

for__naor__20230126

January 26, 2023

0.1 Naor Movshovitz Uranus model results

Specify deviation of a gravity model specified by $J_2=J_{2p}$, $J_4=J_{4p}$, and $J_6=J_{6p}$, and a reference radius of 25559 km from R. French et al. (2023) Fit 13 for Uranus gravity field, specified by: $J_{2_obs} = (3510.653 \pm 0.389) \times 10^{-6}$ $J_{4_obs} = (-34.054 \pm 0.439) \times 10^{-6}$ $J_{6_obs} = -0.5 \times 10^{-6}$ (a priori fixed value) $\rho_{J_2J_4_obs} = 0.9859$

The error bars and correlation coefficient are derived from Monte Carlo simulations that account for estimated systematic offsets of ring centers of mass from their geometric centers, and their uncertainties, including individual uncertainties of the semimajor axes of the rings. This version does not include systematic corrections to the ring radius scale.

Usage:

`r(J2p,J4p,J6p)` returns standard deviation of input model J_2 , J_4 , J_6 values (named J_{2p} , J_{4p} , J_{6p}) from French et al. Fit 13 nominal gravity field solution, accounting for correlation of J_6 with J_2 and J_4 .

Test cases: Fit 13: - specify Fit 13 values of J_2 , J_4 , and $J_6=0.5e-6$; expect $r=0$ Two points on 1-sigma error ellipse from MC run, no J_6 offset, expect $r=1$

Fit 6: $J_6=0$, J_2 and J_4 from Fit 6; expect $r=0$ Fit 8: $J_6=1e-6$, J_2 and J_4 from Fit 8; expect $r=0$

The calculated values of x and y should be small in these test cases - they represent the difference between the adjusted input J_2' and J_4' and our adopted solution.

Revisions: 2023 Jan 26 - rfrench@wellesley.edu - original version

```
[42]: import numpy as np
# All values from Table 14 of uringpolegrav_v1.3 dated 2023 Jan 26
# Eq. C4, with \Delta a=0
def J2corr(J6p):
    J2= 3510.653e-6 + 0.40036 * (J6p-0.5e-6)
    return J2

# Eq. C5, with \Delta a=0
def J4corr(J6p):
    J4= -34.054e-6 + 1.0788 * (J6p-0.5e-6)
    return J4

# Eq. C6, with \Delta a=0
def r(J2p,J4p,J6p):
```

```

# terms in covariance matrix
a = 0.389e-6**2
b = 0.9859 * 0.389e-6 * 0.439e-6
c = 0.439e-6**2
# Eqs. C11, C12 eigenvalues
lambda_1 = (a+c)/2 + np.sqrt(((a-c)/2)**2 + b**2)
lambda_2 = (a+c)/2 - np.sqrt(((a-c)/2)**2 + b**2)
# Eq. C17 - rotation of error ellipse in J2,J4 plane
theta = np.arctan2(lambda_1 - a, b)
# Eqs. C15, C16
x = J2p - J2corr(J6p)
y = J4p - J4corr(J6p)
# Eqs. C13, C14
xp = x * np.cos(theta) + y * np.sin(theta)
yp = -x * np.sin(theta) + y * np.cos(theta)
# Eq. C6 - standard deviation of input J2', J4', J6' from occultation gravity
↳ solution
r = np.sqrt(xp**2/lambda_1 + yp**2/lambda_2)
return r

```

```

[55]: # test the routine
# Fit values from Fit 13 - adopted solution
J2_obs=3510.653e-6
J4_obs=-34.054e-6
J6_obs=0.5e-6

# in each case, specify a trio of J2,J4,J6:
J2p = J2_obs
J4p = J4_obs
J6p = J6_obs

print('Fit 13: Centered on best fit, no offsets:')
print('r = ', r(J2p, J4p, J6p), 'should be close to 0.0')
print(' ')

# sample J2 and J4 point on 1-sigma ellipse from Monte Carlo run
J2p = 3511.0403e-6
J4p = -33.615176e-6
J6p = J6_obs
print('Sample point on 1-sigma error ellipse:')
print('r = ', r(J2p, J4p, J6p), 'should be close to 1.0')
print(' ')

# another sample J2 and J4 point on 1-sigma ellipse from Monte Carlo run
J2p = 3511.0100288843000e-6
J4p = -33.626751350831206e-6
J6p = J6_obs

```

```

print('Sample point on 1-sigma error ellipse:')
print('r = ',r(J2p,J4p,J6p),'should be close to 1.0')
print(' ')

# Fit 6 (J6=0)
J2p = 3510.454e-6
J4p = -34.592e-6
J6p = 0
print('Fit 6:')
print('r = ',r(J2p,J4p,J6p),'should be close to 0.0')
print('x=J2p-J2corr(J6p)',J2p-J2corr(J6p),' should be close to 0')
print('y=J4p-J4corr(J6p)',J4p-J4corr(J6p),' should be close to 0')
print(' ')

# Fit 8 (J6=1e-6 )
J2p = 3510.853e-6
J4p = -33.517e-6
J6p = 1e-6
print('Fit 8 J6P=1e-6:')
print('r = ',r(J2p,J4p,J6p),'should be close to 0.0')
print('x=J2p-J2corr(J6p)',J2p-J2corr(J6p),' should be close to 0')
print('y=J4p-J4corr(J6p)',J4p-J4corr(J6p),' should be close to 0')
print(' ')

```

Fit 13: Centered on best fit, no offsets:

r = 0.0 should be close to 0.0

Sample point on 1-sigma error ellipse:

r = 1.0014287352596614 should be close to 1.0

Sample point on 1-sigma error ellipse:

r = 1.0046248586540873 should be close to 1.0

Fit 6:

r = 0.003256936518861295 should be close to 0.0

x=J2p-J2corr(J6p) 1.1800000000554423e-09 should be close to 0

y=J4p-J4corr(J6p) 1.4000000000030698e-09 should be close to 0

Fit 8 J6P=1e-6:

r = 0.029948016207693662 should be close to 0.0

x=J2p-J2corr(J6p) -1.7999999971443637e-10 should be close to 0

y=J4p-J4corr(J6p) -2.3999999999171e-09 should be close to 0

[]: