

# AA279D Project

Paxton Scott and Chris Osgood



AA 279D - Spacecraft Formation-Flying and Rendezvous  
Stanford University

## Revision History

Table 1: Summary of project revisions.

Rev	Changes
PS1	<ul style="list-style-type: none"> <li>- Created document</li> <li>- Added problem set 1 material</li> <li>- Added survey of DSS's to choose from</li> <li>- Chose our DSS and provided key information on it</li> <li>- Performed absolute orbit determination</li> <li>- Updated plots for better readability</li> <li>- Fixed mean vs osculating issue</li> </ul>
PS2	<ul style="list-style-type: none"> <li>- Added problem set 2 material</li> <li>- Performed relative motion simulations</li> <li>- Updated delta-v calculation to correctly adjust semi-major axis.</li> </ul>
PS3	<ul style="list-style-type: none"> <li>- Added problem set 3 material</li> <li>- Updated RTN graphs to have correct axes and correct ROE vector typos.</li> </ul>
PS4	<ul style="list-style-type: none"> <li>- Added problem set 4 material</li> <li>- Updated calculations in section 4.1.6 to accurately asses delta-v lower bound.</li> </ul>
PS5	<ul style="list-style-type: none"> <li>- Added problem set 5 material</li> <li>- Updated formation keeping to correct for <math>\delta a</math> and <math>\delta \lambda</math></li> <li>- Updated closed form analysis accordingly</li> </ul>
PS6	<ul style="list-style-type: none"> <li>- Added problem set 6 material</li> <li>- Updated plots for better readability</li> </ul>
PS7	<ul style="list-style-type: none"> <li>- Added problem set 7 material</li> <li>- Updated conclusion of controls</li> </ul>
PS8	<ul style="list-style-type: none"> <li>- Added problem set 8 material</li> <li>- Updated plots</li> <li>- Updated noise covariance matrices</li> </ul>
PS9	<ul style="list-style-type: none"> <li>- Added problem set 9 material</li> <li>- Performed final project edits</li> <li>- Added conclusion and final remarks</li> </ul>

## Contents

<b>1 Problem Set 1</b>	<b>1</b>
1.1 Problem 1: Your Mission, Your Challenge . . . . .	1
1.1.1 Survey of current DSS's . . . . .	1
1.1.2 Our Chosen DSS . . . . .	2
1.2 Problem 2: Absolute Orbit Simulation . . . . .	3
1.2.1 Answers . . . . .	3
<b>2 Problem Set 2</b>	<b>10</b>
2.1 Problem 1: Everything is Relative . . . . .	10
2.1.1 Answers . . . . .	10
<b>3 Problem Set 3: Linearized relative orbit in Cartesian coordinates.</b>	<b>15</b>
3.1 Problem 1. We are Close in Near-Circular Orbits. . . . .	15
3.1.1 Answers . . . . .	15
3.2 Problem 2. We are Close in Eccentric Orbits. . . . .	19
3.2.1 Answers . . . . .	19
<b>4 Problem Set 4: Linearized relative orbit motion in relative orbit elements.</b>	<b>34</b>
4.1 Problem 1. Relative Orbits. . . . .	34
4.1.1 Initial Chief OE . . . . .	34
4.1.2 Initial Deputy ROE . . . . .	34
4.1.3 Baseline Simulation . . . . .	34
4.1.4 RTN Plots for Baseline Simulation . . . . .	40
4.1.5 QNS ROE Plots for Baseline Simulation . . . . .	44
4.1.6 Minimum Change to Remove J2 Secular Effects . . . . .	46
4.1.7 Simulation with No J2 Secular Effects . . . . .	47
4.1.8 Linearized Relative Motion Under J2 . . . . .	49
<b>5 Problem Set 5: Implement impulsive control law</b>	<b>51</b>
5.1 Problem 1. What are the control orbits? . . . . .	51
5.1.1 Significant operational modes. . . . .	51
5.1.2 Mode definitions . . . . .	51

5.1.3	Controls while operating in mode . . . . .	52
5.1.4	Controls for switching between modes . . . . .	52
5.1.5	What actuators are you considering? . . . . .	52
5.1.6	Given the control requirement, what absolute or relative orbit dynamics models are needed? . . . . .	52
5.2	Problem 2. Impulsive Control Law. . . . .	53
5.2.1	Implementation and Justification . . . . .	53
5.2.2	Impulsive control laws for formation keeping and and reconfiguration. . . . .	53
5.2.3	Formation reconfiguration . . . . .	53
5.2.4	Least squares Reconfiguration Maneuver . . . . .	54
5.2.5	Formation Keeping . . . . .	55
5.2.6	RTN visualization . . . . .	57
5.2.7	Delta-V Closed Form Solutions and Lower Bound and Comparisons . . . . .	57
5.2.8	Control systems over time . . . . .	59
<b>6</b>	<b>Problem Set 6: Implement continuous control law</b>	<b>65</b>
6.1	Problem 1: Continuous Control Law . . . . .	65
6.1.1	Implementation and justification . . . . .	65
6.1.2	Discuss performance and visualize results. . . . .	68
<b>7</b>	<b>Problem Set 7: Finalize relative orbit control laws, design, and start implementing relative navigation system.</b>	<b>74</b>
7.1	Finalize relative orbit control . . . . .	74
7.2	Navigation System Design . . . . .	74
7.2.1	State Representation . . . . .	74
7.2.2	Dynamics Model . . . . .	74
7.2.3	Linearized Dynamics Model . . . . .	75
7.2.4	Onboard sensors . . . . .	75
7.2.5	Nonlinear measurement model . . . . .	75
7.2.6	Associated sensitivity matrix which you would use to perform the EKF . . . . .	76
7.2.7	Our UKF Filtering Scheme . . . . .	76
7.2.8	Simple test cases. . . . .	77

<b>8 Problem Set 8: Implement navigation filter.</b>	<b>78</b>
8.1 Ground truth generation . . . . .	78
8.2 Convert ground truths into measurements by adding Gaussian noise. . . . .	78
8.3 Set initial estimate and covariance. . . . .	78
8.4 Define process and measurement noise covariance . . . . .	78
8.5 Combine all the parameters to mechanize the Unscented Kalman Filter . . . . .	78
8.6 Functionality and performance of the UKF . . . . .	78
<b>9 Problem Set 9: Final Project and Closing the loop between Nav and Control</b>	<b>83</b>
9.1 Problem 1: Catching Up . . . . .	83
9.1.1 General . . . . .	83
9.1.2 PSet 1 . . . . .	83
9.1.3 PSet 2 . . . . .	83
9.1.4 PSet 3 . . . . .	83
9.1.5 PSet 4 . . . . .	83
9.1.6 PSet 5: Impulsive control . . . . .	84
9.1.7 PSet 6: Continuous control . . . . .	84
9.1.8 PSet 7: Navigation filter setup . . . . .	84
9.1.9 PSet 8: UKF . . . . .	84
9.2 Problem 2: Integration of Navigation and Control . . . . .	85
9.2.1 Control law selection . . . . .	85
9.2.2 Baseline . . . . .	85
9.2.3 Corrupt ground truth for controller . . . . .	86
9.2.4 Closing the loop between Navigation and Controller (with analysis) . . . . .	88
9.2.5 Performance metrics . . . . .	91
9.3 Problem 3: Extension of our choosing . . . . .	97
9.4 Problem 4: Conclusion and final remarks . . . . .	98
9.4.1 Brief Summary . . . . .	98
9.4.2 Results Assessment . . . . .	99
9.4.3 Lessons Learned . . . . .	99
9.4.4 Future Work . . . . .	99
9.5 Acknowledgements . . . . .	100

**References**

**101**

## List of Figures

1	Orbit simulation performed using ECI position and velocity . . . . .	4
2	Orbit simulation performed using analytical Keplerian propagation, excluding J2 effects. . . . .	4
3	Errors (in RTN) between ECI simulation and analytical Keplerian propagation, excluding J2 effects. . . . .	5
4	Orbital elements over time without the J2 effect. . . . .	6
5	Angular momentum, eccentricity vector, and mechanical energy over time without the J2 effect. . . . .	6
6	Orbital elements over time with the J2 effect. . . . .	7
7	Angular momentum, eccentricity vector, and mechanical energy over time with the J2 effect. . . . .	7
8	Orbit using Keplerian Simulation, Including Mean J2 Effects . . . . .	8
9	Orbital Elements using Keplerian Simulation and ECI Simulation, Including Mean J2 Effects . . . . .	8
10	Orbital Element Errors between Keplerian Simulation and ECI Simulation, Including Mean J2 Effects . . . . .	9
11	Relative motion generated from our numerical relative motion simulator, ignoring perturbations. . . . .	11
12	Relative orbit from fundamental orbital differential equations of absolute motion. . . . .	11
13	Error between relative motion numerical simulation and orbit from fundamental differential equations. . . . .	12
14	Error between the relative motion numerical simulation and the fundamental differential equations for absolute motion simulation when the deputy and the chief have a different semi-major axis. . . . .	12
15	Relative orbit in the RTN frame after introducing a change in semi-major axis. . . . .	13
16	Relative position and velocity before and after applying an along track maneuver to cancel the relative difference in semi-major axis. . . . .	14
17	Semi-major axis of chief and deputy over time. We see that the semi-major axis of the deputy decreases at the point of our along track burn. . . . .	14
18	Relative position in the RTN frame. We can see that the propagation using the standard HCW solution results in substantial drift along the T axis. . . . .	17
19	The relative velocity is bounded. . . . .	18
20	We can also observe the drift in the T direction on the 2D graphs. . . . .	18
21	Once we impose the additional constraint of $\dot{y}(0) = -2nx(0)$ we produce bounded relative motion. . . . .	19
22	RTN relative position and velocity in 3D space, using the YA solution. . . . .	20
23	RTN relative position and velocity on each RTN plane, using the YA solution. . . . .	21

24	RTN relative position and velocity in 3D space, using the YA solution and the linear geometric mapping solution. . . . .	22
25	RTN relative position and velocity on each RTN plane, using the YA solution and the linear geometric mapping solution. . . . .	22
26	RTN relative position and velocity in 3D space, using the YA solution, the linear geometric mapping solution, and the solution based on non-linear differential equations. . . . .	23
27	RTN relative position and velocity on each RTN plane, using the YA solution, the linear geometric mapping solution, and the full nonlinear solution. . . . .	24
28	RTN error between the YA solution and the full nonlinear solution. . . . .	24
29	RTN error between the linear geometric mapping solution and the full nonlinear solution. . . . .	25
30	RTN relative position and velocity in 3D space, using the YA solution, the linear geometric mapping solution, and the full nonlinear solution for orbits with non-zero $\delta a$ . . . . .	26
31	RTN relative position and velocity on each RTN plane, using the YA solution, the linear geometric mapping solution, and the full nonlinear solution for orbits with non-zero $\delta a$ . . . . .	27
32	RTN error between the YA solution and the full nonlinear solution for orbits with non-zero $\delta a$ . . . . .	28
33	RTN error between the linear geometric mapping solution and the full nonlinear solution for orbits with non-zero $\delta a$ . . . . .	29
34	RTN relative position and velocity in 3D space, using YA solution and linear geometric mapping solution and full nonlinear solution for orbits with high eccentricity. . . . .	30
35	RTN relative position and velocity on each RTN plane, using the YA solution, the linear geometric mapping solution, and the full nonlinear solution for orbits with high eccentricity. . . . .	31
36	RTN error between the YA solution and the full nonlinear solution for orbits with high eccentricity. . . . .	32
37	RTN error between the linear geometric mapping solution and the full nonlinear solution for orbits with high eccentricity. . . . .	33
38	Chief osculating and mean quasi-non-singular orbital elements vs. time. Without J2 effect. .	35
39	Deputy osculating and mean quasi-non-singular orbital elements vs. time. Without J2 effect. .	36
40	Chief osculating and mean quasi-non-singular orbital elements vs. time. With J2 effect. .	37
41	Deputy osculating and mean quasi-non-singular orbital elements vs. time. With J2 effect. .	38
42	Deputy osculating and mean quasi-non-singular relative orbital elements with respect to chief vs. time. Without J2 effect. . . . .	39
43	Deputy osculating and mean quasi-non-singular relative orbital elements with respect to chief vs. time. With J2 effect. . . . .	40
44	3D RTN plot of deputy's motion relative to chief. Without J2 . . . . .	41
45	Planar RTN plots of deputy's motion relative to chief. Without J2 . . . . .	42

46	3D RTN plot of deputy's motion relative to chief. With J2 . . . . .	43
47	Planar RTN plots of deputy's motion relative to chief. With J2 . . . . .	44
48	Osculating and mean relative quasi-non-singular orbital elements, decomposed into eccentricity vector, inclination vector, and $\delta a - \delta\lambda$ plots. Without J2 . . . . .	45
49	Osculating and mean relative quasi-non-singular orbital elements, decomposed into eccentricity vector, inclination vector, and $\delta a - \delta\lambda$ plots. With J2 . . . . .	46
50	RTN planar plots with $\delta i_y = 0$ . With J2. . . . .	48
51	OE plots with $\delta i_y = 0$ . With J2. . . . .	49
52	Orbital elements calculated using the STM with initial conditions from part 2 and part 6 ( $i_x = 0$ ). . . . .	50
53	A diagram of the two orbital modes. . . . .	51
54	A diagram of the relative orbital elements of the deputy spacecraft as we maneuver from in-train mode to passive safety ellipse mode. . . . .	54
55	A diagram of the relative orbital elements of the deputy spacecraft as we maneuver from in-train mode to passive safety ellipse mode, and then perform formation keeping to keep $\delta e_x \approx 0 \pm \delta e_{x,max}$ over the course of 200 orbits. . . . .	56
56	A diagram of our position in RTN over 200 orbits. There is initially a large change in the along-track, but this is solely due to the reconfiguration maneuver. During the formation keeping maneuvers, the along-track drift is kept in check, within a dead band. . . . .	57
57	Our cumulative delta v for both the initial maneuver from in-train to a passive safety ellipse and subsequent station keeping maneuvers over the course of 200 orbits. . . . .	60
58	Relative eccentricity vector tracking over time. We can see that our relative eccentricity vector begins to drift due to J2 effects and is reset by our impulsive maneuvers. . . . .	61
59	Relative inclination vector tracking over time. It remains largely unaffected by the J2 perturbation. . . . .	62
60	Relative semi-major axis and relative mean longitude tracking over time. We see a poor maneuver cause a non-zero $\delta a$ causing a drift in $\delta\lambda$ . However, this gets fixed in the following formation keeping maneuver. . . . .	63
61	A diagram of the relative orbital elements of the deputy spacecraft as we maneuver from in-train mode to passive safety ellipse mode and then maintain the safety ellipse mode. . . . .	68
62	A diagram of the cumulative delta-v applied over time (measured in number of orbits). Note that the bottom subplot is a zoomed-in plot for the first 3 orbits. We can see that the largest delta-v is applied in the first couple orbits to move the deputy into the passive safety ellipse mode and then a consistent amount of delta-v is required per orbit to oppose the J2 effects on the relative eccentricity vector. . . . .	69

63	A diagram of the tangential and normal components of delta-v over time. Note that the top subplot is zoomed in on the y-axis in order to show the small bursts of delta-v for formation keeping, while the bottom subplot is zoomed in on the x-axis to show the large bursts of delta-v in the first few orbits. We can see that the majority of the delta-v is applied in the first couple of orbits (both in the tangential and radially components), then a pattern emerges of very small bursts to combat error buildup due to perturbations. Notice that the high frequency pattern of normal burns has a lower frequency than that of tangential burns. . . . .	70
64	A diagram of the RTN position of the deputy with respect to the chief over time. We see a large along-track separation at the beginning, which is for the reconfiguration maneuver, but the controller very quickly reduces this and the deputy ends up in the desired cyclical motion of the passive safety ellipse. . . . .	71
65	A diagram of the $\Delta$ ROEs over time. The initial maneuver to go from in-train to passive safety ellipse mode takes only a couple of orbits. Then, after that all errors are close to zero for the formation keeping portion of the controller. Interestingly, $\Delta\delta\lambda$ appears to maintain at a small constant. . . . .	72
66	We can see that the estimated state is reasonable stable and close to the true state. . . . .	79
67	It is not too surprising that we are seeing a pre-fit with residual that is an order of magnitude greater than the post-fit residual because the measurement data is reasonable noisy when it comes in. . . . .	80
68	We can see here that the post fit residual of both position and velocity is less than the 95 percentile upper-bound of noise injected during the measurement model step. This reflects well on our model. . . . .	81
69	The state error is generally less than the 95 percentile upper confidence bound as one would expect. Further, the error terms are cyclical with the 95 percentile upper confidence bound. . . . .	82
70	Baseline performance. Plot showing scaled ROEs vs number of orbits passed. This is without closing the loop between navigation and control, and without injecting noise into the controller inputs. . . . .	85
71	Baseline performance over longer time. Plot showing scaled ROEs vs number of orbits passed. This is without closing the loop between navigation and control, and without injecting noise into the controller inputs. . . . .	86
72	Performance with noisy controller. Plot showing scaled ROEs vs number of orbits passed. The top plot is zoomed in over the first few orbits, and the bottom plot shows up to 15 orbits. . . . .	87
73	Closed loop performance. Plot shows scaled ROEs vs number of orbits passed. In this case, the control law is computed using the UKF's state estimate. . . . .	88
74	Closed loop performance. Plot shows scaled ROEs vs number of orbits passed. With updated controller parameters, and decreased measurement noise. . . . .	90
75	Closed loop performance. Plot shows scaled ROEs vs number of orbits passed. With measurement noise covariance decreased by an order of magnitude. . . . .	91
76	True and estimated ROEs over 10 orbits, with 95% confidence interval. . . . .	92

77	Pre-fit and post-fit residuals. . . . .	93
78	Post-fit residuals with 95% line from measurement noise covariance matrix. . . . .	94
79	UKF state estimate errors. . . . .	95
80	Control input history in each direction, magnitude, and cumulative. . . . .	96
81	Control tracking error. . . . .	97
82	ROEs vs time showing simulation performance, with noisy actuators. . . . .	98

## List of Tables

1	Summary of project revisions. . . . .	1
---	---------------------------------------	---

# 1 Problem Set 1

## 1.1 Problem 1: Your Mission, Your Challenge

### 1.1.1 Survey of current DSS's

We considered the following distributed space systems as possible focuses for our project. In addition to the systems listed below, we considered PROBA-3, mDOT, OneWeb, etc. but dismissed them due to lack of interest or too high similarity to missions listed below.

- **Starling**

- This is a NASA mission set to launch this coming summer. It consists of 4 6U cubesats in sun-synchronous LEO, and is designed as a technology demonstration for new spacecraft formation flying algorithms. The 4 spacecraft will generally fly in a (kind of) line, but may perform maneuvers throughout it's lifetime to reposition the formation. The tech demo will consist of 4 experiments, one of the being StarFOX, run by the Space Rendezvous Lab.

- **GRACE**

- The GRACE mission was launched in 2002 with the goal to accurately map variations in the Earth's gravity field. GRACE consists of two spacecraft in a polar orbit separated by 500 kilometers. The mission was designed for a five year life span. GRACE observed gravitation variations due to runoff and ground water, glacial flow, exchanges between ice sheet, and other natural phenomena.

- **GRAIL**

- The GRAIL mission was essentially the GRACE mission, but for the Moon. It consisted of two spacecraft in low, elliptical, polar lunar orbit to accurately measure the Moon's gravity field. The mission was launched in the latter half of 2011, and reached the Moon in a few months. It completed it's mission in about a year, before impacting the lunar surface.

- **SWARM-EX**

- SWARM-EX (Space Weather Atmospheric Reconfigurable Multiscale Experiment) consists of three 3U CubeSats that will measure ionized and neutral gases in Earth's upper atmosphere. Launch is targeted for 2024 and the CubeSats will orbit at an altitude between 300-600 km. The benefit of three satellites is the ability to record spatial variance in atmospheric measurements.

- **VISORS**

- VIrtual Super Optics Reconfigurable Swarm (VISORS) is a distributed telescope that consists of two 6U CubeSats with a target to launch in 2024. One spacecraft contains an optical payload while the other contains the detector. The goal of the two space craft is to take high resolution images of the solar corona. The two space craft will orbit in a near-circular sun synchronous low Earth Orbit. When the two space craft take a photo they need to maintain a millimeter level of accuracy.

- **Starlink**

- Starlink is a satellite internet constellation created by SpaceX with 3,500 satellites in LEO. Starlink satellites orbit at about 550 km and are in the the smallsat-class of 100-500kg. The goal of Starlink is to provide global internet coverage.

### 1.1.2 Our Chosen DSS

We chose the **Starling** mission as our framework for our course project. We chose this because we wanted our DSS to be geocentric and consist of more than 2 spacecraft. And because we liked its mission objectives.

All information below can be found on NASA's website here: [1].

**Mission Name and Operator:** The mission is called the Starling mission and is operated chiefly by NASA. Specifically, NASA Ames Research Center heads the project. Firefly Aerospace is operating the launch, along with NASA's Kennedy Space Center, while Nanoracks, LLC is in charge of launch integration. Blue Canyon Technologies is the main bus manufacturer and will support mission operations, and L3Harris will provide ground software support for navigation and control.

**Mission Objectives:**

The primary mission objective is to demonstrate the use of 4 advanced distributed space system technologies that could potentially allow spacecraft formation flying with none to minimal ground support. The 4 technologies are: “swarm maneuver planning and execution,” “communications networking,” “relative navigation,” and “autonomous coordination between spacecraft.”

The secondary mission objectives are to perform some Earth observation data collection in the form of atmospheric phenomenon detection, and to provide valuable experience to the parties involved for use in the design and operation of future distributed space system missions.

**Number and Type of Satellites:** 4 satellites, each (almost) identical 6U cubesats.

**Orbital Parameters, Tentative Launch Dates, and Mission Duration:**

*Note: Many papers and websites contain out of date information on these parameters (i.e. NASA's website still says it's supposed to launch in 2022); we've done our best to provide the most up to date information, but some of it may inevitably be wrong, ambiguous, or may be subject to change in the future.*

Absolute orbit: Sun-synchronous Earth orbit. The only information on altitude is “over 300 miles” (482 km).

Relative orbit: Tens of km to 200 km range between satellites, no more than 270km. In-train and passive safety ellipse formations. Passive safety ellipse should have relative motion such that relative orbits remain in field of view of observing satellites.

Tentative launch date: Summer 2023.

Tentative mission duration: Around 6 months to a year or two.

**Basic Description of Functioning/Scientific Principles:**

The StarFOX experiment uses star trackers for optical angles-only navigation. The lead and last satellites are observing satellites that look inwards at the satellites to track their motion and determine relative orbits. The ROMEO experiment use algorithms to autonomously perform manuevers. The MANET experiment will “two-way S-band crosslink radios/antennas” to perform swarm communication that is

robust to losing network nodes. Demonstrating autonomous monitoring of Earth's ionosphere will use dual band GPS receivers to measure the density of atmospheric phenomenon.

