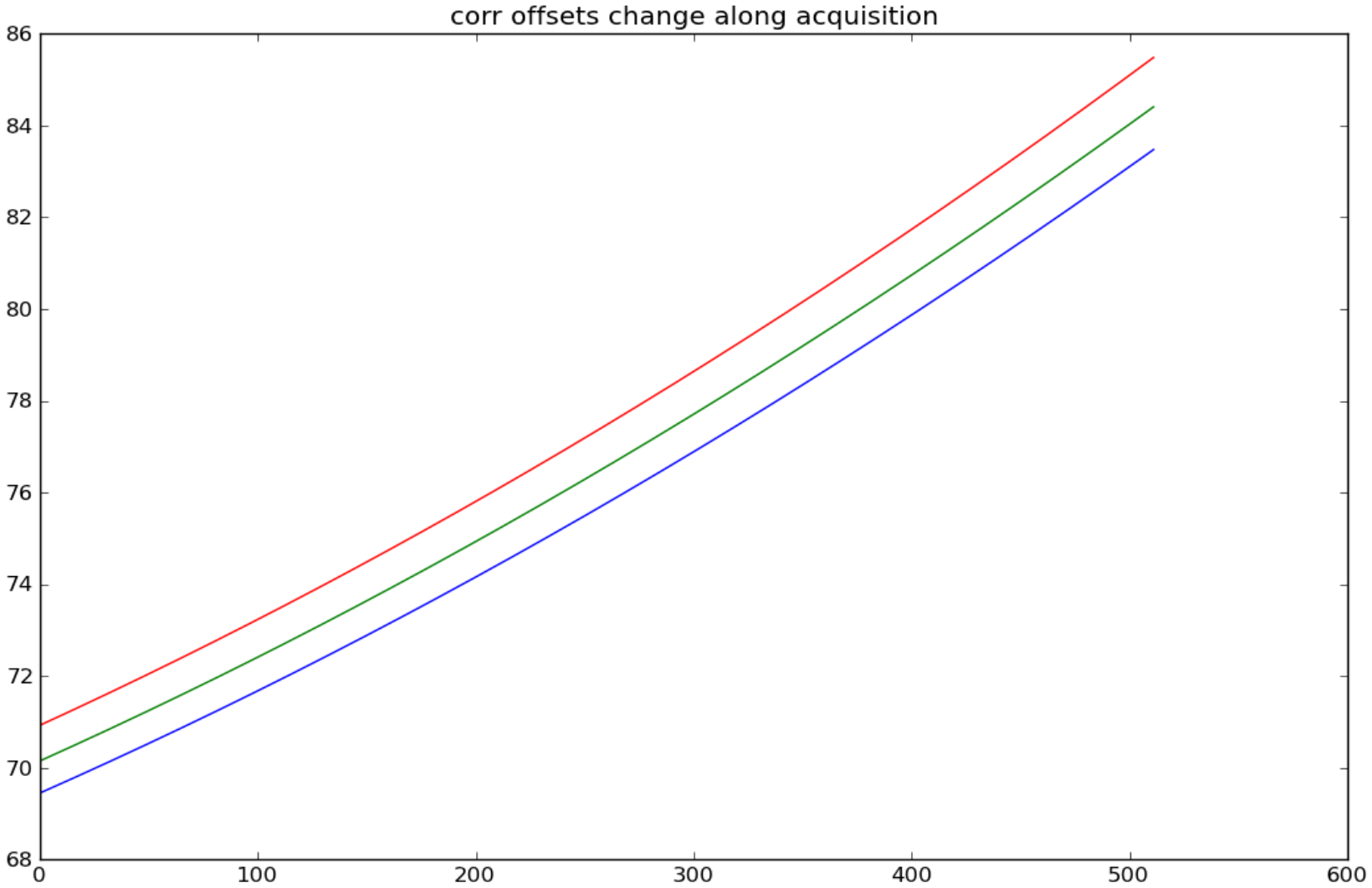
- The reason: acquisition configuration for different lines, for changing view angles




Footprint for view angle 30
20000
15000
10000
5000
0
-5000
-10000
-15000
-20000
along track distance (meters)
-200000
-100000
0
100000
200000
300000
400000
500000
600000
cross track distance (meters)


example corregistration offsets for central line of view angle 30 acquisition
86
84
82
80
78
76
74
72
70
0
100
200
300
400
500
600



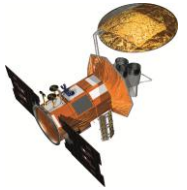

corr offsets change along acquisition
86
84
82
80
78
76
74
72
70
68
0
100
200
300
400
500
600

Corregistration solution

- Instead of using image correlation, which depends heavily on image characteristics, such as non homogeneous, correlation offsets are calculated based on current pointing angle, and geolocation obtained from it.

