FGS2J_Calibration Python Tool

Full solution using multiple data points

Tony Sohn 5/27/2021

FGS to STA Calibration Algorithms

From NGAS Document "JWST FGS to STA Calibration Tool - User Guide"

The unit vector of the line of sight of the Guide Star in the FGS Guide 1 ICS frame is given by

$$u_{FGS} = R_{J\to FGS} R_{ECI\to J} R_{GS\to ECI}(RA, DEC, PA) R_{GS\_Apparent\to GS} u_{GS}$$

Where

u_{FGS} – Guide Star LOS unit vector in FGS Guide 1 ICS frame

$R_{J\to FGS}$ – J-frame to FGS alignment matrix

$R_{ECI\to J}$ – Rotation matrix from ECI frame to J-frame

$R_{GS\to ECI}(RA, DEC, PA)$ – Guide Star Attitude Matrix defined by (RA, DEC, PA)

$R_{GS\_Apparent\to GS}$ – Guide Star Apparent Attitude resulting from velocity aberration computed spacecraft velocity relative to the sun

u_{GS} – Unit vector of X axis

FGS to STA Calibration Algorithms

From NGAS Document “JWST FGS to STA Calibration Tool - User Guide”

$$u_{FGS} = R_{J \to FGS} R_{ECI \to J} R_{GS \to ECI}(RA, DEC, PA) R_{GS\_Apparent \to GS} u_{GS}$$

- u_{FGS} is a 3-element vector for a single star
- This matrix equation can not be solved with a single star
- Current code assumes single star input, so uses a workaround:
  - Use old $R_{J \to FGS}$ to calculate $u_{FGS,old}$
  - Calculate the angular offset θ = difference b/w $u_{FGS,measured}$ and $u_{FGS,old}$
  - Apply this θ to the old $R_{J \to FGS}$ matrix and obtain updated $R_{J \to FGS}$
  - The clocking angle is fixed and does NOT get constrained

New Python Tool for Full Matrix Solution

$$u_{FGS} = R_{J \to FGS} R_{ECI \to J} R_{GS \to ECI}(RA, DEC, PA) R_{GS\_Apparent \to GS} u_{GS}$$

- With multiple stars, we can solve the full matrix equation:
  - Build a matrix with multiple entires of $\boldsymbol{u_{FGS}}$
  - For each case, calculate the right hand side (except for $\boldsymbol{R_{J \to FGS}}$) = $\boldsymbol{A}$
  - Matrix equation becomes in the form: $\boldsymbol{R_{J \to FGS}} \cdot \boldsymbol{A} = \boldsymbol{u_{FGS}}$
  - Transpose both sides: $(\boldsymbol{R_{J \to FGS}} \cdot \boldsymbol{A})^{\mathbf{T}} = \boldsymbol{A}^{\mathbf{T}} \cdot \boldsymbol{R_{FGS \to J}} = \boldsymbol{u_{FGS}}^{\mathbf{T}}$
  - Overdetermined system with >3 stars
  - Solve using least-squares minimization (scipy.linalg.lstsq)
  - As with the case of original script, input PA does not affect outcome