**Key DNG&C Requirements:** The demonstration of the technologies require that the satellites remain close enough to each other to be resolvable on a star tracker (StarFOX) and be able to communicate through S-band (MANET). The secondary Earth observation mission objective requires global coverage, which, with just 4 satellites, requires a near polar orbit, thus the choice of sun-synchronous orbit. Further, the leader/follower technique for relative navigation requires the “in-train” (or close to it) formation. Additionally, since Starling is testing new technologies, there is inherently a requirement for passive safety (as seen in the “passive safety ellipse” formation) so that if anything goes wrong, the satellites do not collide with each other.

**Classification of DSS According to Separation Between Elements and Required Navigation/Control Accuracy:** Swarm / Formation Flying. The four 6-unit CubeSats fly in two formations with tens of km to 270 km range between satellites. Requires relaxed to moderate levels of relative orbit accuracy.

## 1.2 Problem 2: Absolute Orbit Simulation

All code is linked to GitHub in PSet 9.

### 1.2.1 Answers

- (a) Exact initial conditions are unknown, but we do know it should be a sun-synchronous orbit launched in July 2023. We will assume the following initial orbital elements:

$$\begin{bmatrix} a \\ e \\ i \\ \Omega \\ \omega \\ \nu \end{bmatrix} = \begin{bmatrix} 7000\text{km} \\ 0.001 \\ 98^\circ \\ 0 \\ 90^\circ \\ 0 \end{bmatrix}$$

And, we will choose orbital insertion at noon GMT on July 4th 2023.

- (b) Our inertial frame will be ECI. The position and velocity vectors are as follows:

$$r_{ECI} = \begin{bmatrix} 0 \\ -973.237495013738 \\ 6924.9446047098 \end{bmatrix} \text{km}$$

$$v_{ECI} = \begin{bmatrix} -7.55359893407486 \\ 0 \\ 0 \end{bmatrix} \text{km/s}$$

- .
- (c) Plot:

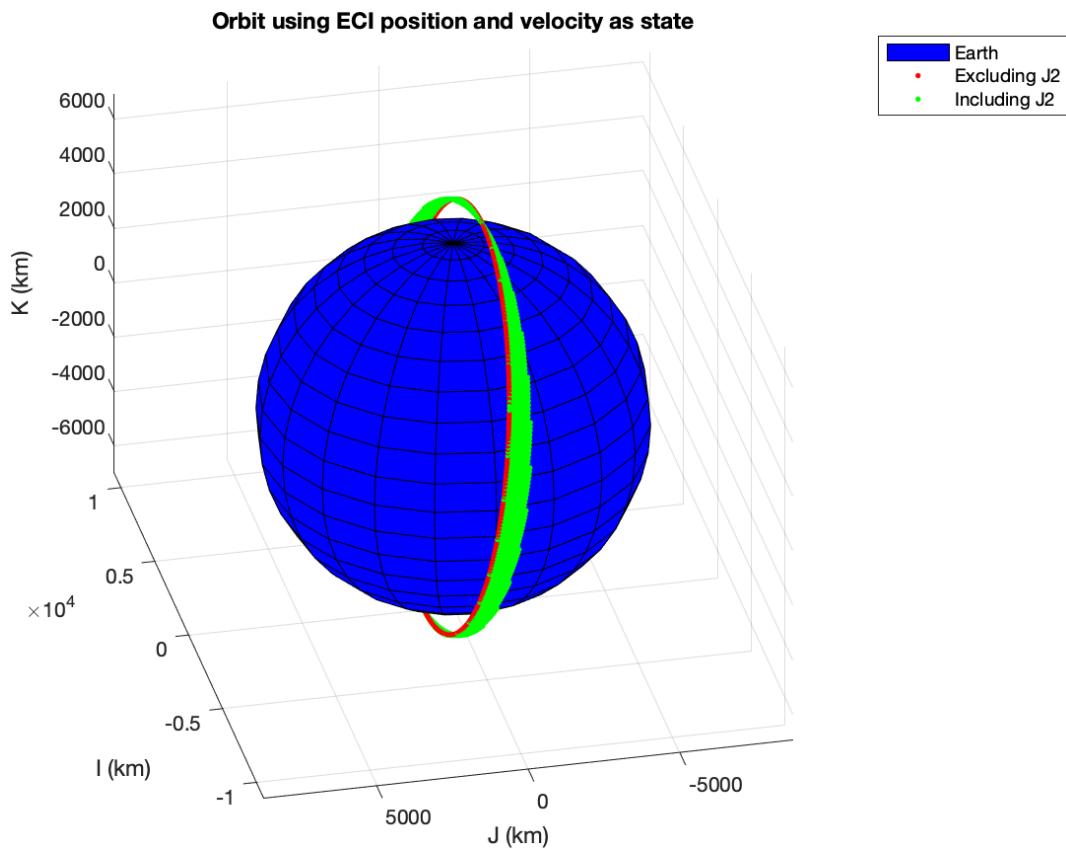


Figure 1: Orbit simulation performed using ECI position and velocity

(d) Plots:

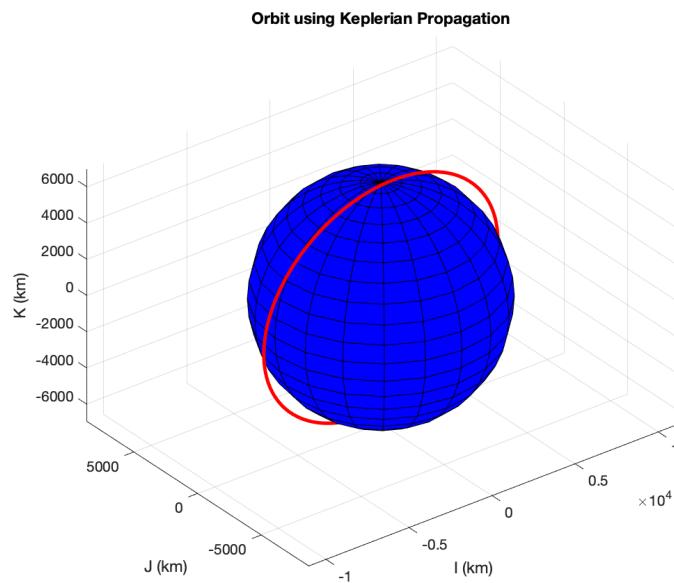


Figure 2: Orbit simulation performed using analytical Keplerian propagation, excluding J2 effects.

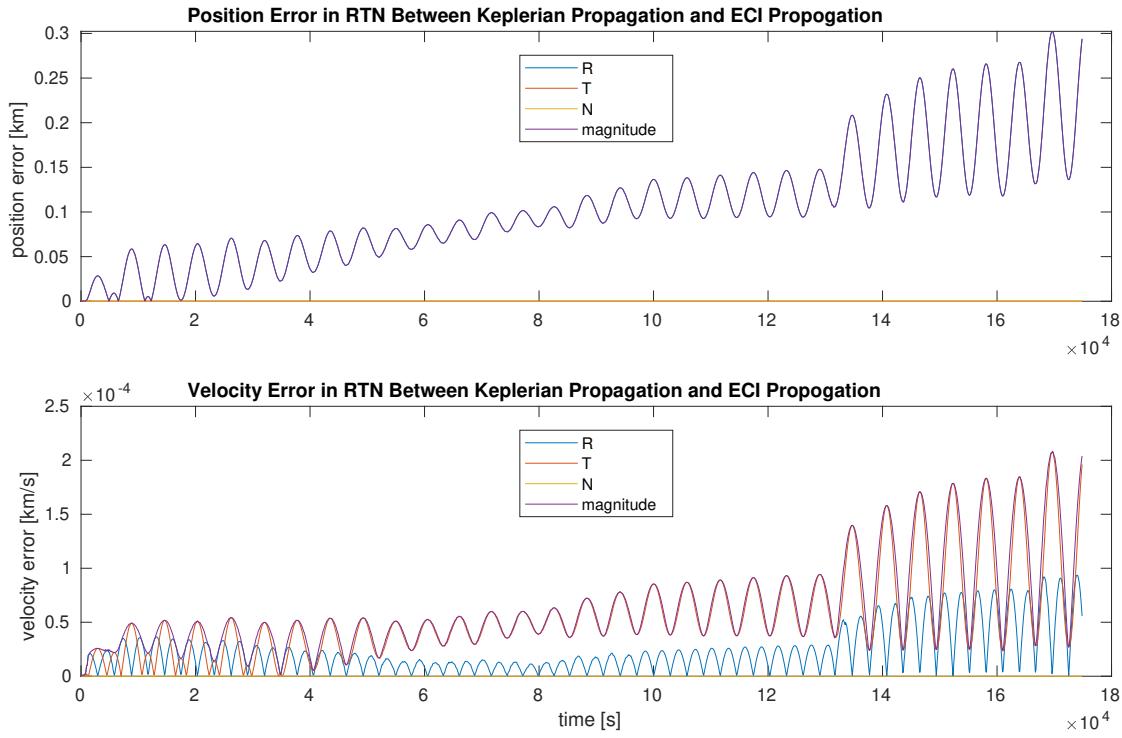


Figure 3: Errors (in RTN) between ECI simulation and analytical Keplerian propagation, excluding J2 effects.

- (e) Excluding J2 effects, we see that the eccentricity vector, angular momentum vector, and specific mechanical energy, as well as  $a$ ,  $e$ ,  $i$ ,  $\Omega$ , and  $\omega$  are all constant (within numerical precision, of course). This makes sense, since without perturbations, all these quantities should be constant. And, we see the true anomaly  $\nu$  change as expected, throughout the orbit. The RAAN appears to be making large steps in the graph, but we realized this was likely a function of small oscillations between 0 and  $1.99\pi$ .

Including J2 effects, we see that both our eccentricity vector and angular momentum vector change (process) due to the effects of J2. However, we see that the specific mechanical energy stays roughly constant with the exception of a small cyclic variation. We do see reasonably large variation in our  $a$  orbital element (20 km) which we think is a function of numerical errors. Further,  $e$ , and  $i$  all stay constant, as expected because J2 does not affect them, but we do see  $\Omega$  and  $\omega$  change, since J2 does affect these. And, of course, we still see  $\nu$  change, as expected.

Plots:

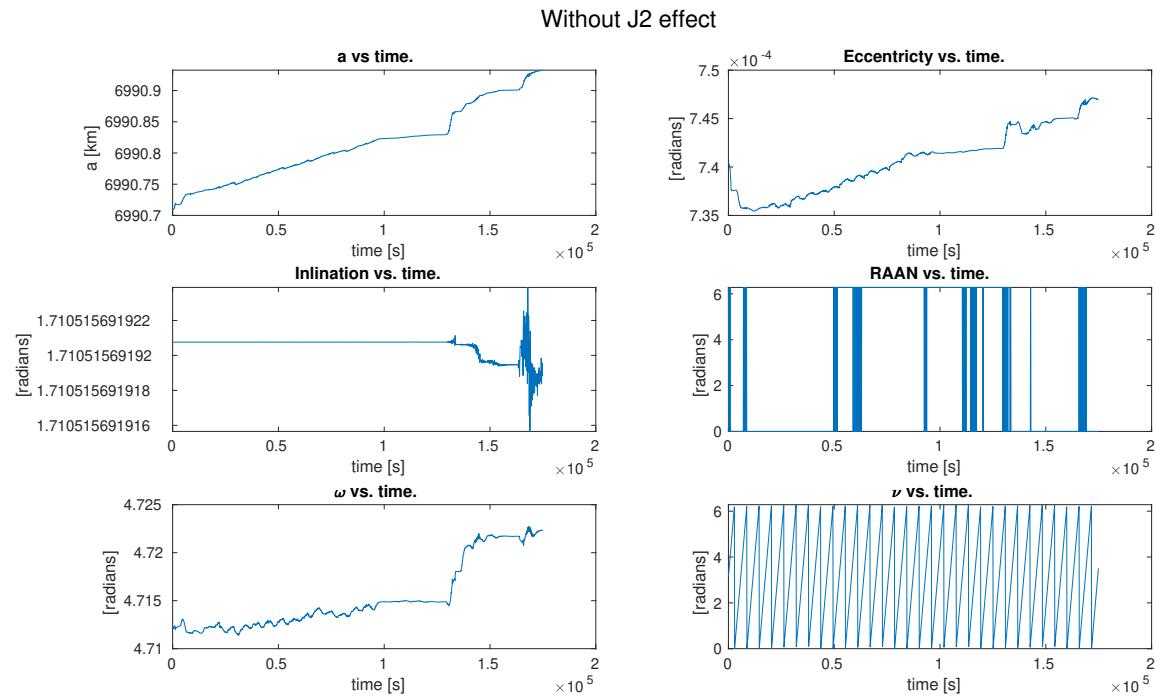


Figure 4: Orbital elements over time without the J2 effect.

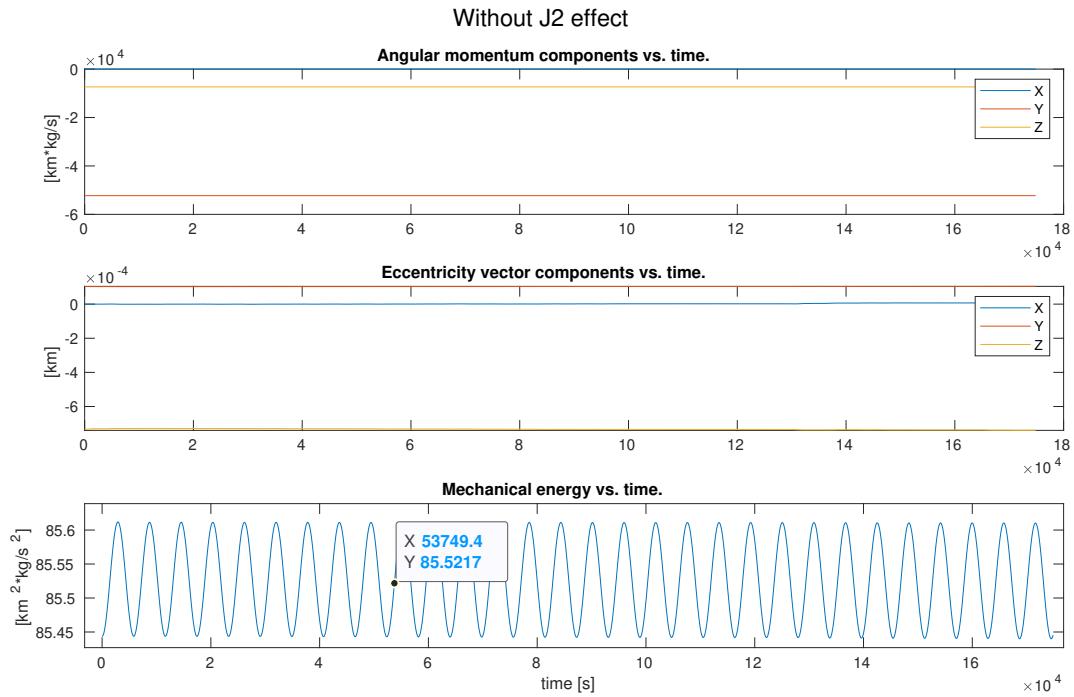


Figure 5: Angular momentum, eccentricity vector, and mechanical energy over time without the J2 effect.

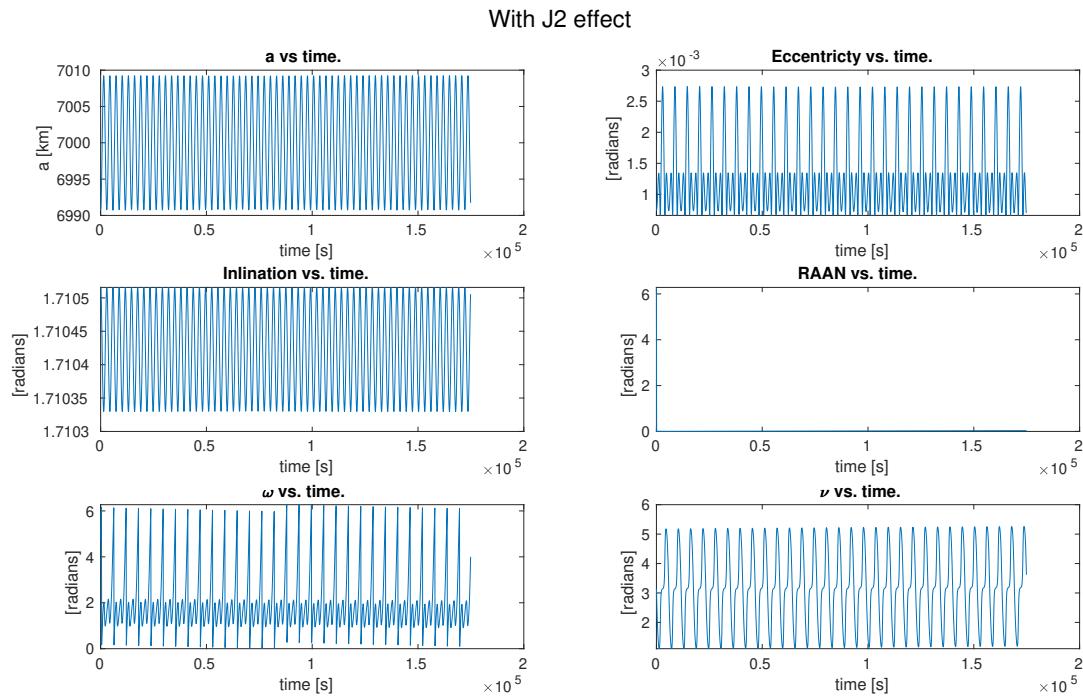


Figure 6: Orbital elements over time with the J2 effect.

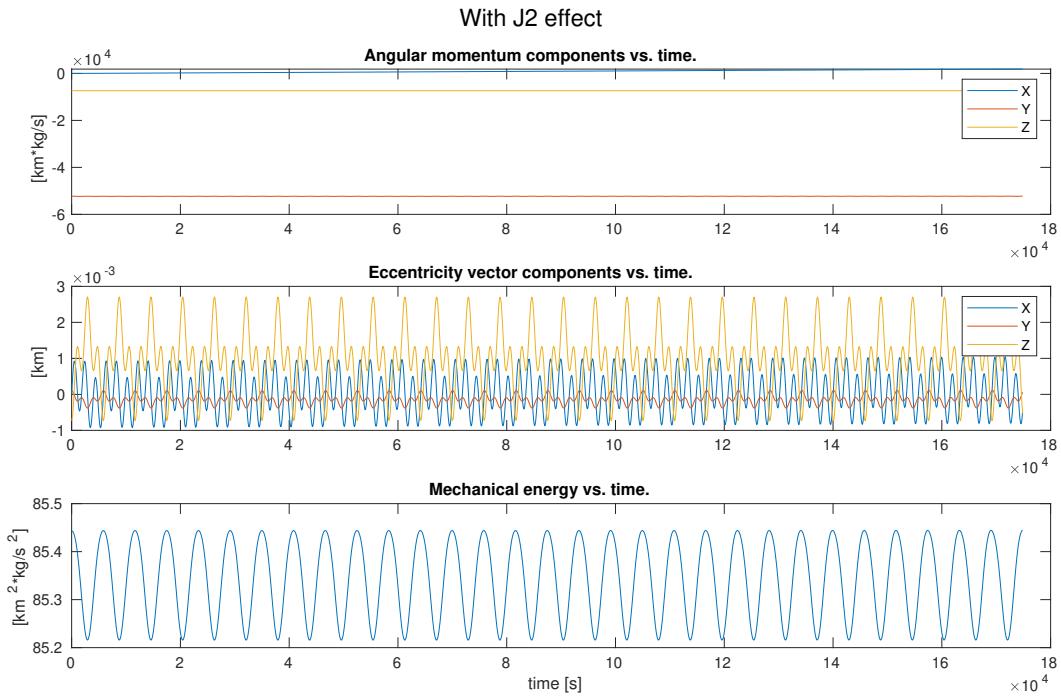


Figure 7: Angular momentum, eccentricity vector, and mechanical energy over time with the J2 effect.

(f) We definitely see some errors between our simulations here, but everything seems within reason, so we conclude that our simulation show general consistency with each other. The biggest errors we see are in  $a$  and  $\omega$ , but we chalk both up to numerical precision errors and/or osculating vs mean orbital elements errors.