- For selected acquisition lines, forward line geolocation is fixed, and after line geolocation is recalculated, incrementing the line time, until both lines are corregistered (by now, the criteria is to minimize geolocation error among them).

- Corregistered images quality quality is improved by using this Mechanism.



Pointing Angle

Pointing angle estimation

- Pointing angle determination is a difficult problem.

- It is not possible to infer it from telemetry, and, due to control limitations, it is needed to use visual comparisons to estimate it.

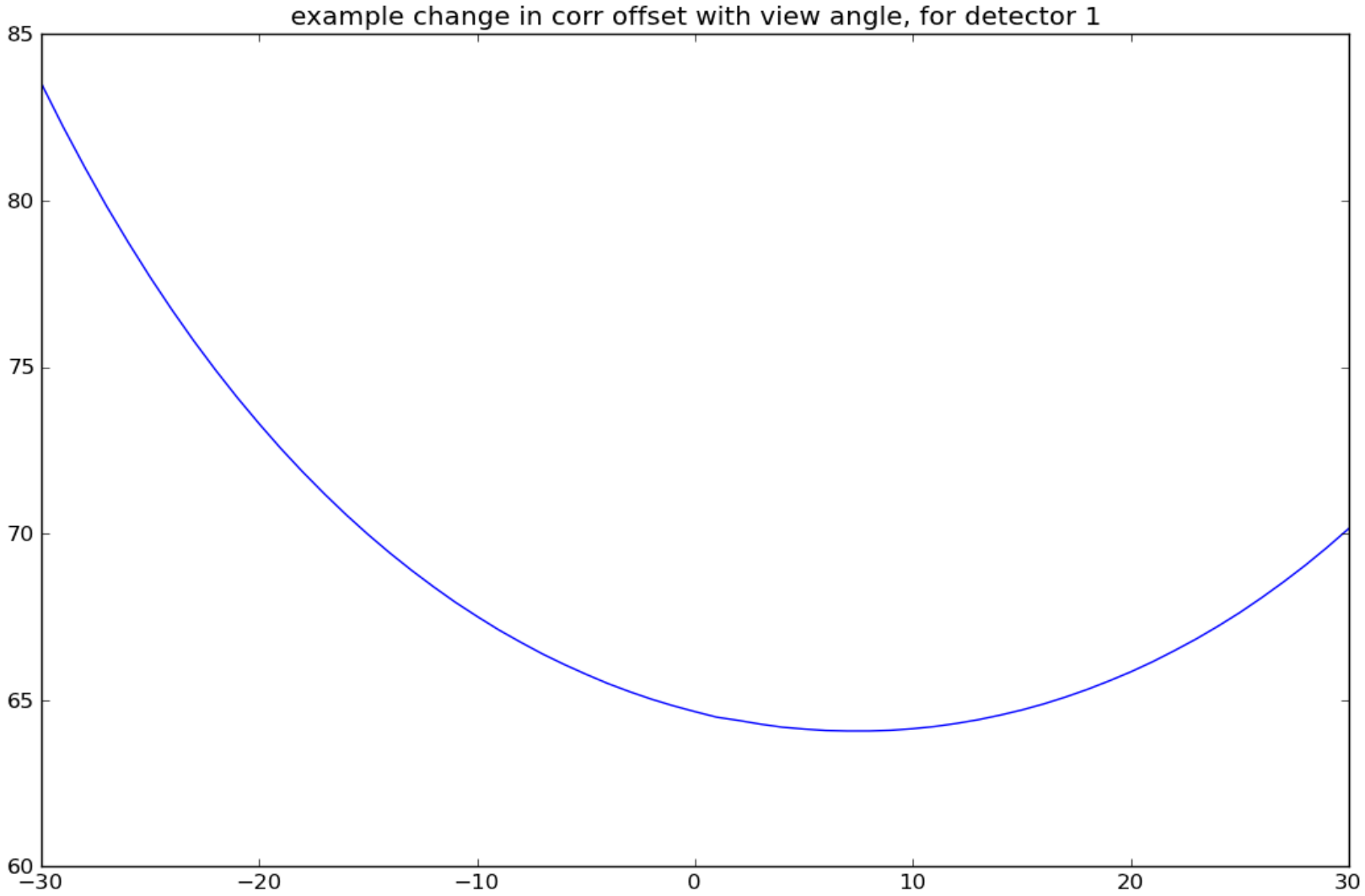
- Sensor engineering is currently estimating pointing angle by using a visual comparison mechanism, with a expected estimation error of 0.1 to 0.3 degrees. The estimation depends on the occurrence of visual clues in acquisition.