Orbit:

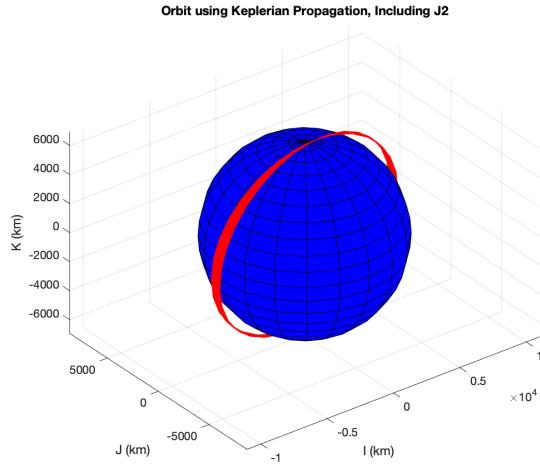


Figure 8: Orbit using Keplerian Simulation, Including Mean J2 Effects

Superimposed orbital elements:

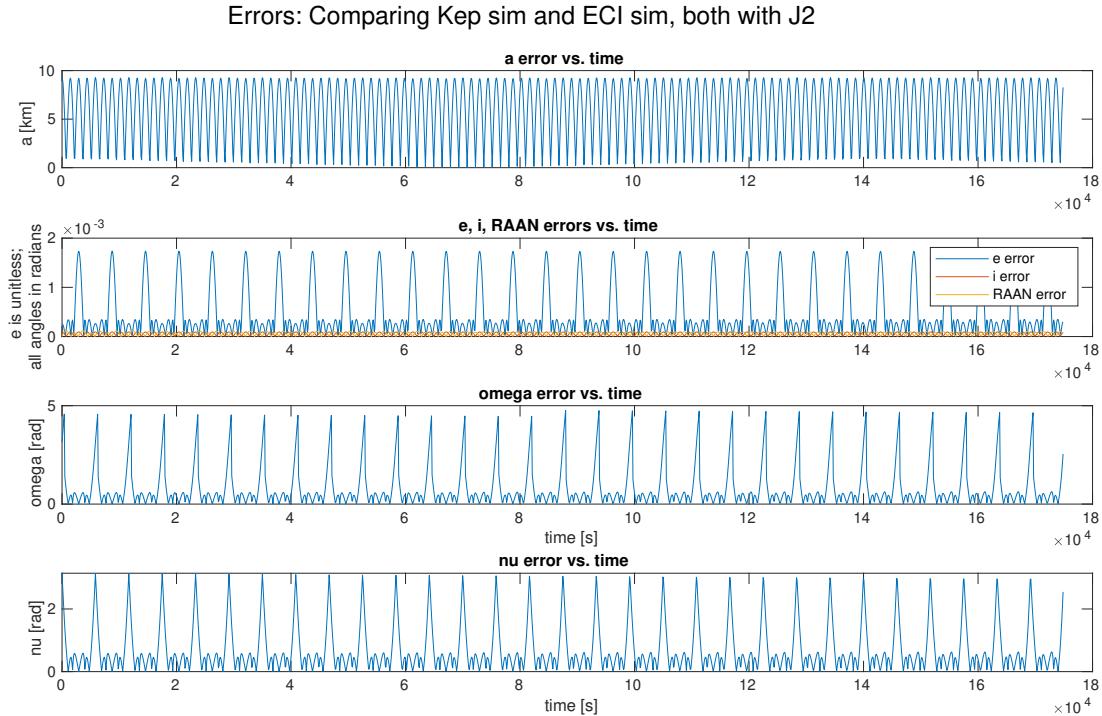


Figure 9: Orbital Elements using Keplerian Simulation and ECI Simulation, Including Mean J2 Effects

Orbital element errors:

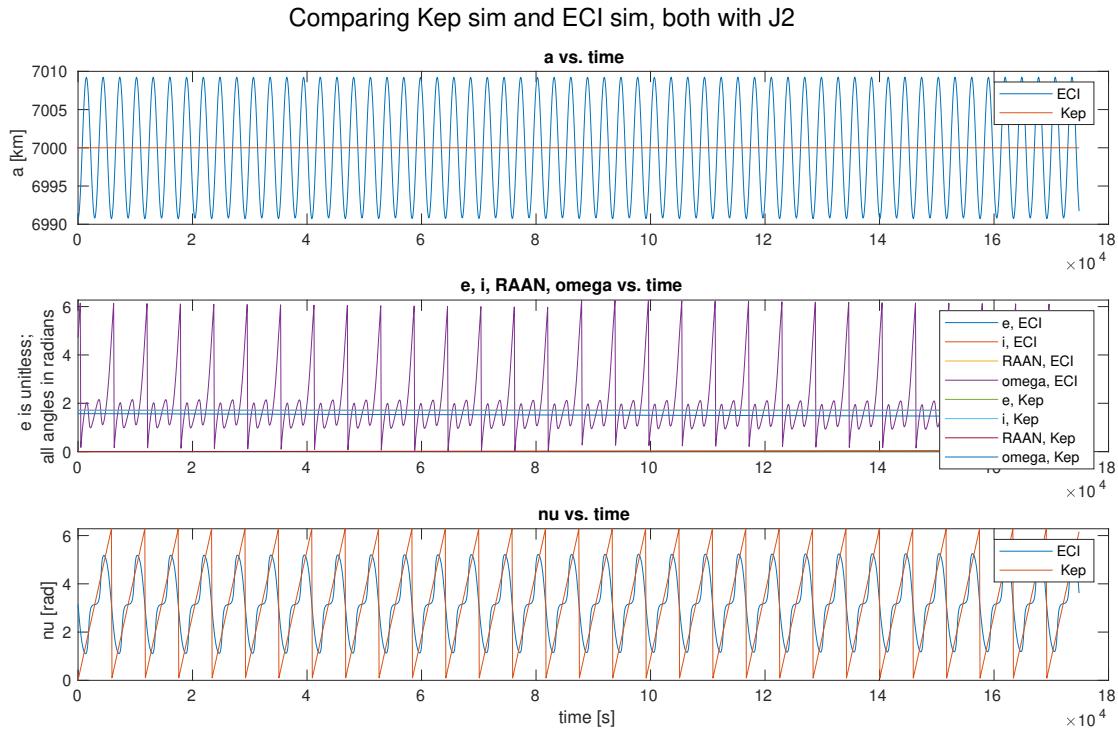


Figure 10: Orbital Element Errors between Keplerian Simulation and ECI Simulation, Including Mean J2 Effects

- (g) If we cared about mean orbital elements, we could simply time average the osculating orbital elements we get from part (c) over multiple orbits to get mean orbital elements. To go from mean orbital elements to osculating elements, we could use a Brouwer transformation which accepts mean elements as inputs and outputs osculating elements.

## 2 Problem Set 2

### 2.1 Problem 1: Everything is Relative

Note: For the sake of simplicity, we will do this problem with just 1 deputy, however the whole problem can simply be repeated for the rest of the deputies with different initial relative orbits.

Further note that all code is linked to GitHub in PSet 9.

#### 2.1.1 Answers

- (a) Recall our chief's absolute orbit:

$$\begin{bmatrix} a \\ e \\ i \\ \Omega \\ \omega \\ \nu \end{bmatrix} = \begin{bmatrix} 7000\text{km} \\ 0.001 \\ 98^\circ \\ 0 \\ 90^\circ \\ 0 \end{bmatrix}.$$

Our deputy's orbits will have the same semi-major axis, but will have some of their other elements slightly perturbed. Our only real constraint is that the separation distance between the chief and deputy cannot exceed about 270 km. Let's choose an along-track separation of around 60 km, which gives roughly about a 0.5 degrees of change in the true anomaly. Then, we will tweak the other elements here and there, to give some nice relative motion, keeping the tweaks small so that we don't go over 200 km total separation (we did end up going over the 200 km "max," but it shouldn't matter for this pset, and it helps us differentiate numerical errors).

$$\begin{bmatrix} a \\ e \\ i \\ \Omega \\ \omega \\ \nu \end{bmatrix} = \begin{bmatrix} 7000\text{km} \\ 0.01 \\ 98.5^\circ \\ 0.5^\circ \\ 90.5^\circ \\ 0.5^\circ \end{bmatrix}.$$

- (b) Plots:

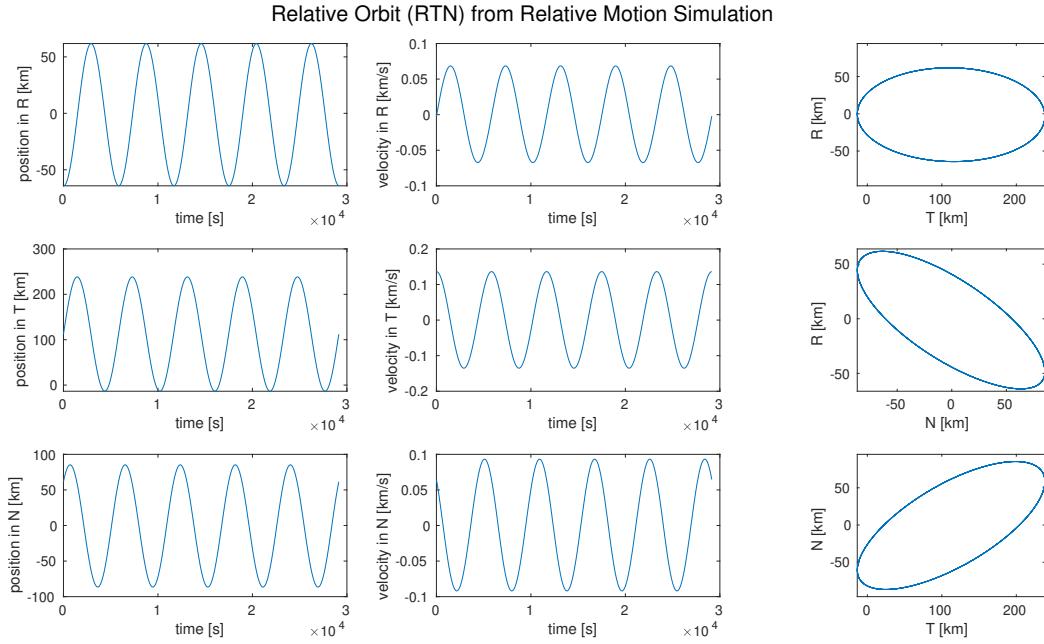


Figure 11: Relative motion generated from our numerical relative motion simulator, ignoring perturbations.

(c) Plots:

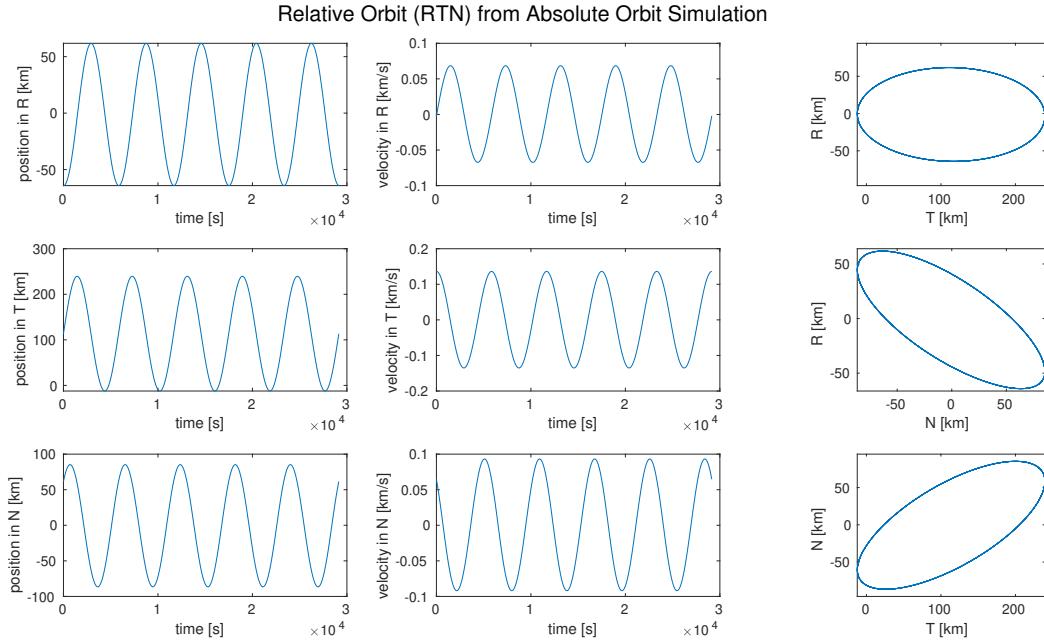


Figure 12: Relative orbit from fundamental orbital differential equations of absolute motion.

(d) Plots: Error plots when chief and deputy have the same semi-major axis:

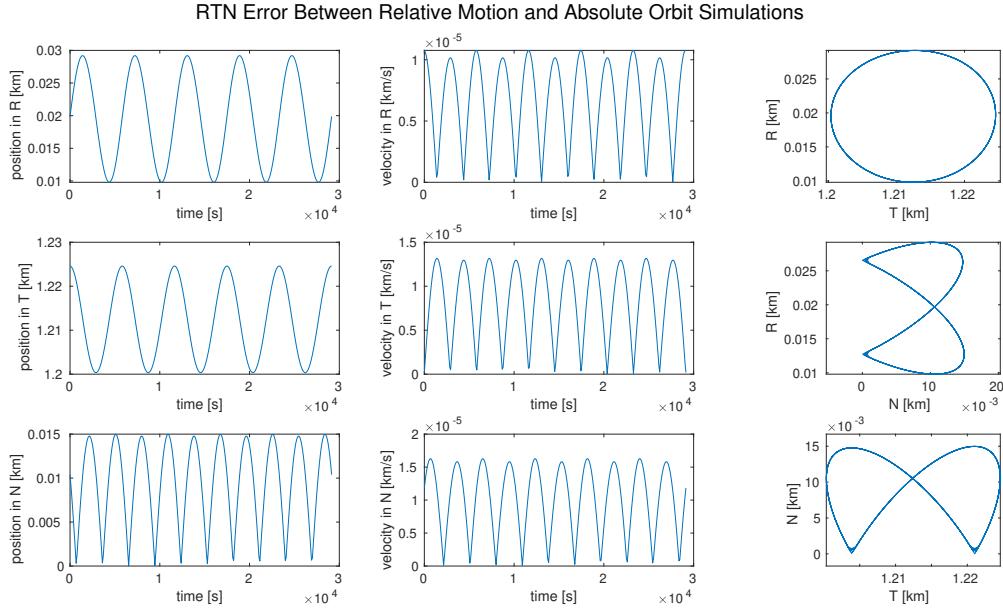


Figure 13: Error between relative motion numerical simulation and orbit from fundamental differential equations.

Error plots when chief and deputy have different semi-major axes:

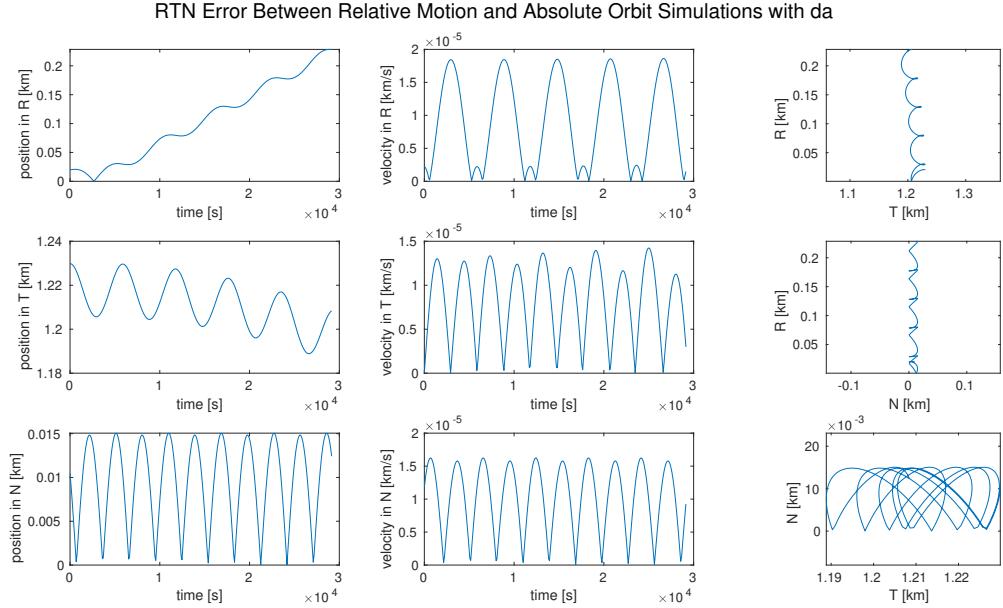


Figure 14: Error between the relative motion numerical simulation and the fundamental differential equations for absolute motion simulation when the deputy and the chief have a different semi-major axis.

- (e) Plots with a chief and deputy with a difference in semi-major axis:

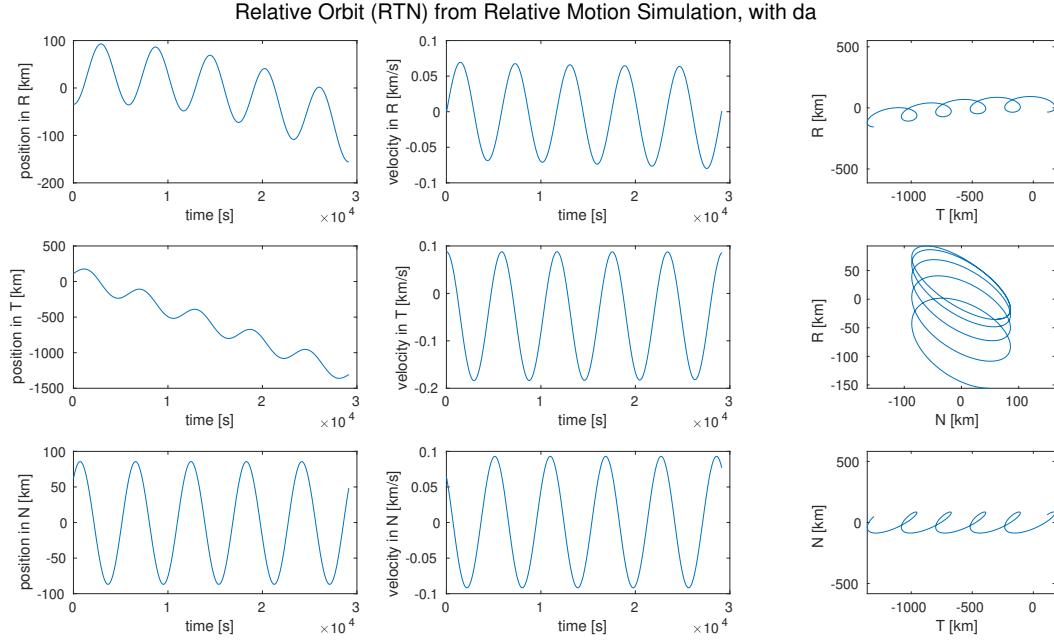


Figure 15: Relative orbit in the RTN frame after introducing a change in semi-major axis.

- (f) In order to return to bounded periodic relative motion, we need to perform a maneuver of the deputy that eliminates the difference in semi-major axis of the chief and deputy. Looking at the GVEs, we know the most efficient way to do this is to perform an along-track burn at periapsis. This will affect the eccentricity of the deputy's orbit, but if the goal is to choose the most efficient maneuver to achieve bounded periodic relative motion, without constraints on said motion, then we do not care about the effects of this maneuver on other orbital elements. We perform a burn in the along-track direction following:

$$\Delta v_T = \frac{\Delta anr}{2a\sqrt{1-e^2}},$$

where  $n$  is the mean motion and  $r$  is the radius of the deputy.

- (g) We simulated about 2 orbits, then once the deputy returned to periapsis, we calculated and applied the delta-v required to eliminate the difference in semi-major axis, then continued to simulate the rest of the orbits.

Plots:

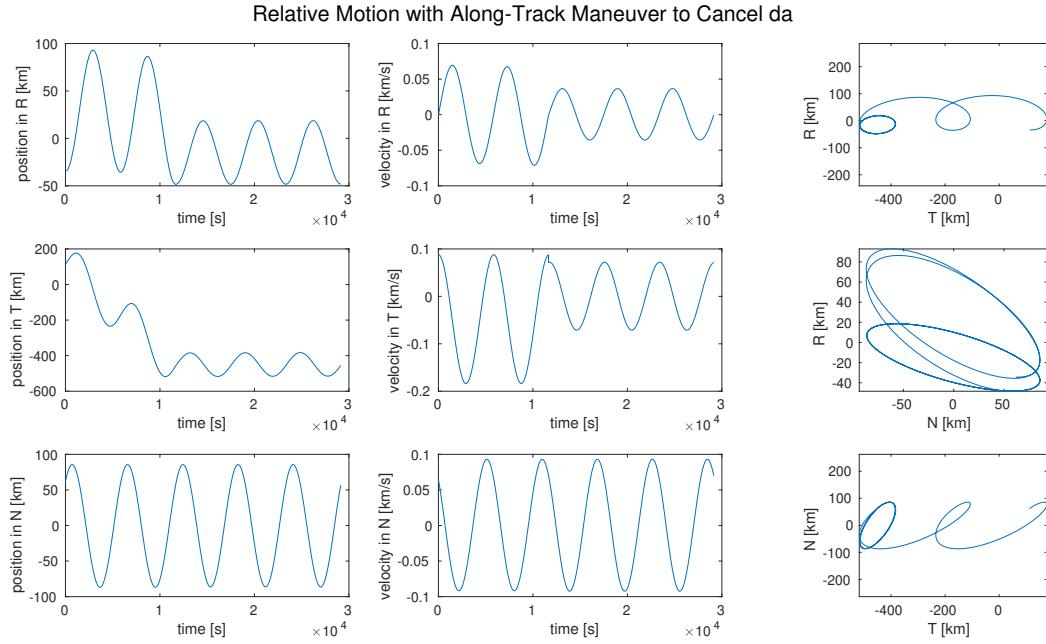


Figure 16: Relative position and velocity before and after applying an along track maneuver to cancel the relative difference in semi-major axis.

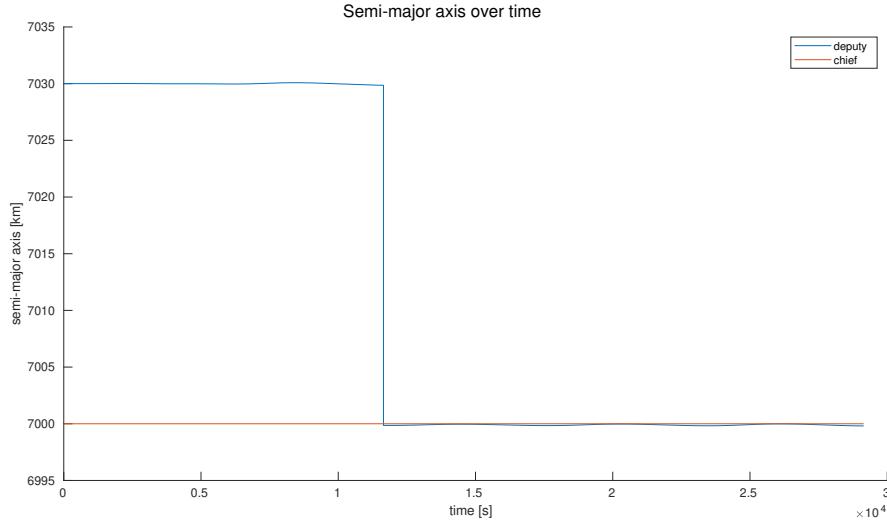


Figure 17: Semi-major axis of chief and deuputy over time. We see that the semi-major axis of the deputy decreases at the point of our along track burn.

Obviously, there is still a bit of difference in the semi-major axis after our burn, which is likely due to numerical error that has accumulated in the two orbits before our burn.

### 3 Problem Set 3: Linearized relative orbit in Cartesian coordinates.

#### 3.1 Problem 1. We are Close in Near-Circular Orbits.

##### 3.1.1 Answers

- (a) Recall our chief's absolute orbit:

$$\begin{bmatrix} a \\ e \\ i \\ \Omega \\ \omega \\ \nu \end{bmatrix} = \begin{bmatrix} 7000 \text{ km} \\ 0.001 \\ 98^\circ \\ 0 \\ 90^\circ \\ 0 \end{bmatrix}.$$

Our deputy orbit will be slightly closer to the chief orbit than in the previous problem set in order to satisfy  $\rho/r_0 \approx 0.001$ . Thus our deputy orbit for this problem is:

$$\begin{bmatrix} a \\ e \\ i \\ \Omega \\ \omega \\ \nu \end{bmatrix} = \begin{bmatrix} 7000 \text{ km} \\ 0.001 \\ 98.01^\circ \\ 0.05 \\ 90.05^\circ \\ 0.01 \end{bmatrix}.$$

- (b) Inertial position and velocity of chief:

$$r_{ECI} = \begin{bmatrix} 0 \\ -973.2 \\ 6924.9 \end{bmatrix} \text{ km.}$$

$$v_{ECI} = \begin{bmatrix} -7.55 \\ 0 \\ 0 \end{bmatrix} \text{ km/s.}$$

Relative position and velocity of deputy in RTN frame:

$$r_{RTN} = \begin{bmatrix} -0.0031 \\ 6.4727 \\ 1.2265 \end{bmatrix} \text{ km.}$$

$$v_{RTN} = \begin{bmatrix} 1.749e-07 \\ 6.6901e-06 \\ 0.0065 \end{bmatrix} \text{ km/s}$$

ROEs:

$$\begin{bmatrix} \delta a \\ \delta \lambda \\ \delta e \\ \delta i \\ \delta \Omega \\ \delta \omega \\ \delta \nu \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0.0001745329 \\ 0.0008726646 \\ 0.0008726646 \\ 0.0001745329 \end{bmatrix},$$

$$\begin{bmatrix} \delta a \\ \delta \lambda \\ \delta e_x \\ \delta e_y \\ \delta i_x \\ \delta i_y \end{bmatrix} = \begin{bmatrix} 0 \\ 9.25e-04 \\ -8.7266e-07 \\ -3.808e-10 \\ 1.7453e-04 \\ 8.64e-04 \end{bmatrix}.$$

(c) Integration coefficients.

$$\begin{bmatrix} K_1 \\ K_2 \\ K_3 \\ K_4 \\ K_5 \\ K_6 \end{bmatrix} = \begin{bmatrix} -2.48959e-06 \\ 2.3182e-08 \\ 2.0463e-06 \\ 0.000924623 \\ 0.000864853 \\ 0.0001752099 \end{bmatrix}.$$

(d) Propagation using standard solution of HCW.

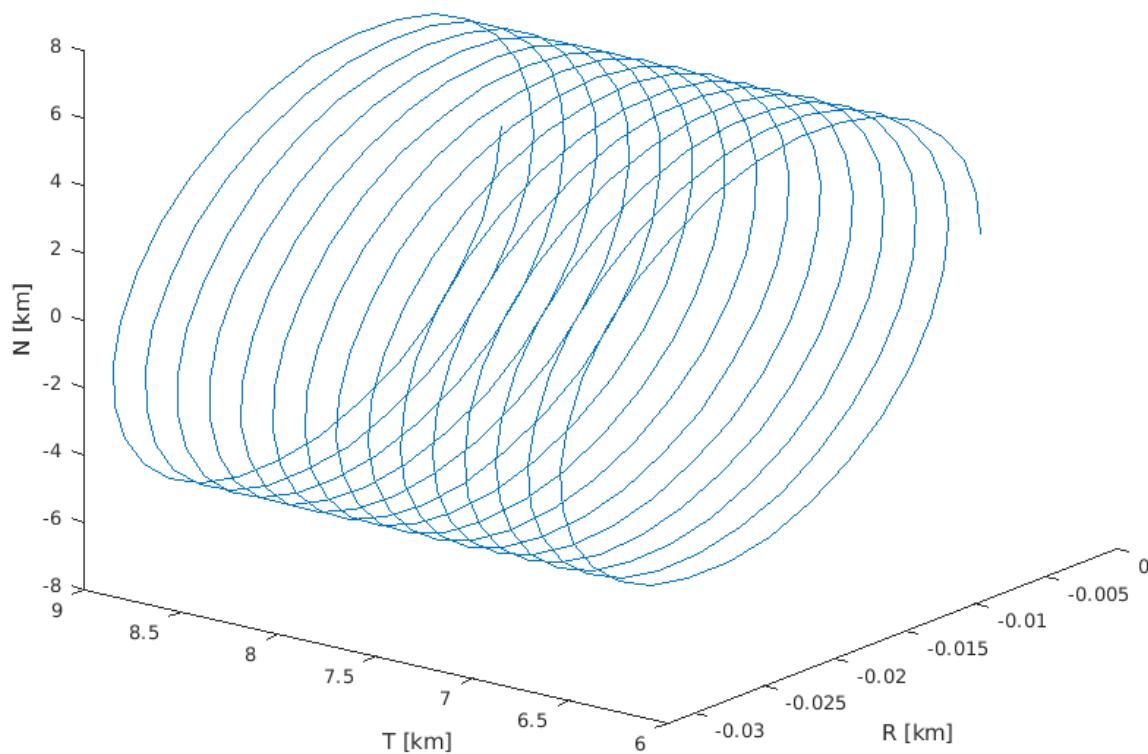
**Relative position in RTN over time.**

Figure 18: Relative position in the RTN frame. We can see that the propagation using the standard HCW solution results in substantial drift along the T axis.

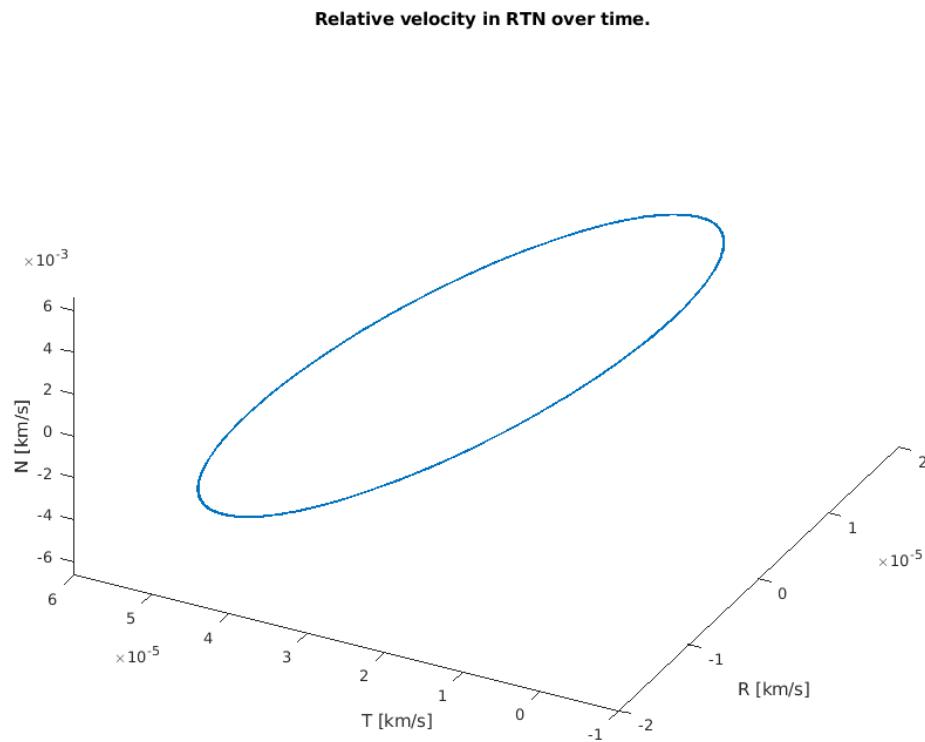


Figure 19: The relative velocity is bounded.

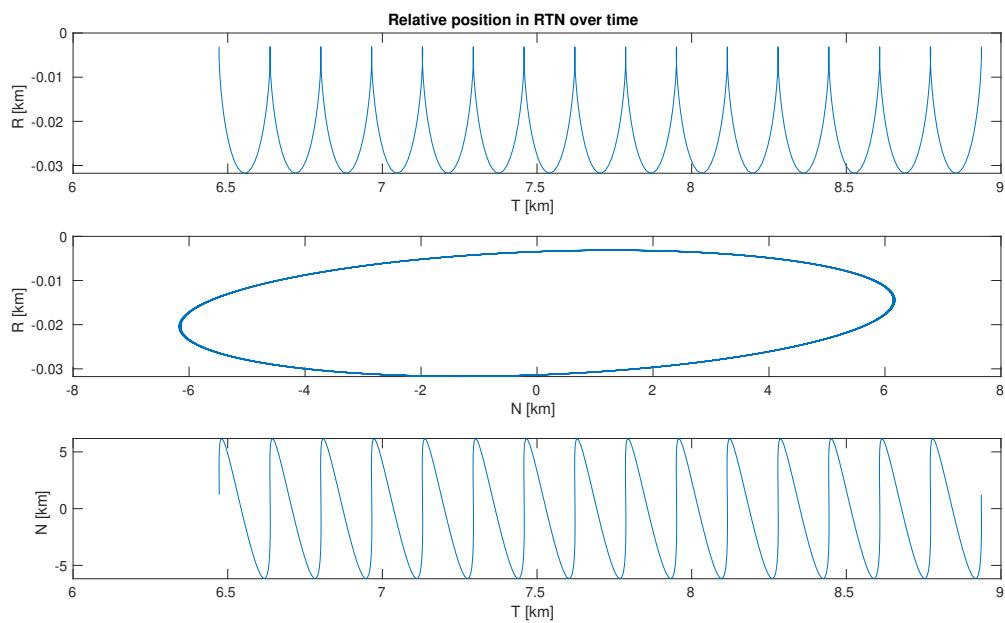


Figure 20: We can also observe the drift in the  $T$  direction on the 2D graphs.

(e) Discuss general behavior.

We would expect to see bounded relative motion as our initial conditions have the energy matching condition of  $\delta a = 0$ . However, the HCW is a linear system. In order to stabilize along track motion it is necessary to impose the constraint that  $\dot{y}(0) = -2nx(0)$ . In the above plots, we did not impose that constraint and as a result saw substantial along track drift.

If we impose that constraint, we indeed produce a results with bounded relative motion.

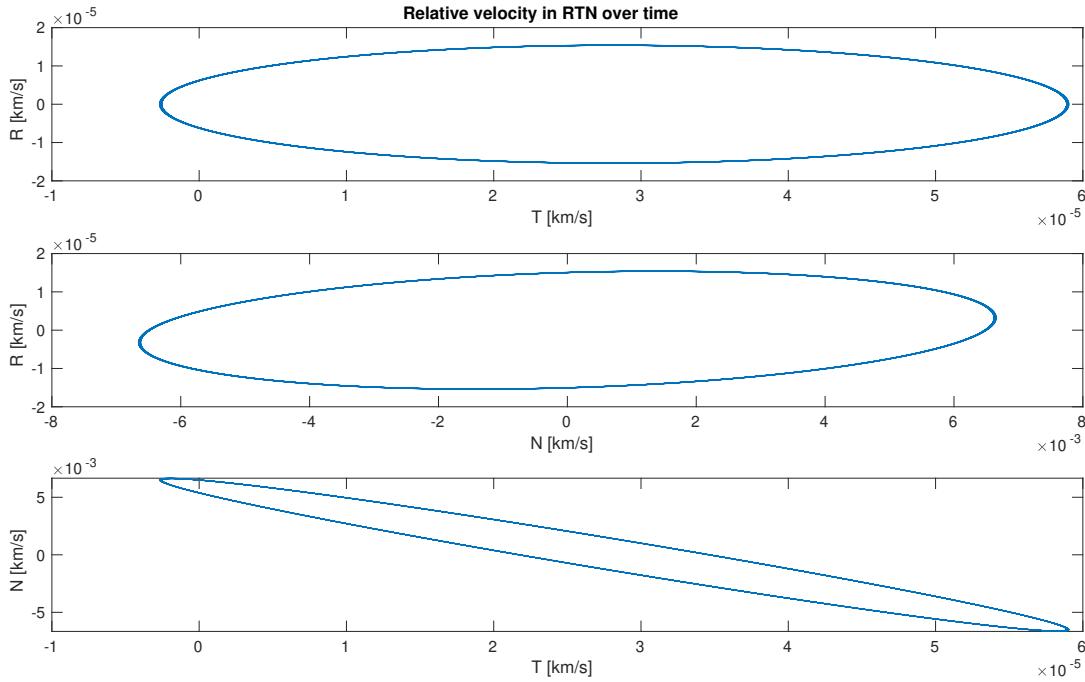


Figure 21: Once we impose the additional constraint of  $\dot{y}(0) = -2nx(0)$  we produce bounded relative motion.

### 3.2 Problem 2. We are Close in Eccentric Orbits.

#### 3.2.1 Answers

- (a) We have kept our initial conditions as similar as possible, while keeping them realistic and increasing the eccentricity. We chose a chief eccentricity of 0.2, and in order to ensure our perigee was not inside the Earth, we increased our semi-major axis to 9000 km. And for our deputy, we increased the eccentricity to very close to the chief's and we matched the chief's semi-major axis. Thus, we use the following initial conditions.

For the chief:

$$\begin{bmatrix} a \\ e \\ i \\ \Omega \\ \omega \\ \nu \end{bmatrix} = \begin{bmatrix} 9000 \text{ km} \\ 0.2 \\ 98^\circ \\ 0 \\ 90^\circ \\ 0 \end{bmatrix}.$$

Our deputy orbit is:

$$\begin{bmatrix} a \\ e \\ i \\ \Omega \\ \omega \\ \nu \end{bmatrix} = \begin{bmatrix} 9000 \text{ km} \\ 0.1999 \\ 98.01^\circ \\ 0.05 \\ 90.05^\circ \\ 0.01 \end{bmatrix}.$$

(b) The YA solution integration constants are:

$$\begin{bmatrix} K_1 \\ K_2 \\ K_3 \\ K_4 \\ K_5 \\ K_6 \end{bmatrix} = \begin{bmatrix} -0.00000341222 \\ 0.00002427882 \\ 0.00010664247 \\ 0.00087229707 \\ 0.00086390372 \\ 0.00017540717 \end{bmatrix}.$$

(c) Relative motion propagated using YA solution:

3D plots:

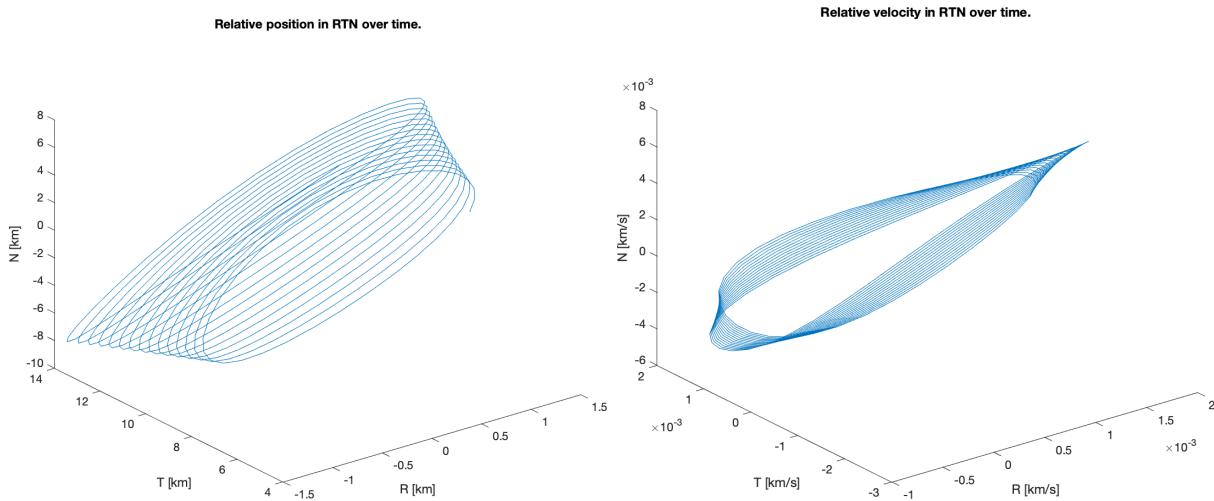


Figure 22: RTN relative position and velocity in 3D space, using the YA solution.

Planar plots:

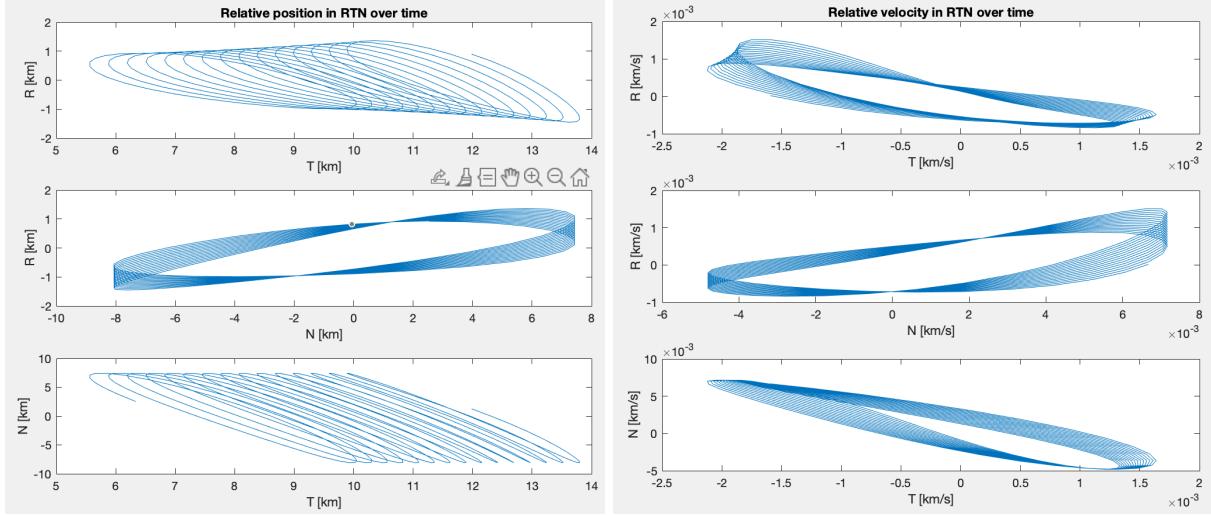


Figure 23: RTN relative position and velocity on each RTN plane, using the YA solution.

- (d) Yes, the trends appear consistent with what we'd expect given our initial conditions and integration constants. There is drift, but not too much, since our  $\rho/r_0$  is very small.

No, the relative motion is not bounded, because we are propagating using linearized equations which are merely approximations of the full dynamics. We could fix this by imposing constraints on our equations of propagation, similar to what we did with the HCW equations. We could set our initial orbital elements such that  $c_3 = 0$  from the TH equations. But that seems a bit of a backwards solution. Instead, we could apply corrections, such as to true anomaly or amplitude of YA solutions, throughout the propagation in order to keep the relative motion bounded.

- (e) Quasi non-singular relative orbital elements:

$$\begin{bmatrix} \delta a \\ \delta \lambda \\ \delta e_x \\ \delta e_y \\ \delta i_x \\ \delta i_y \end{bmatrix} = \begin{bmatrix} 0 \\ 0.0008652437 \\ -0.0001744456 \\ -0.0001000761 \\ 0.0001745329 \\ 0.0008641719 \end{bmatrix}.$$

- (f) Propagation using linear geometric mapping solution (plotted over plots from part (c)):

3D plots:

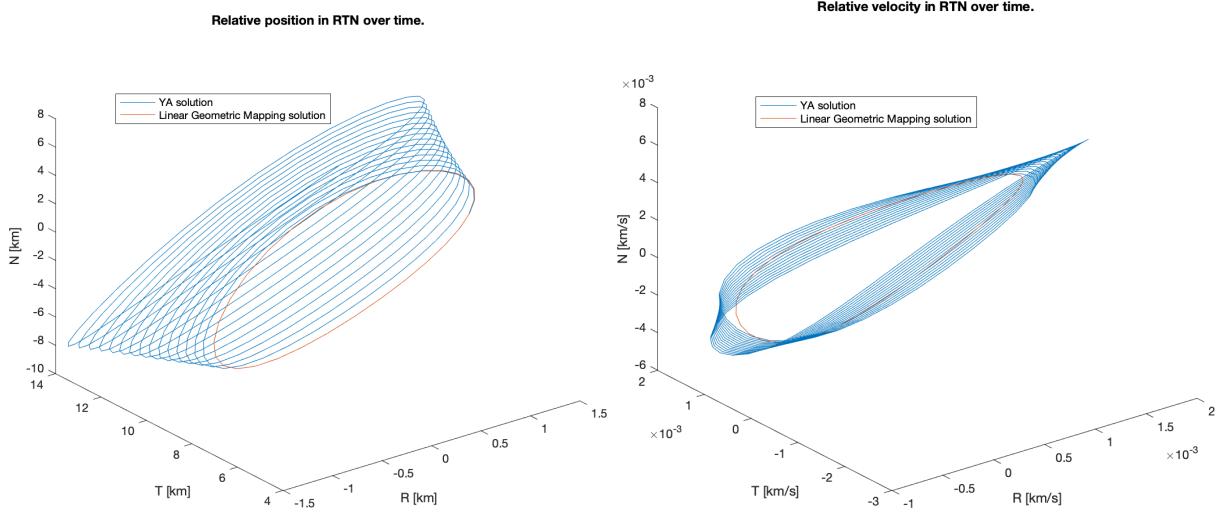


Figure 24: RTN relative position and velocity in 3D space, using the YA solution and the linear geometric mapping solution.

Planar plots:

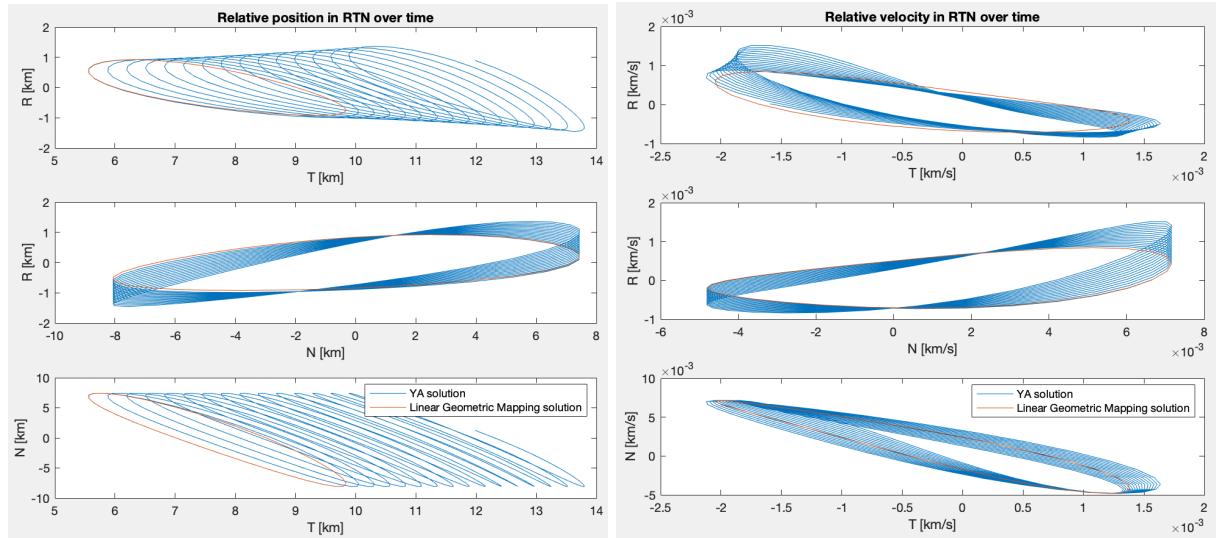


Figure 25: RTN relative position and velocity on each RTN plane, using the YA solution and the linear geometric mapping solution.

- (g) Yes, the results appear as expected. In fact, the results of the linear geometric mapping are quite a bit better than we expected, but it makes sense since our relative orbital elements are so small. It also makes sense that our motion here is quite bounded without having to impose any external constraints since we expect the mapping to be more accurate and properly impose the energy matching condition.

We can use a simple transformation to convert our YA integration constants to quasi non-singular ROEs. Our ROEs converted directly from the YA integration constants are:

$$\begin{bmatrix} \delta a \\ \delta \lambda \\ \delta e_x \\ \delta e_y \\ \delta e_z \\ \delta i_x \\ \delta i_y \end{bmatrix} = \begin{bmatrix} -0.000003412221 \\ 0.0008650867 \\ -0.000174463 \\ -0.00010237677 \\ 0.00017540717 \\ 0.0008639037 \end{bmatrix}.$$

These are very very close! Not exact, but the error is small enough to chalk up to numerical precision errors.