- After successful visual comparison, a processing may be done, and checked with manually selected control points, in order to refine the geolocation.

Pointing angle estimation from image correlation

- A parallel approach to estimate the pointing angle is being studied by the geometric calibration group.

- It is based in the inverse of the process used to corregistration: tries to infer the pointing angle from the corregistration offsets found in image correlation.


example change in corr offset with view angle, for detector 1
85
80
75
70
65
60
−30
−20
−10
0
10
20
30

Characteristics of the image correlation

- Image correlation used to do this is based in normalized image correlation (pearson coefficient).

- The fast implementation of it is a little different from standard fast normalized image correlation, in that it uses pre-calculated offsets over unnormalized correlation, instead of only gains.

- As corregistration offset changes along acquisition, correlation windows must be used, to avoid the undesirable weighting of good correlation zones on the global (full image) correlation calculation.

- In order to get sub-line accuracy, maximum is found by using the slope of the correlation results.

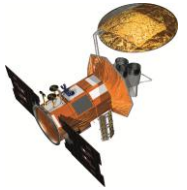
Not validated yet

- The method does not depend on coast clues on the images, but only on good non homogeneous zones.

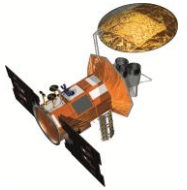
- Besides certain partial results are promising, both the geolocation model (mainly the alignment matrix in relation with the pointing angle), and the correlation process need to be improved.

- But it may be limited by the expected geolocation errors, and the achievable resolution in the obtained line offsets from the images.

- This is on going work, and it is expected to be included in the processing chain, if the method is found to be reliable and accurate enough.



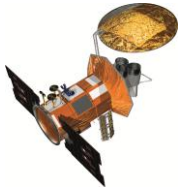
Geolocation Errors



Geolocation error from pointing knowledge requirement

- Pointing knowledge error: 0.03 degrees (max 915 mts)

- Bias plus random



Bias geolocation error

- Error in alignment measurement from ST cube to ref cube: 0.005 degrees (max 180 mts)

- Cyclic (orbital) Thermoelastic deformation errors between ref cube and NIRST reference system: 0.015 degrees (max 529 mts)

- Error in alignment between NIRST reference system and ref cube: 0.5 degrees (max 10 km)

- Goal to reduce it to 0 by post launch geometric calibration. Cyclic Thermoelastic deformation most difficult to correct.

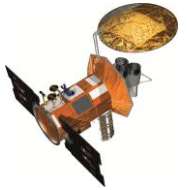
Random geolocation error

- Error in attitude measurement: 0.011 degrees (max 388 mts)

- Error in mirror pointing knowledge: 0.25 degrees (max 5 km)

- Goal is to have it as the final random error, after post launch geometric calibration.

Concluding



Known problems

- Geolocation errors due to star tracker error from moon interference

- Bugs in generation of sun and moon position in ECEF

Todo

- Known problems

- Determine if pointing angle estimation from correlation is viable, and in such a case, implement it.

- Complete post launch calibration

- Assess geolocation accuracy

- Implementation of intersection with digital elevation models

- Implementation of resampling by using point spread function