(h) Adding full nonlinear propagation:

3D plots:

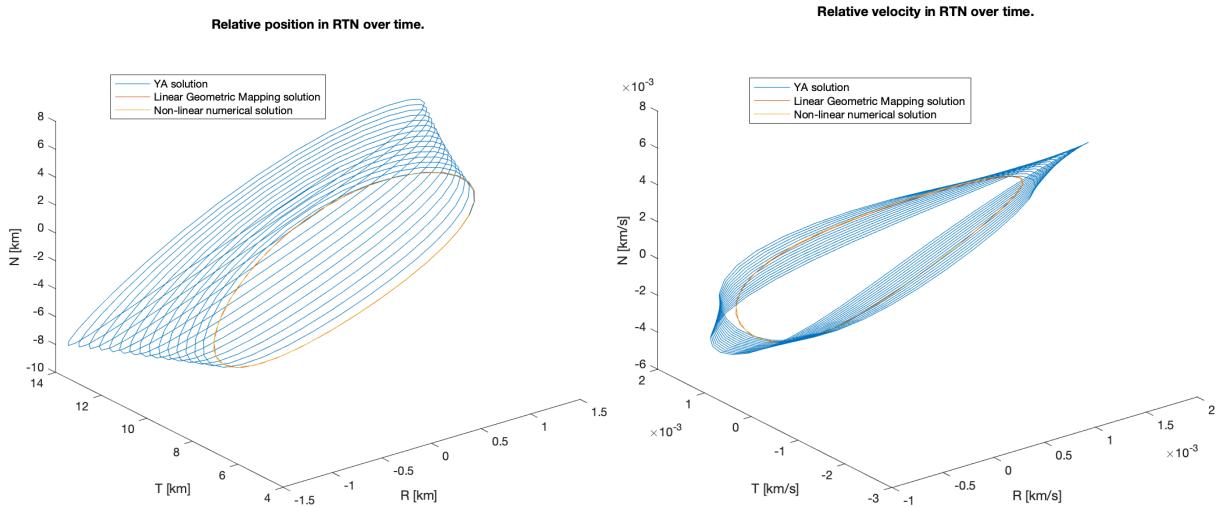


Figure 26: RTN relative position and velocity in 3D space, using the YA solution, the linear geometric mapping solution, and the solution based on non-linear differential equations.

Planar plots:

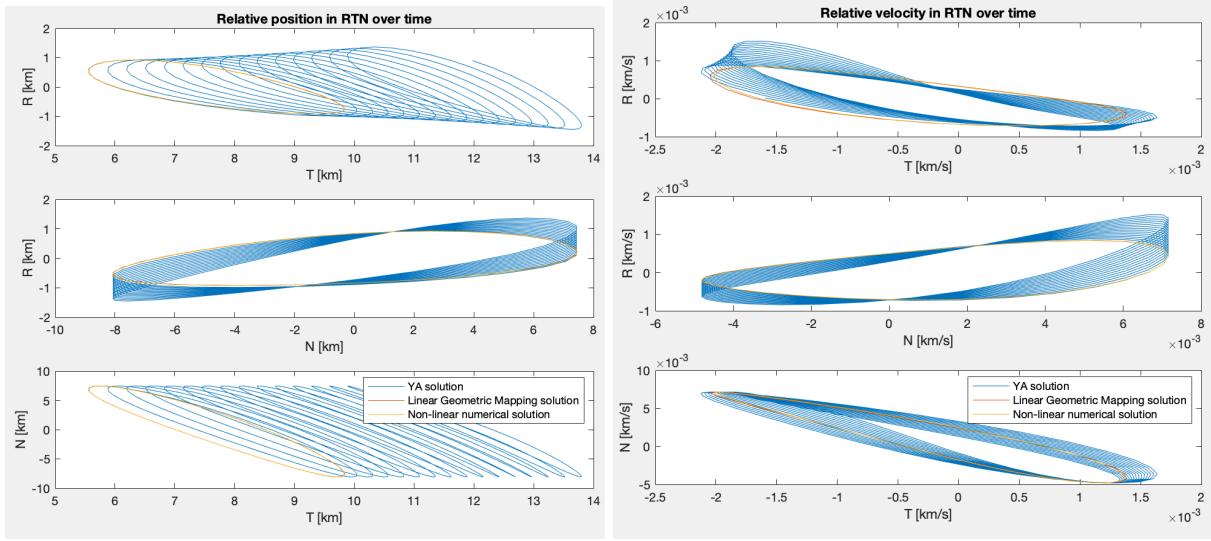


Figure 27: RTN relative position and velocity on each RTN plane, using the YA solution, the linear geometric mapping solution, and the full nonlinear solution.

Errors:

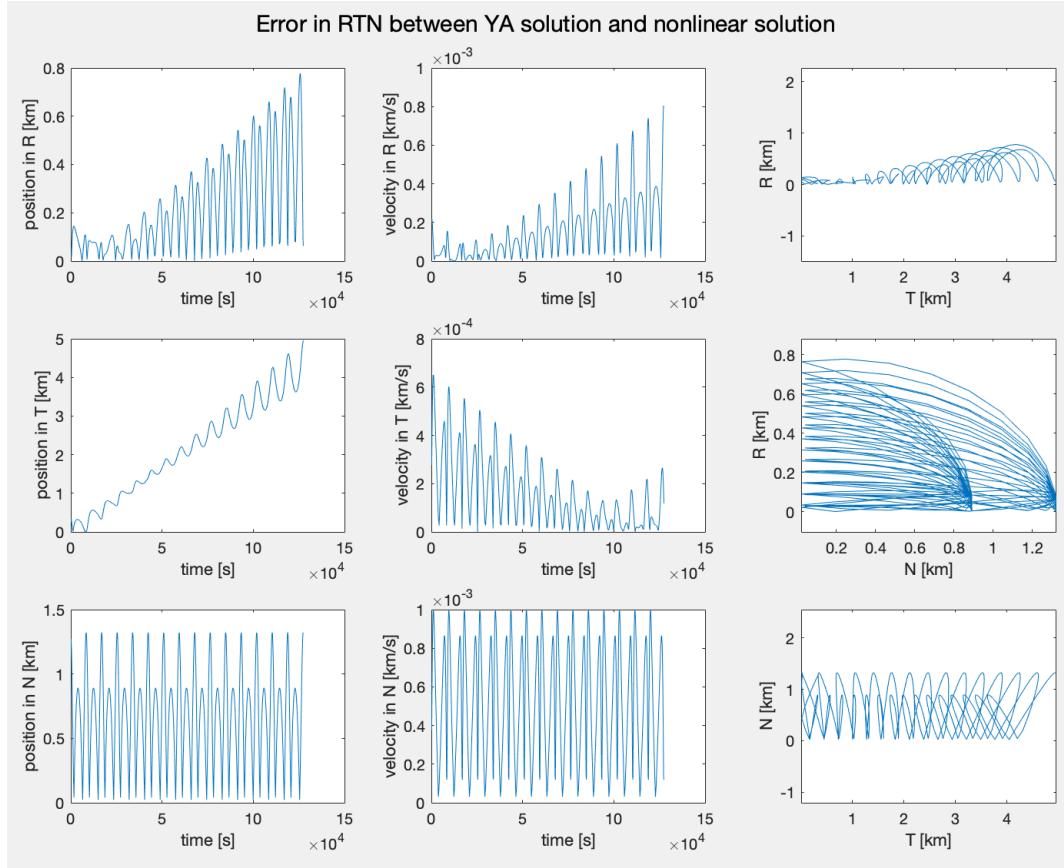


Figure 28: RTN error between the YA solution and the full nonlinear solution.

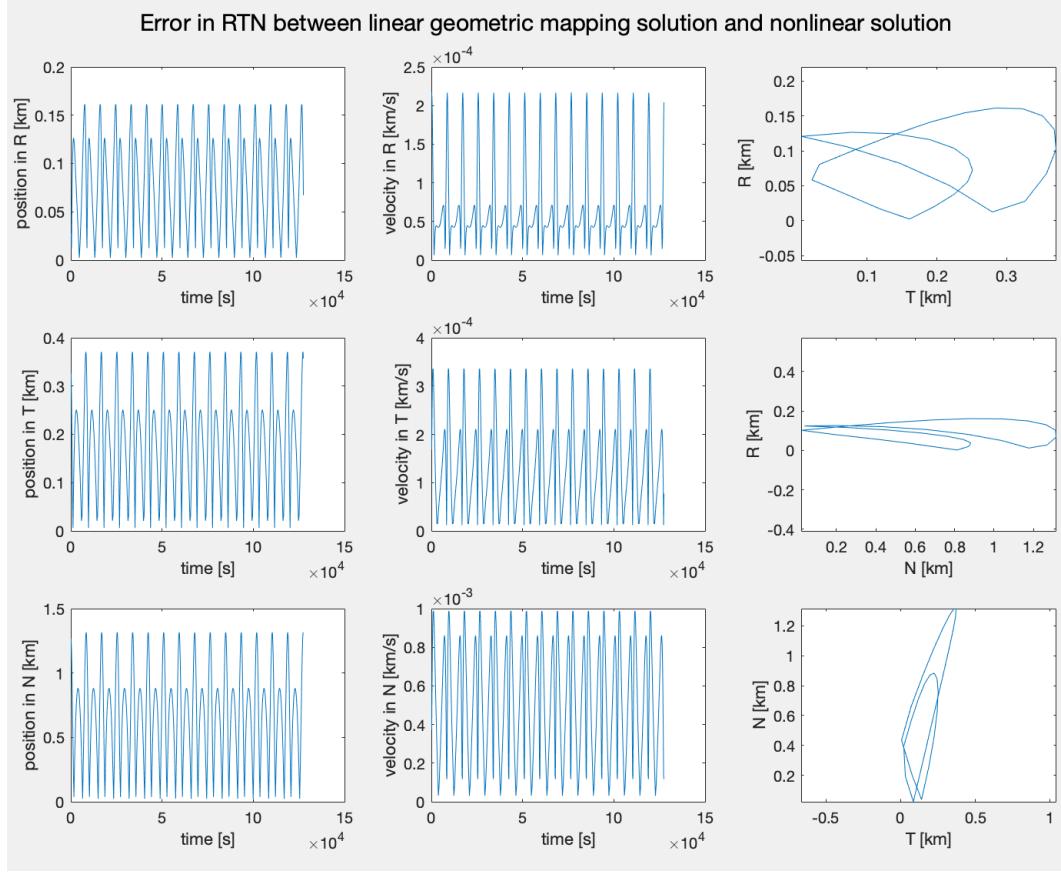


Figure 29: RTN error between the linear geometric mapping solution and the full nonlinear solution.

The linear geometric mapping solution is more accurate, as it bounds relative motion, whereas the YA solutions do not guarantee bounded motion if external constraints are not imposed.

- (i) (i) We will use the same chief orbital elements, but increase the deputy's semi-major axis by 100 km.

Our deputy orbit is:

$$\begin{bmatrix} a \\ e \\ i \\ \Omega \\ \omega \\ \nu \end{bmatrix} = \begin{bmatrix} 9100 \text{ km} \\ 0.1999 \\ 98.01^\circ \\ 0.05 \\ 90.05^\circ \\ 0.01 \end{bmatrix}.$$

We get the following results:

$$\begin{bmatrix} K_1 \\ K_2 \\ K_3 \\ K_4 \\ K_5 \\ K_6 \end{bmatrix} = \begin{bmatrix} 0.0112505086130587 \\ 0.0000369668 \\ -0.0000112123 \\ 0.000854669 \\ 0.00085914 \\ 0.00017735 \end{bmatrix},$$

$$\begin{bmatrix} \delta a \\ \delta \lambda \\ \delta e_x \\ \delta e_y \\ \delta e_z \\ \delta i_x \\ \delta i_y \end{bmatrix} = \begin{bmatrix} 0.011111 \\ 0.00086524 \\ -0.00017444 \\ -0.000100076 \\ 0.000174533 \\ 0.000864172 \end{bmatrix},$$

$$\begin{bmatrix} \delta a \\ \delta \lambda \\ \delta e_x \\ \delta e_y \\ \delta e_z \\ \delta i_x \\ \delta i_y \end{bmatrix} = \begin{bmatrix} 0.01125050 \\ 0.00084369 \\ -0.000158116 \\ 0.00001000763 \\ 0.000177356 \\ 0.0008591439 \end{bmatrix}.$$

from K

3D plots:

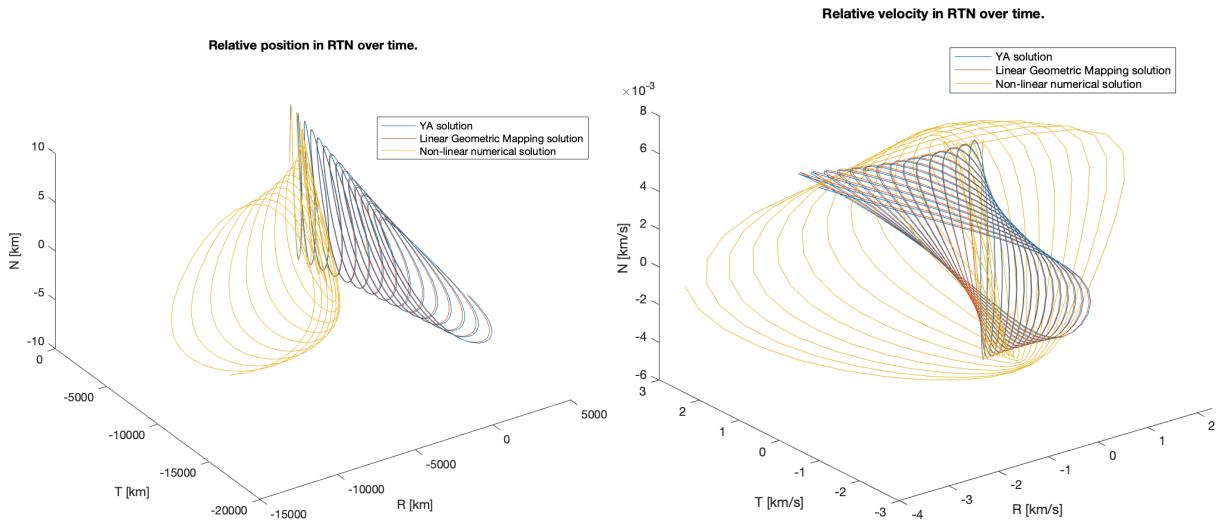


Figure 30: RTN relative position and velocity in 3D space, using the YA solution, the linear geometric mapping solution, and the full nonlinear solution for orbits with non-zero  $\delta a$ .

Planar plots:

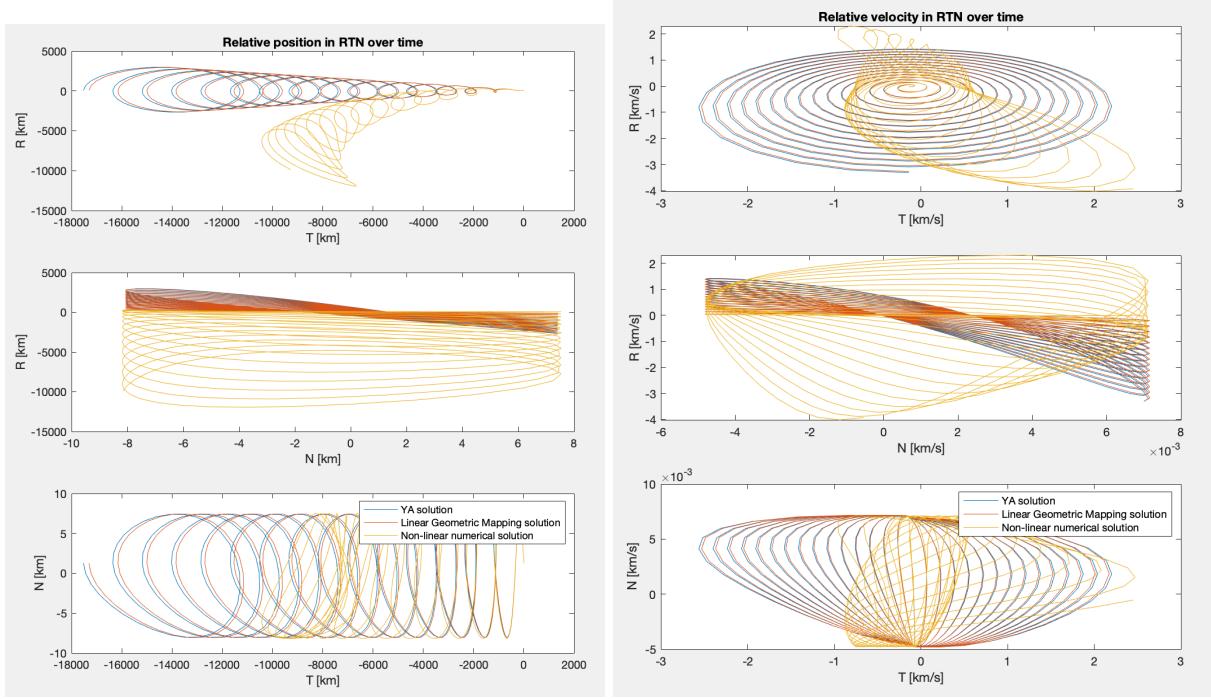


Figure 31: RTN relative position and velocity on each RTN plane, using the YA solution, the linear geometric mapping solution, and the full nonlinear solution for orbits with non-zero  $\delta a$ .

As expected, we see a drift of the deputy in the negative T direction due to a larger semi-major axis meaning that the deputy moves slower (on average) than the chief because the deputy has a larger orbital period.

Errors:

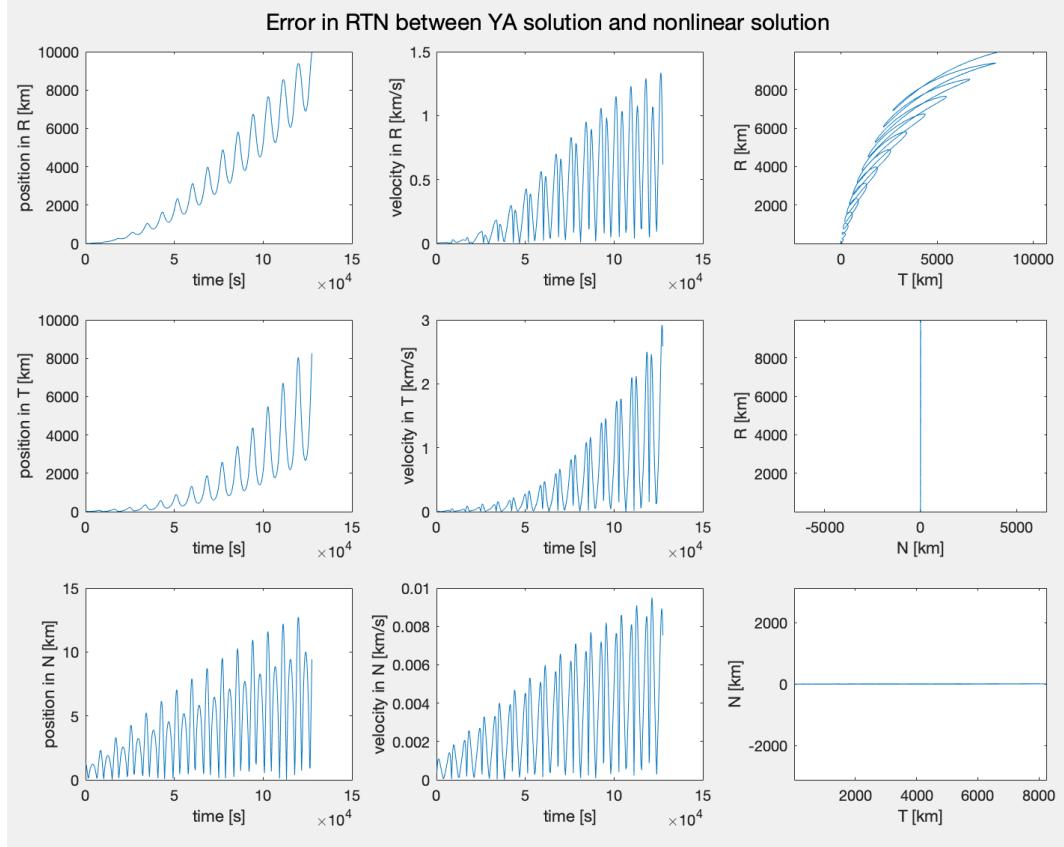


Figure 32: RTN error between the YA solution and the full nonlinear solution for orbits with non-zero  $\delta a$ .

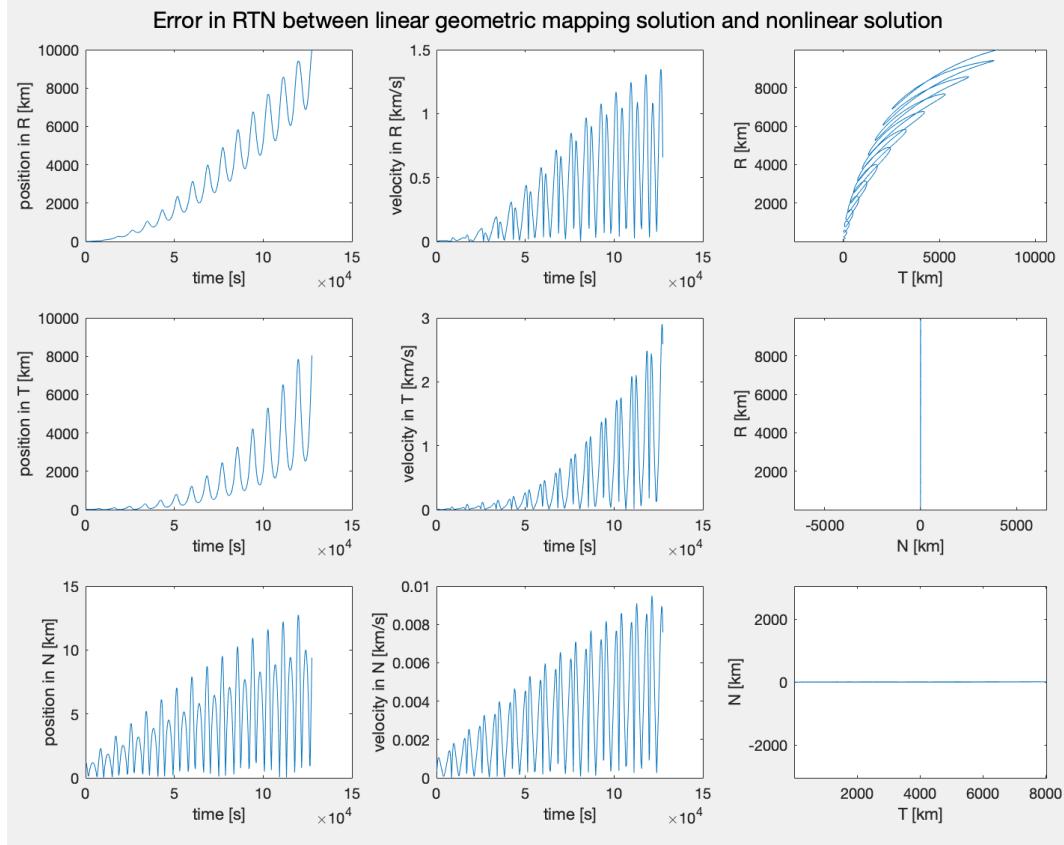


Figure 33: RTN error between the linear geometric mapping solution and the full nonlinear solution for orbits with non-zero  $\delta a$ .

Again, the linear geometric mapping is more accurate than the YA solutions, but neither are good approximations, due to the large difference in semi-major axis.

- (ii) For these orbital elements, we had to raise the semi-major axis of both the chief and deputy by quite a bit to keep the perigee out of the Earth.

For the chief:

$$\begin{bmatrix} a \\ e \\ i \\ \Omega \\ \omega \\ \nu \end{bmatrix} = \begin{bmatrix} 23000 \text{ km} \\ 0.7 \\ 98^\circ \\ 0 \\ 90^\circ \\ 0 \end{bmatrix}.$$

Our deputy orbit is:

$$\begin{bmatrix} a \\ e \\ i \\ \Omega \\ \omega \\ \nu \end{bmatrix} = \begin{bmatrix} 23000 \text{ km} \\ 0.7 \\ 98.01^\circ \\ 0.05 \\ 90.05^\circ \\ 0.01 \end{bmatrix}.$$

We get the following results:

$$\begin{bmatrix} K_1 \\ K_2 \\ K_3 \\ K_4 \\ K_5 \\ K_6 \end{bmatrix} = \begin{bmatrix} -0.00001132007 \\ 0.000042185458 \\ 0.000006401528 \\ 0.000811694045 \\ 0.00086400121 \\ 0.000175385245 \end{bmatrix},$$

$$\begin{bmatrix} \delta a \\ \delta \lambda \\ \delta e_x \\ \delta e_y \\ \delta i_x \\ \delta i_y \end{bmatrix} = \begin{bmatrix} 0 \\ 0.0007732087 \\ -0.000610865 \\ -0.0000002665402 \\ 0.0001745329 \\ 0.0008641719 \end{bmatrix},$$

$$\begin{bmatrix} \delta a \\ \delta \lambda \\ \delta e_x \\ \delta e_y \\ \delta i_x \\ \delta i_y \end{bmatrix}_{\text{from } K} = \begin{bmatrix} -0.00001132007 \\ 0.000773378 \\ -0.00061099959 \\ -0.0000032647795 \\ 0.0001753852 \\ 0.0008640012 \end{bmatrix}.$$

3D plots:

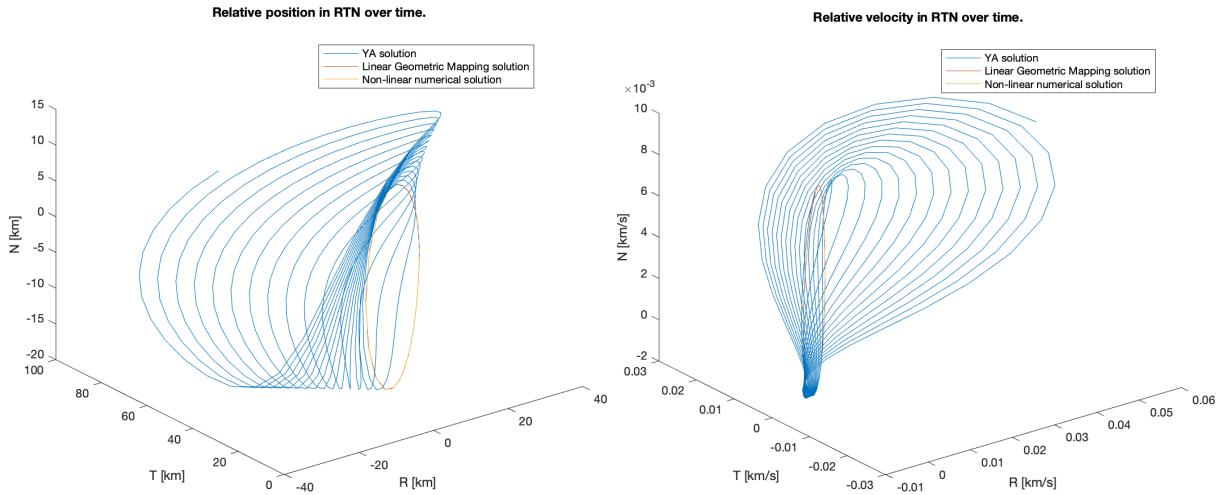


Figure 34: RTN relative position and velocity in 3D space, using YA solution and linear geometric mapping solution and full nonlinear solution for orbits with high eccentricity.

Planar plots:

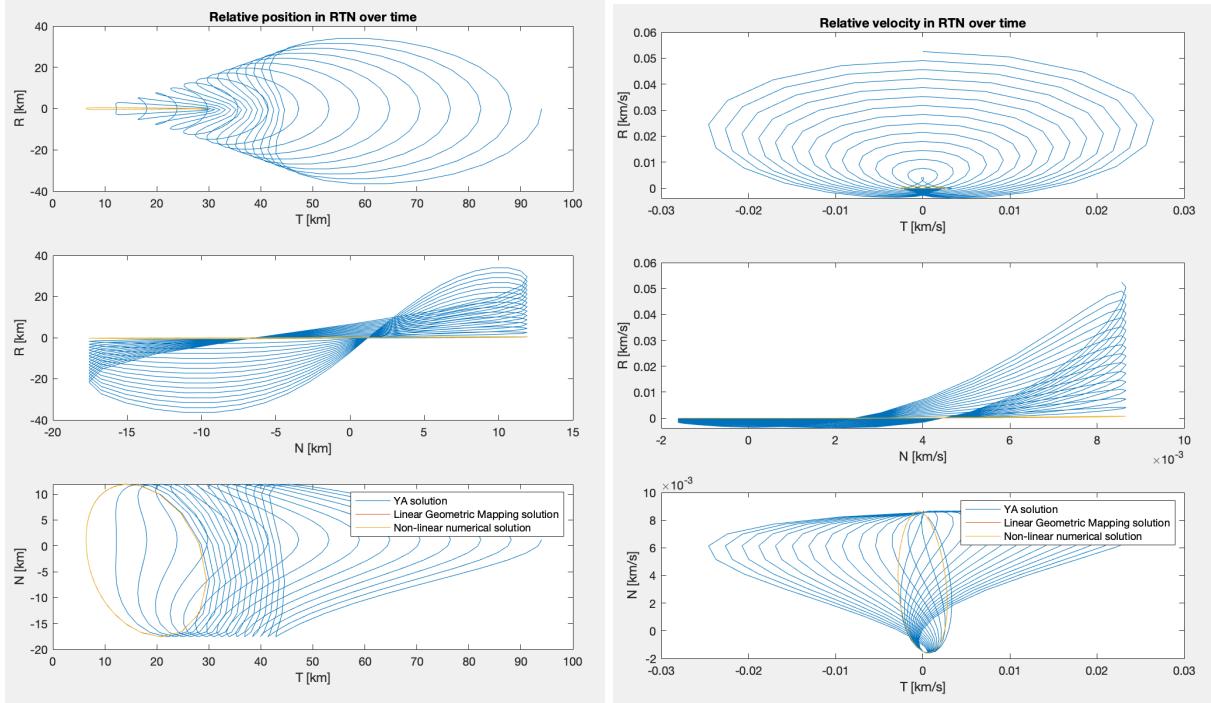


Figure 35: RTN relative position and velocity on each RTN plane, using the YA solution, the linear geometric mapping solution, and the full nonlinear solution for orbits with high eccentricity.

Here we see much thinner ellipses, elongated in the  $T$  direction. This makes sense because the high eccentricity exaggerates the along-track separation near periapsis while minimizing the along-track separation near apoapsis. Furthermore, due to the large eccentricity, we seem to get interesting twists in our relative motion ellipses, which is consistent with what we saw in lecture.

Errors:

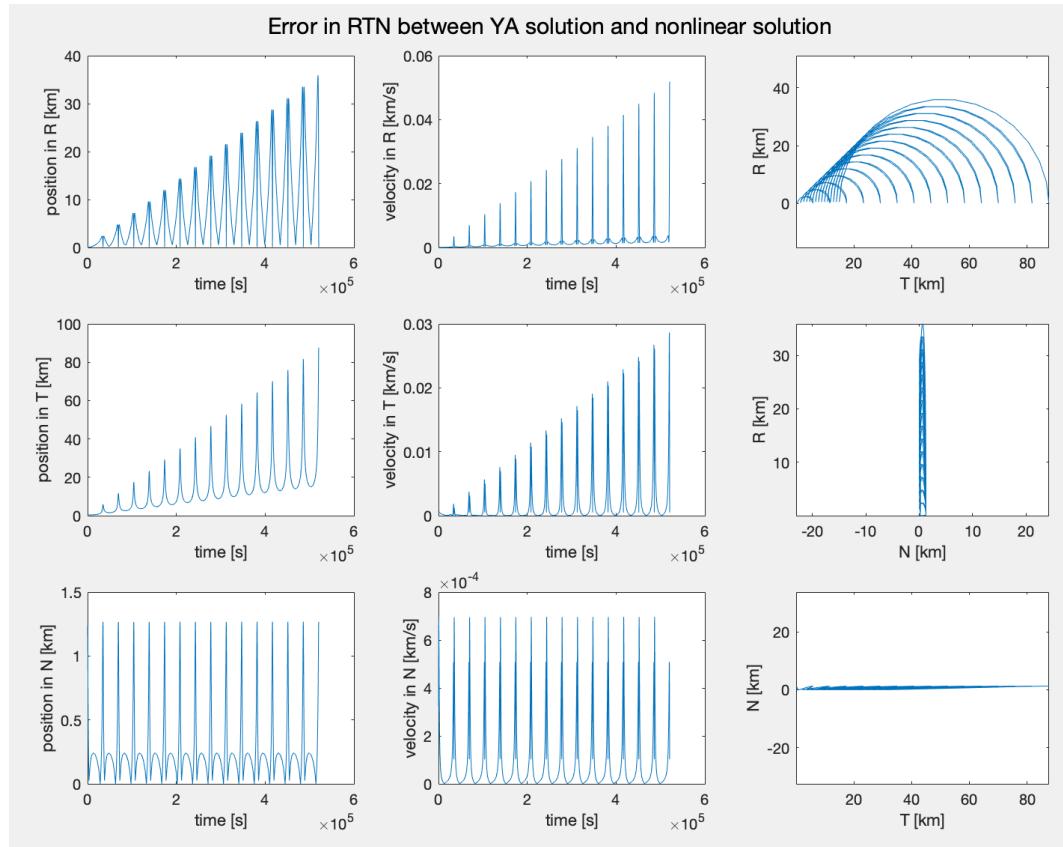


Figure 36: RTN error between the YA solution and the full nonlinear solution for orbits with high eccentricity.

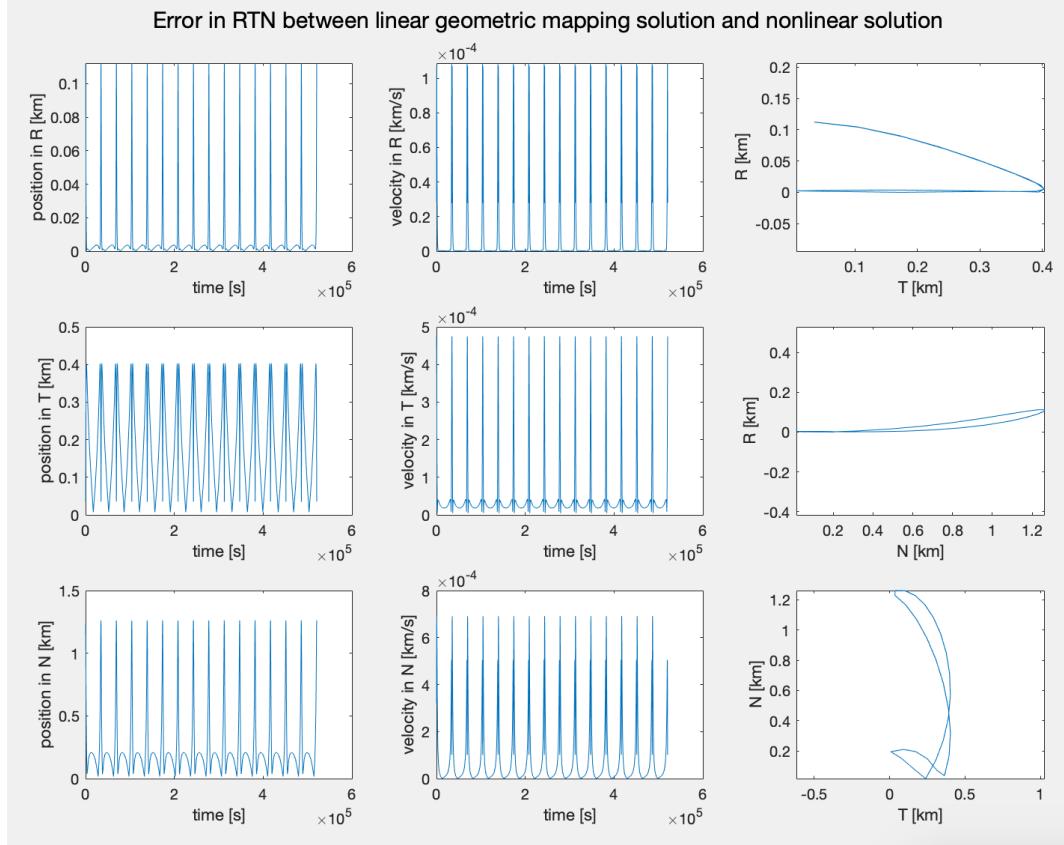


Figure 37: RTN error between the linear geometric mapping solution and the full nonlinear solution for orbits with high eccentricity.

Again, here the linear geometric mapping is the more accurate solution over the YA solutions, mostly due to the bounded motion, as before.

## 4 Problem Set 4: Linearized relative orbit motion in relative orbit elements.

### 4.1 Problem 1. Relative Orbits.

#### 4.1.1 Initial Chief OE

Osculating initial conditions for chief. These are the same as used previously, and carry the same justification.

$$\begin{bmatrix} a \\ e \\ i \\ \Omega \\ \omega \\ \nu \end{bmatrix} = \begin{bmatrix} 7000\text{km} \\ 0.001 \\ 98^\circ \\ 0 \\ 90^\circ \\ 0 \end{bmatrix}.$$

#### 4.1.2 Initial Deputy ROE

Relative state of deputy (quasi-non-singular) relative orbit elements.

$$a_c \begin{bmatrix} \delta a \\ \delta \lambda \\ \delta e \\ \delta i \\ \delta \Omega \\ \delta \omega \\ \delta \nu \end{bmatrix} = \begin{bmatrix} 0 \\ 100 \\ 50 \\ 100 \\ 30 \\ 200 \end{bmatrix} [\text{m}]$$

$$\begin{bmatrix} \delta a \\ \delta \lambda \\ \delta e \\ \delta i \\ \delta \Omega \\ \delta \omega \\ \delta \nu \end{bmatrix} = \begin{bmatrix} 0 \\ 1.42857142e-05 \\ 7.142857142e-06 \\ 1.42857142e-05 \\ 4.28571423e-06 \\ 2.8571423e-05 \end{bmatrix}$$

$$\begin{bmatrix} a \\ e \\ i \\ \Omega \\ \omega \\ \nu \end{bmatrix} = \begin{bmatrix} 7000\text{km} \\ 0.00101431 \\ 1.710427 \\ 0.00002885222 \\ 1.56375419 \\ 0.007074779 \end{bmatrix}$$

#### 4.1.3 Baseline Simulation

The plots below are for 15 orbits of simulation time.

**(a) and (c):** Osculating and Mean quasi-non-singular orbital elements:

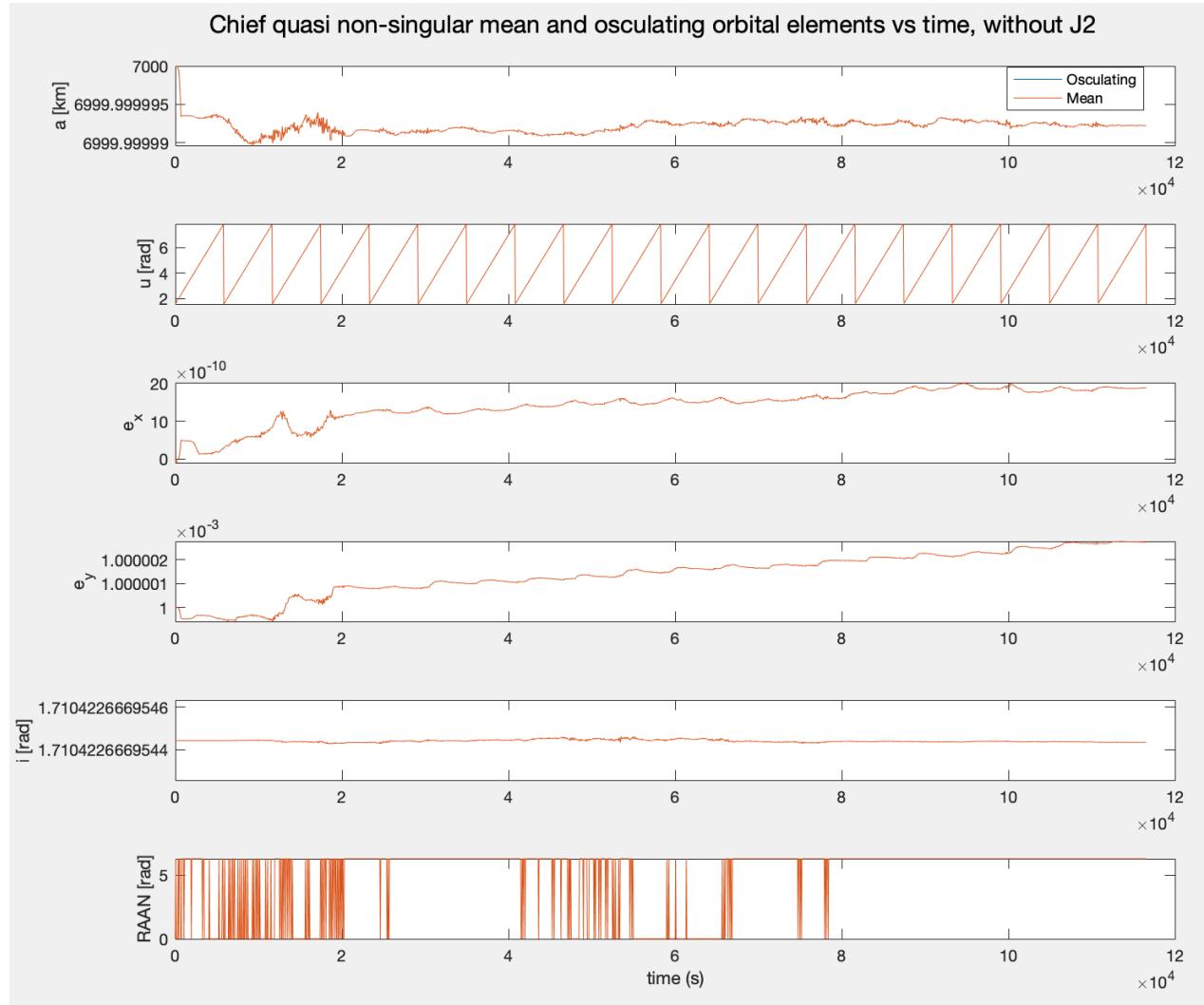


Figure 38: Chief osculating and mean quasi-non-singular orbital elements vs. time. Without J2 effect.

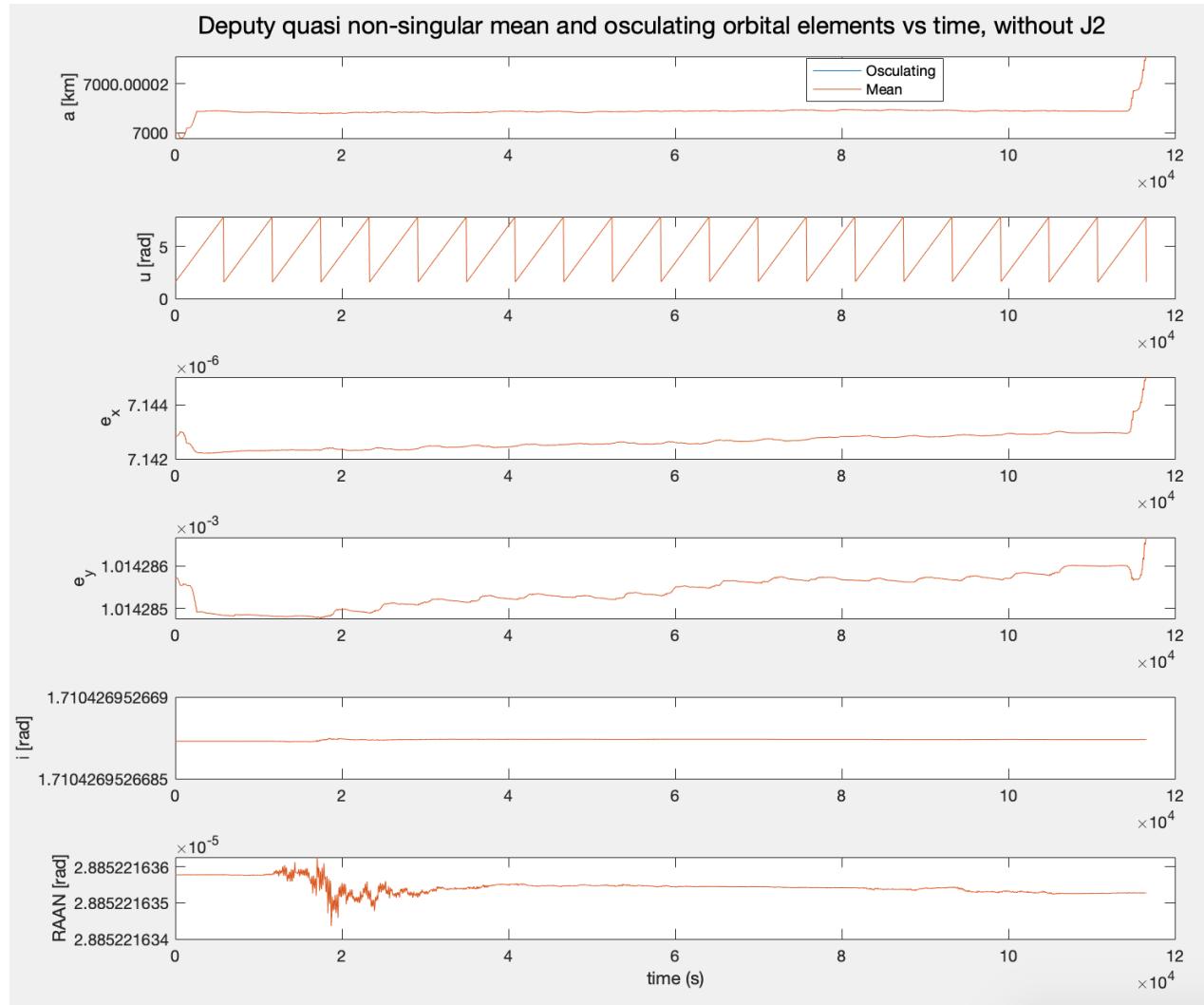


Figure 39: Deputy osculating and mean quasi-non-singular orbital elements vs. time. Without J2 effect.

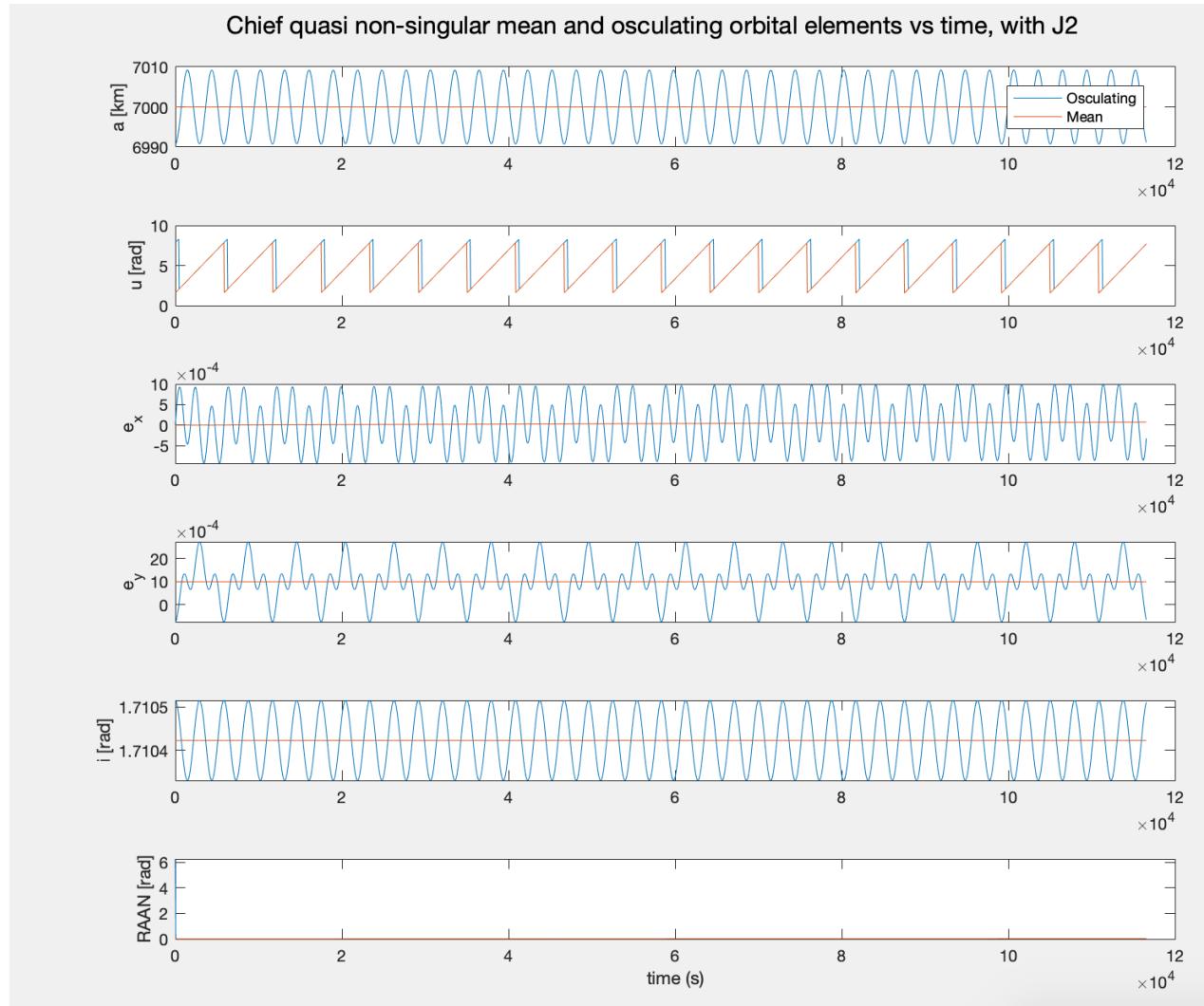


Figure 40: Chief osculating and mean quasi-non-singular orbital elements vs. time. With J2 effect.

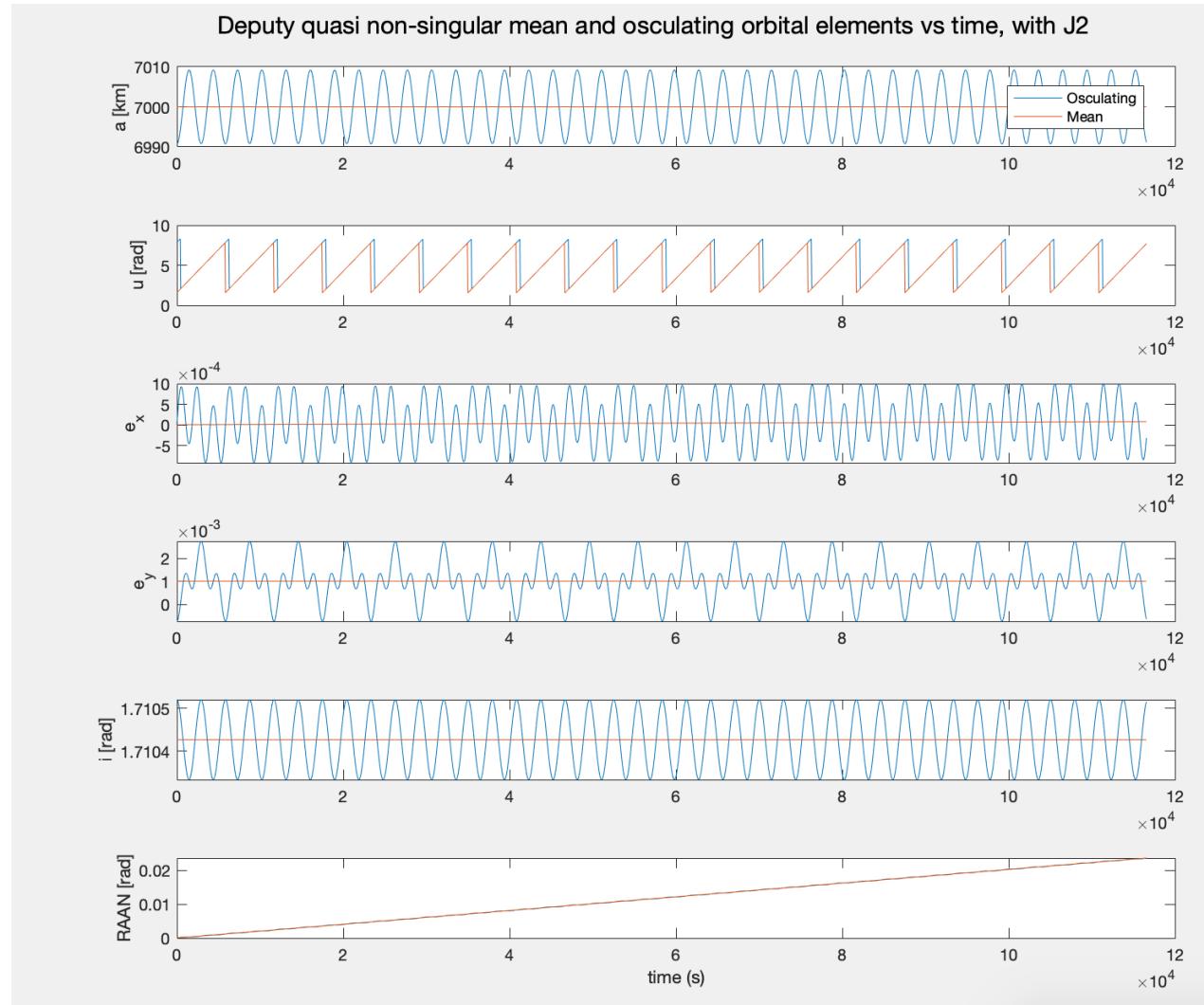


Figure 41: Deputy osculating and mean quasi-non-singular orbital elements vs. time. With J2 effect.

**(b) and (d):** Osculating and Mean quasi-non-singular relative orbital elements:

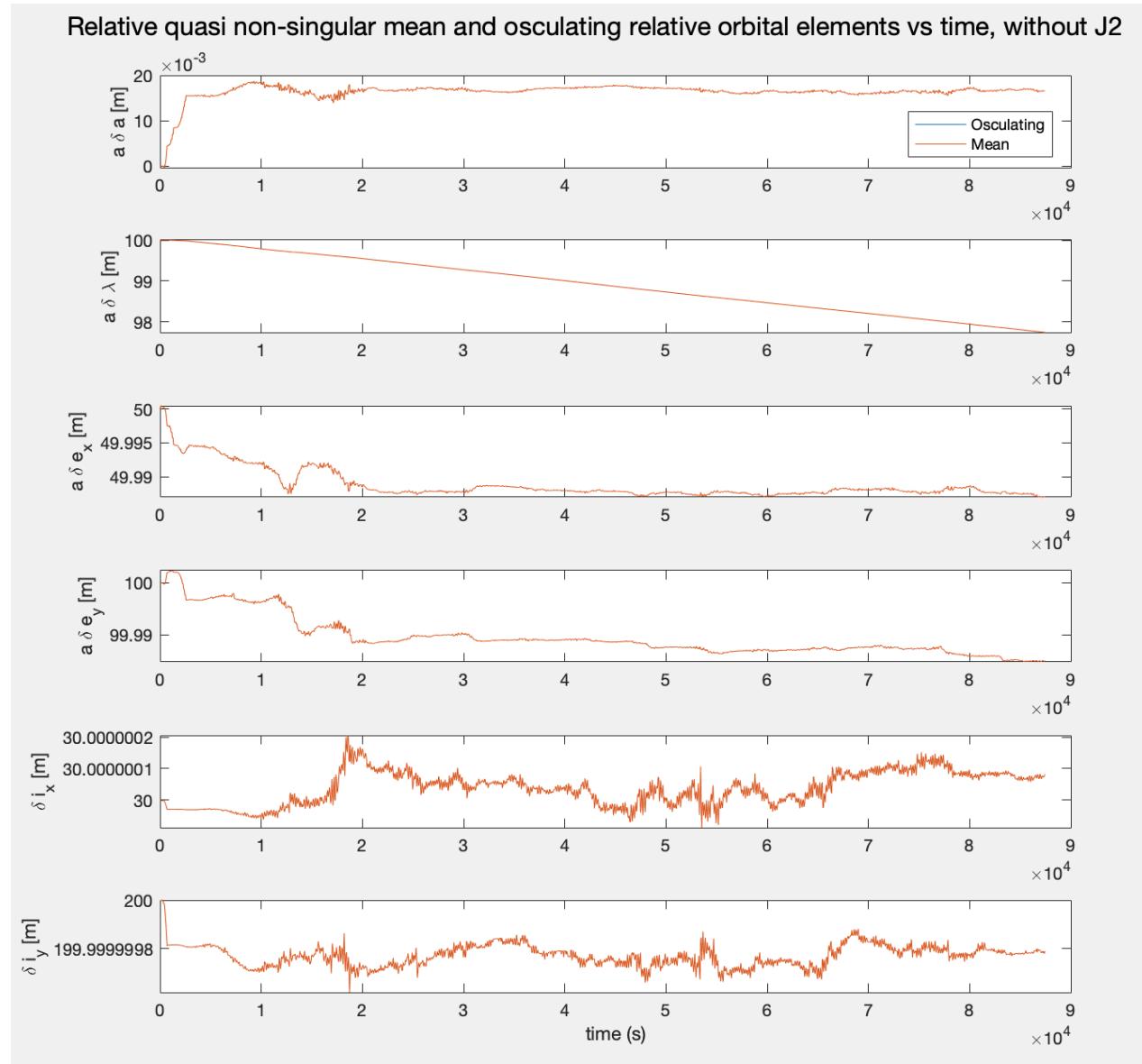


Figure 42: Deputy osculating and mean quasi-non-singular relative orbital elements with respect to chief vs. time. Without J2 effect.

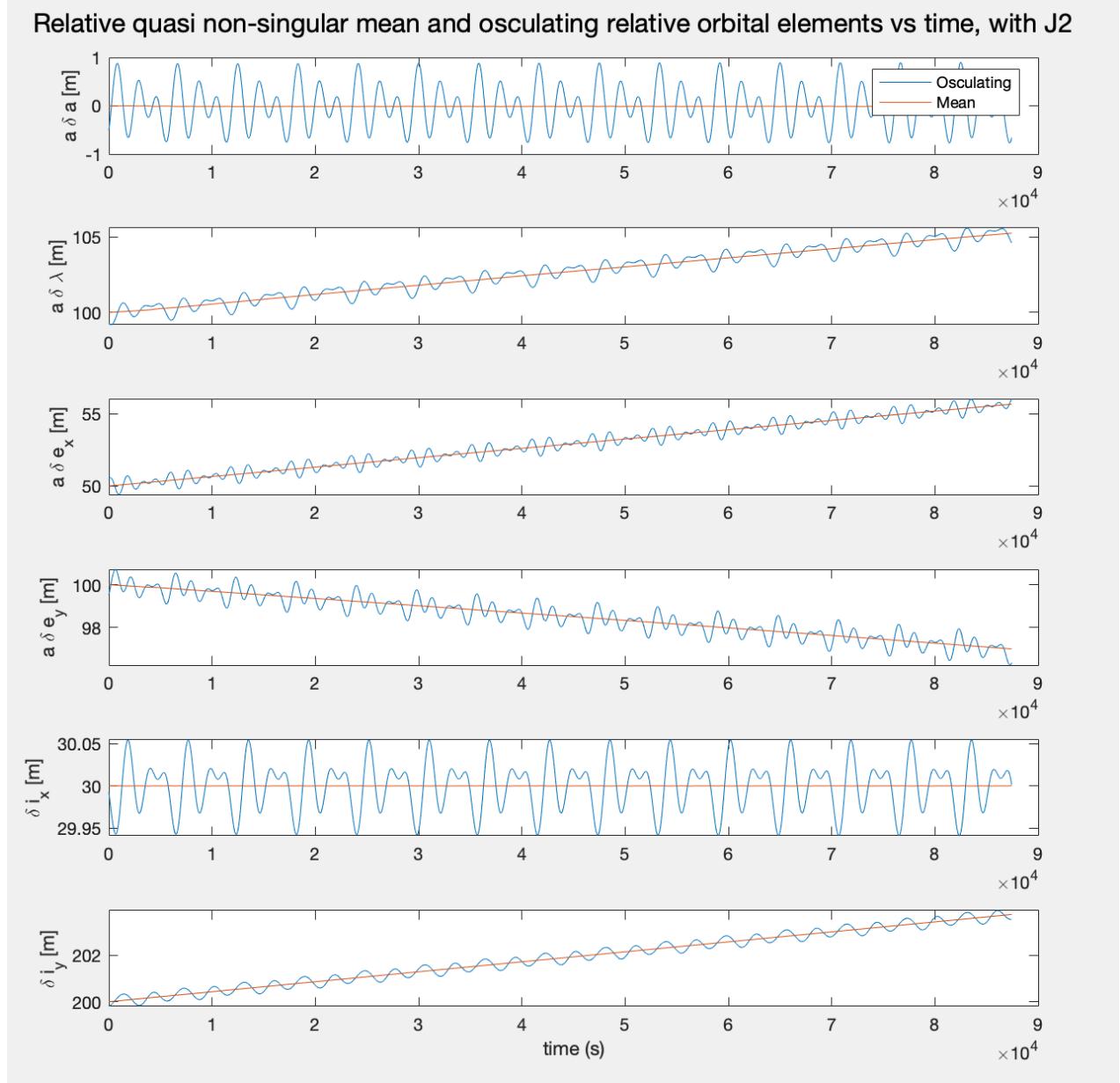


Figure 43: Deputy osculating and mean quasi-non-singular relative orbital elements with respect to chief vs. time. With J2 effect.

#### 4.1.4 RTN Plots for Baseline Simulation

The plots below are for 45 orbits of simulation time.

The results are as expected. We see some noise in the TR plot, but it mostly stays as the same ellipse. And when we add the J2 affect, we see the shrinking of the ellipse in the NR plot, as expected.

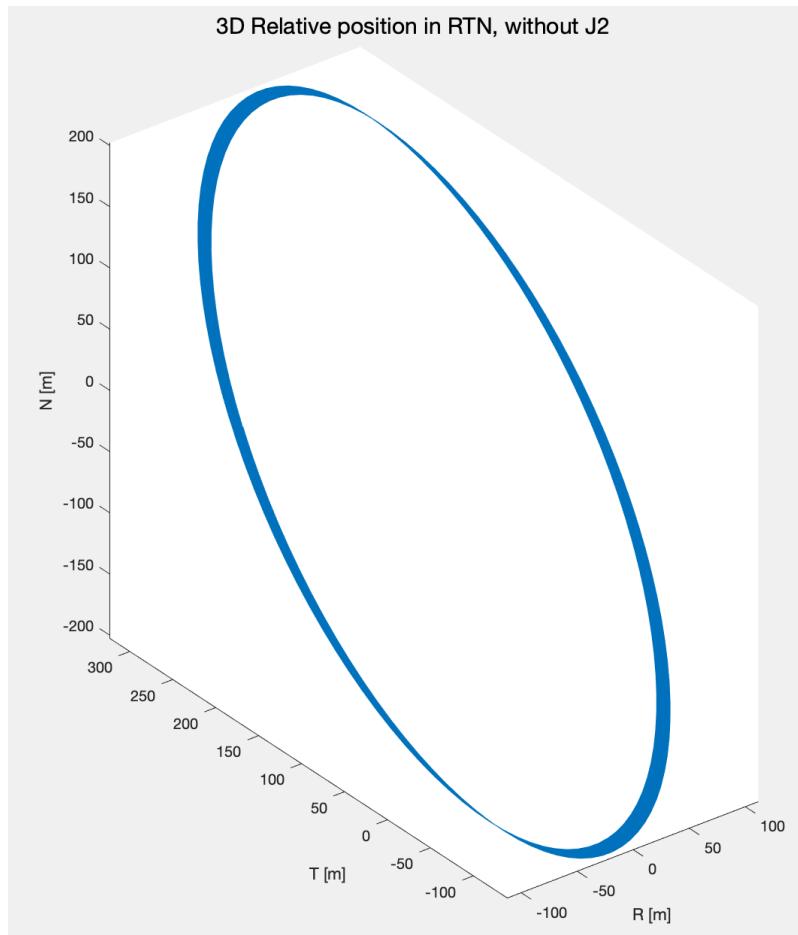


Figure 44: 3D RTN plot of deputy's motion relative to chief. Without J2

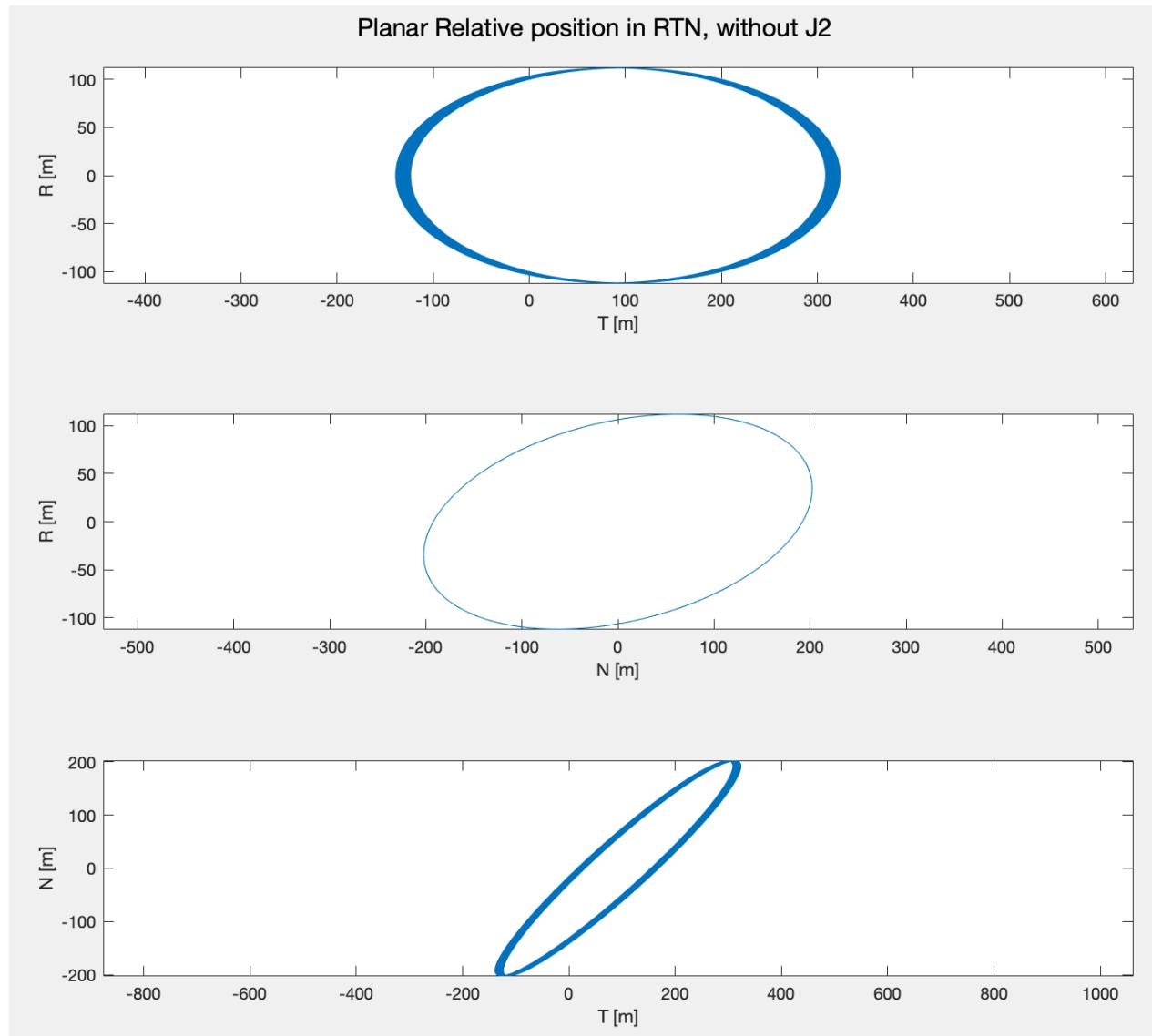


Figure 45: Planar RTN plots of deputy's motion relative to chief. Without J2

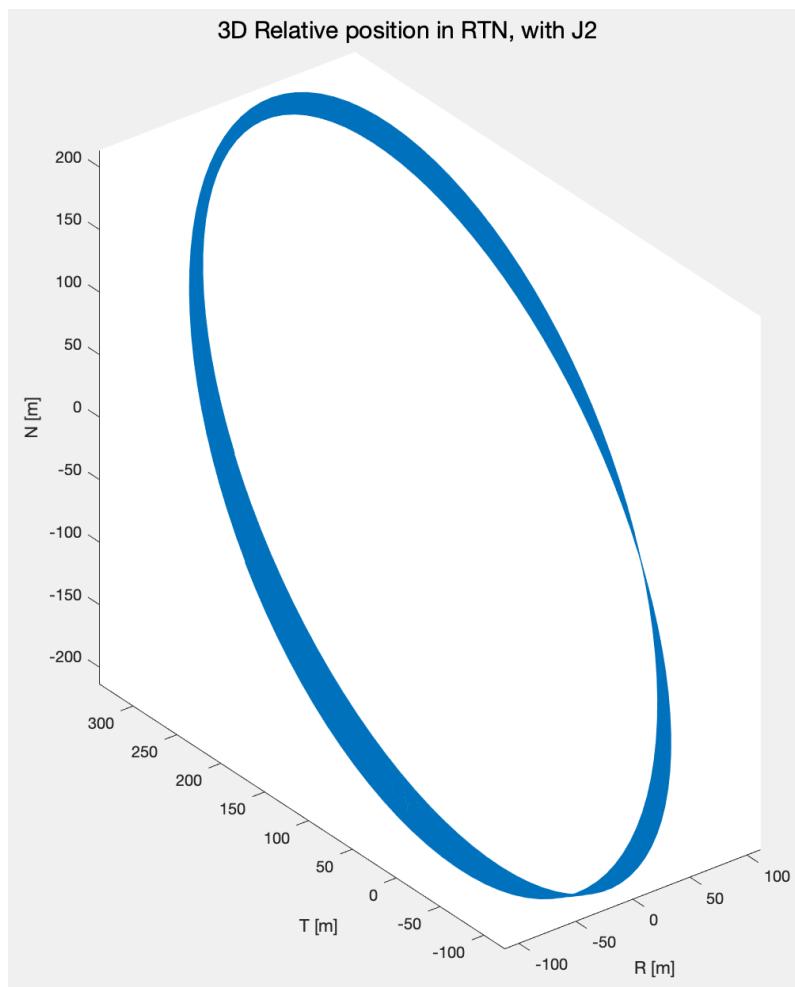


Figure 46: 3D RTN plot of deputy's motion relative to chief. With J2

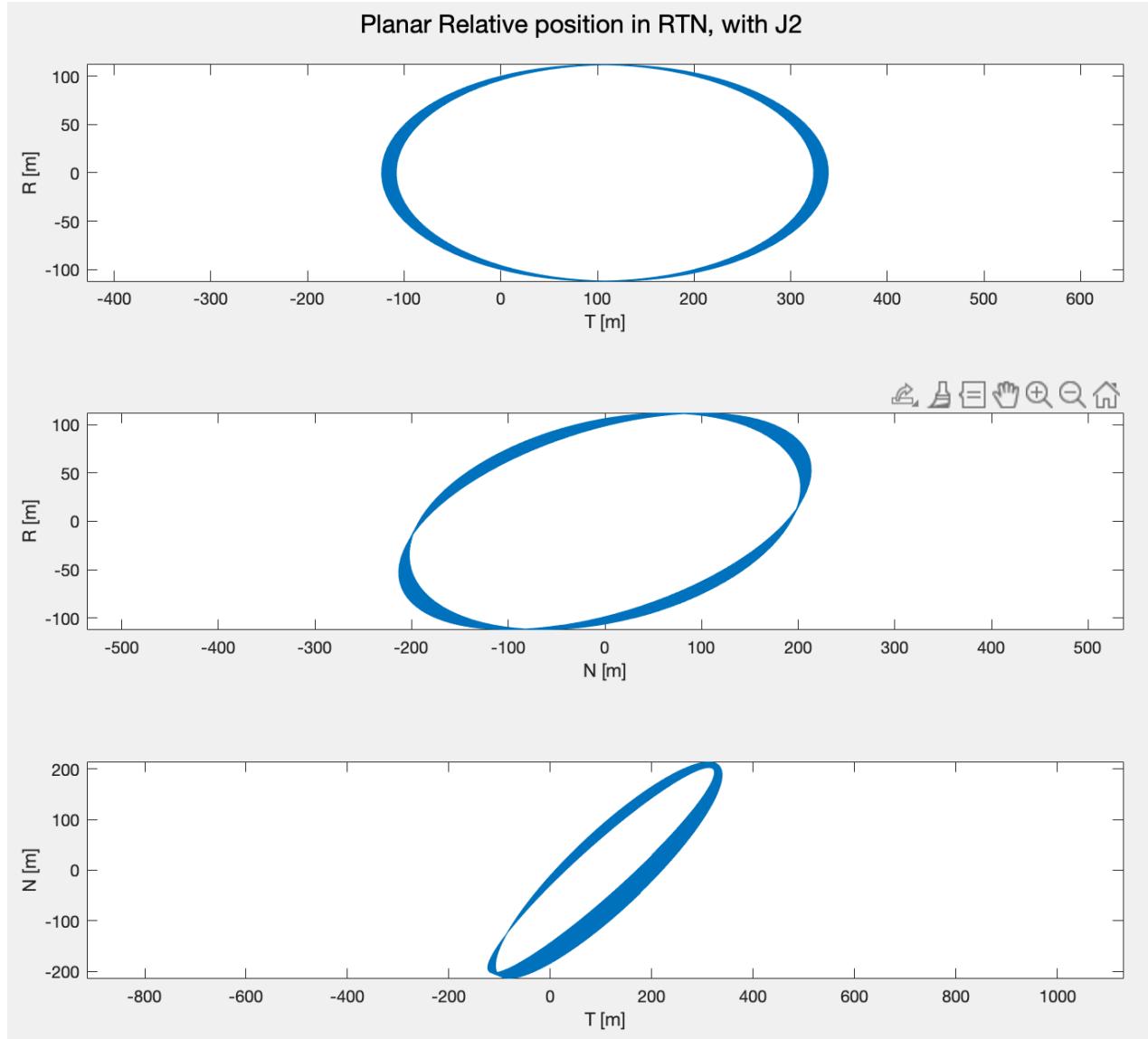


Figure 47: Planar RTN plots of deputy's motion relative to chief. With J2

#### 4.1.5 QNS ROE Plots for Baseline Simulation

The results are as expected. These plots are quite zoomed in, so we get to see a lot of the numerical error. Specifically notice that for the  $\delta\lambda$  plot, it looks like it is drifting a lot, but we believe it is just due to numerical error.

With J2, this also makes a lot of sense. We see the relative eccentricity vector drift in a circle, and the relative inclination vector drift in a vertical line.

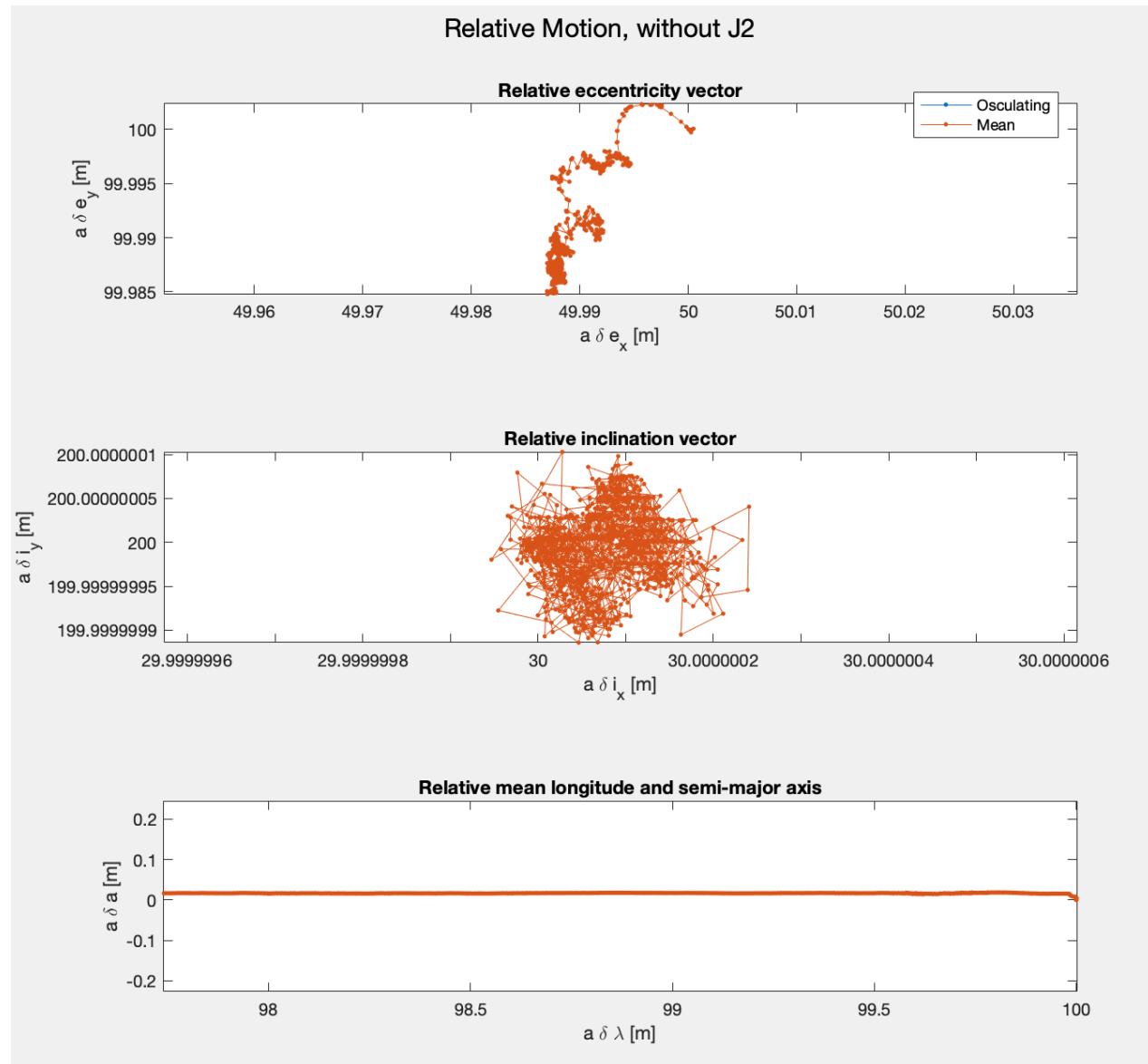


Figure 48: Osculating and mean relative quasi-non-singular orbital elements, decomposed into eccentricity vector, inclination vector, and  $\delta a - \delta\lambda$  plots. Without J2

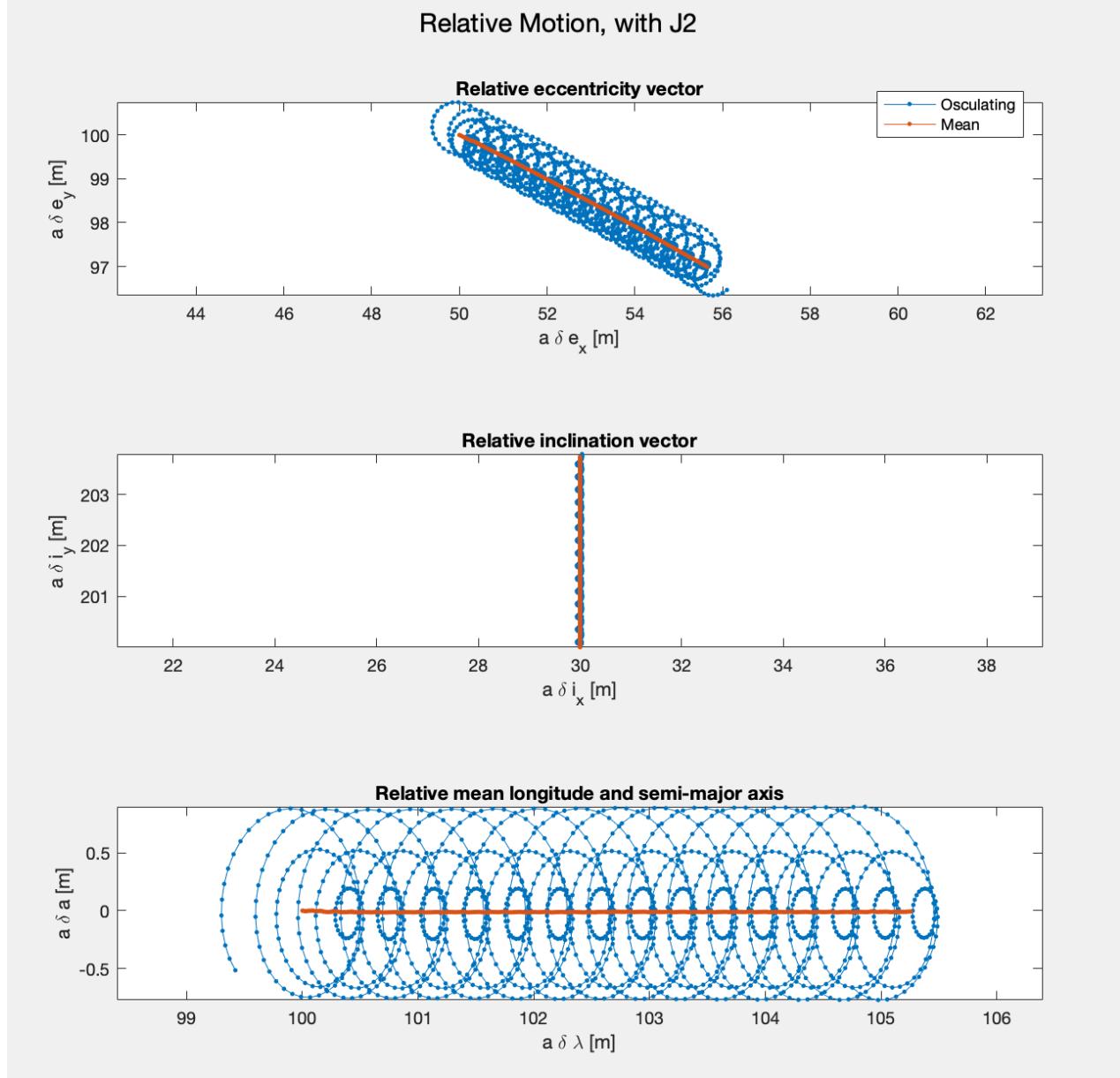


Figure 49: Osculating and mean relative quasi-non-singular orbital elements, decomposed into eccentricity vector, inclination vector, and  $\delta a - \delta \lambda$  plots. With J2

#### 4.1.6 Minimum Change to Remove J2 Secular Effects

Mean J2 effects are functions of  $a$ ,  $e$ , and  $i$ . So, in order for us to remove any secular effects due to J2 that act on the relative motion between the chief and the deputy, we know we'll need to modify  $\delta e_x$ ,  $\delta e_y$ ,  $\delta i_x$ , and  $\delta i_y$ . Looking at the equations on slide 30 of lecture 5/6, we see that the only relative orbital element that affects  $\vec{\delta i}$  and  $\delta \lambda$  is  $\delta i_x$ . We can easily see that setting  $\delta i_x = 0$ , will remove this secular relative motion. Furthermore, we see that the J2 effect on the eccentricity vector only acts periodically (sometimes over long time periods albeit) since it only affects the phase angle of the eccentricity vector, thus we don't have any secular effects to address there. So the only maneuver we need to do is one that

gets  $\delta i_x = 0$ .

That maneuver is a cross-track burn when  $u_M = 0$ . Specifically,

$$\begin{aligned} a\delta i_x &= \delta v_n \cos(u_M)/n \\ a\delta i_y &= \delta v_n \sin(u_M)/n \end{aligned}$$

In order to avoid affecting  $\delta i_y$ , and maximize the affect on  $\delta i_x$ , we select a  $u_M = 0$  where  $u_M$  is the Maneuver's mean argument of latitude ( $u_M = \omega + M$ ). Thus the  $\delta v_n$  magnitude of the maneuver is:

$$\delta v_n = a * n * \delta i_x$$

where  $n$  is mean motion.

$$\delta v_n = \sqrt{\frac{\mu}{a^3}} * a * \delta i_x$$

$$\delta v_n = (0.0011[1/s]) * (30[m])$$

$$\delta v_n = -32[mm/s]$$

#### 4.1.7 Simulation with No J2 Secular Effects

Yes, the secular effects are removed. Notice that  $\delta i$  vector stays constant.  $\delta e$  still rotates around in a circle, but the secular behavior of this is zero. Of course, we still see the noise in  $\delta a$  causing a very small drift in  $\delta \lambda$ , but this is just due to numerical error.

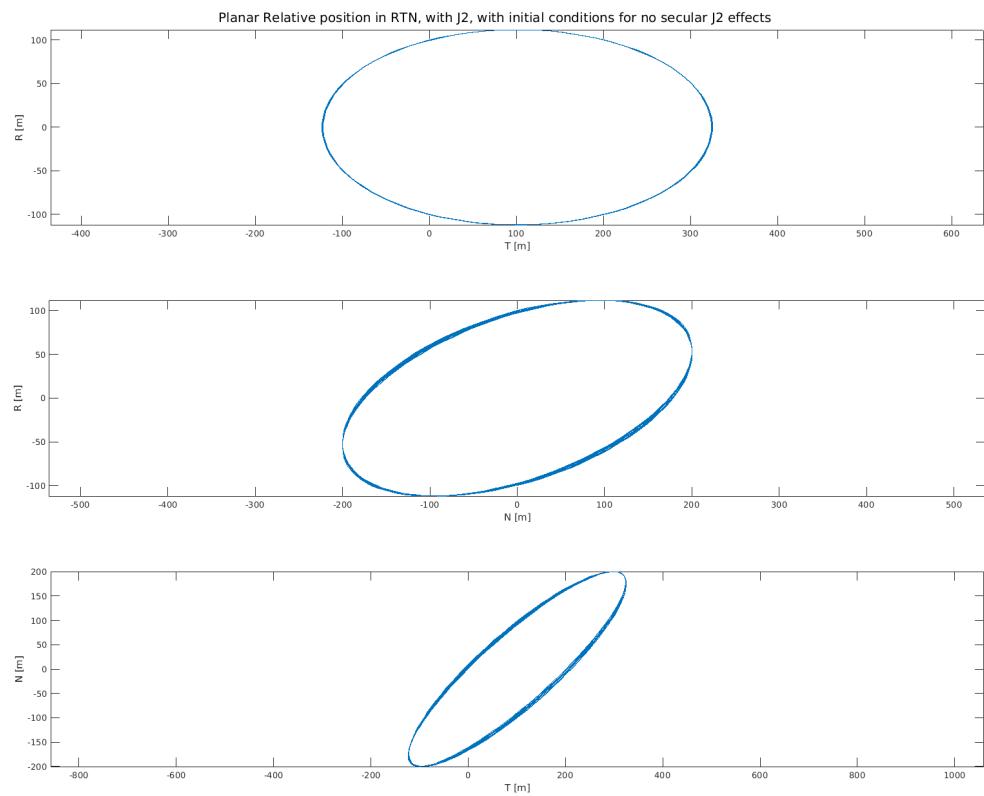


Figure 50: RTN planar plots with  $\delta i_y = 0$ . With J2.

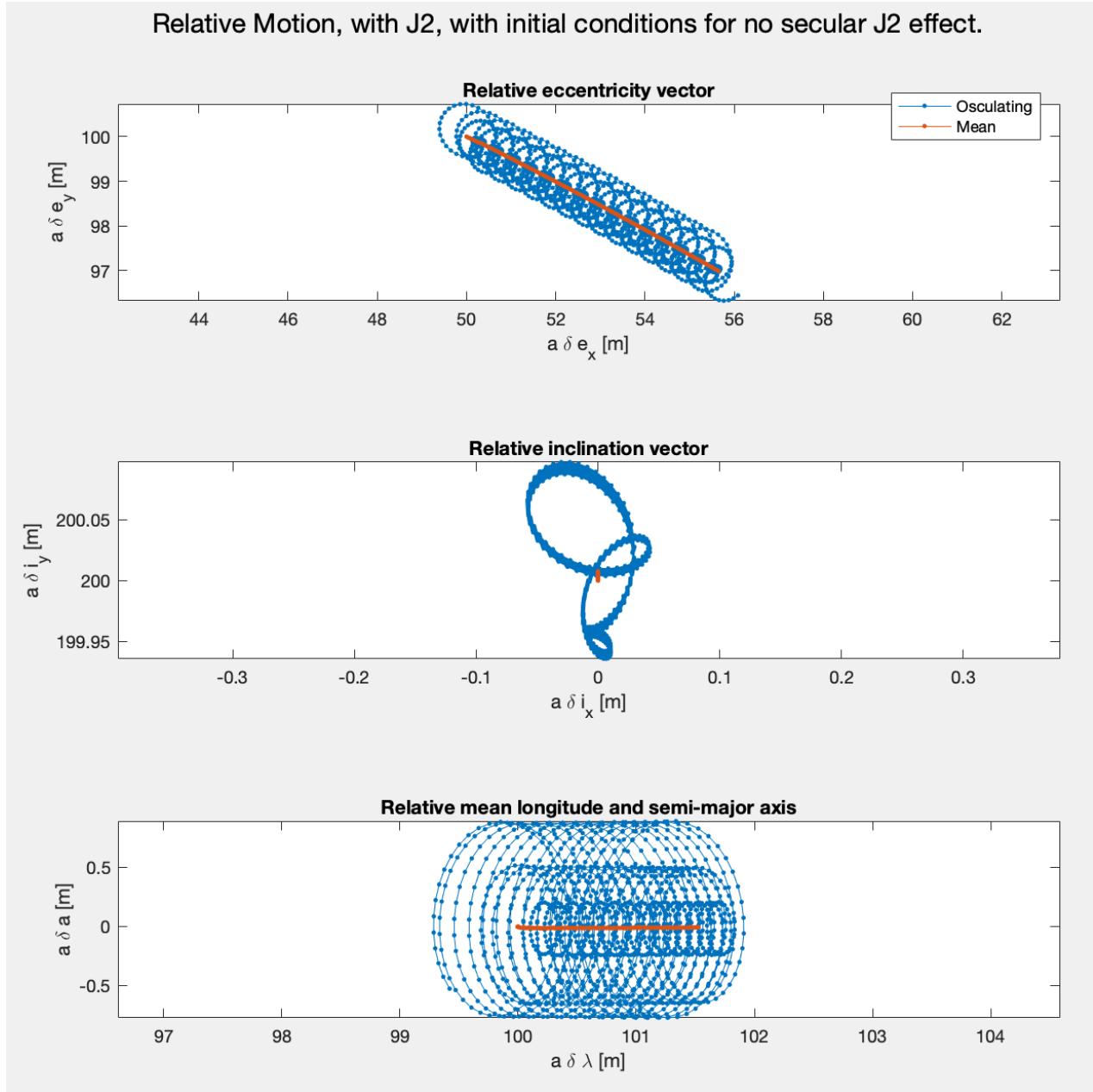


Figure 51: OE plots with  $\delta i_y = 0$ . With J2.

#### 4.1.8 Linearized Relative Motion Under J2

When we apply the STM to the initial conditions from part 2 and recreate the mean orbital elements shown in plot 49. When applying STM to the initial conditions with  $i_x = 0$  we see that there is no drift of inclination or eccentricity vectors.

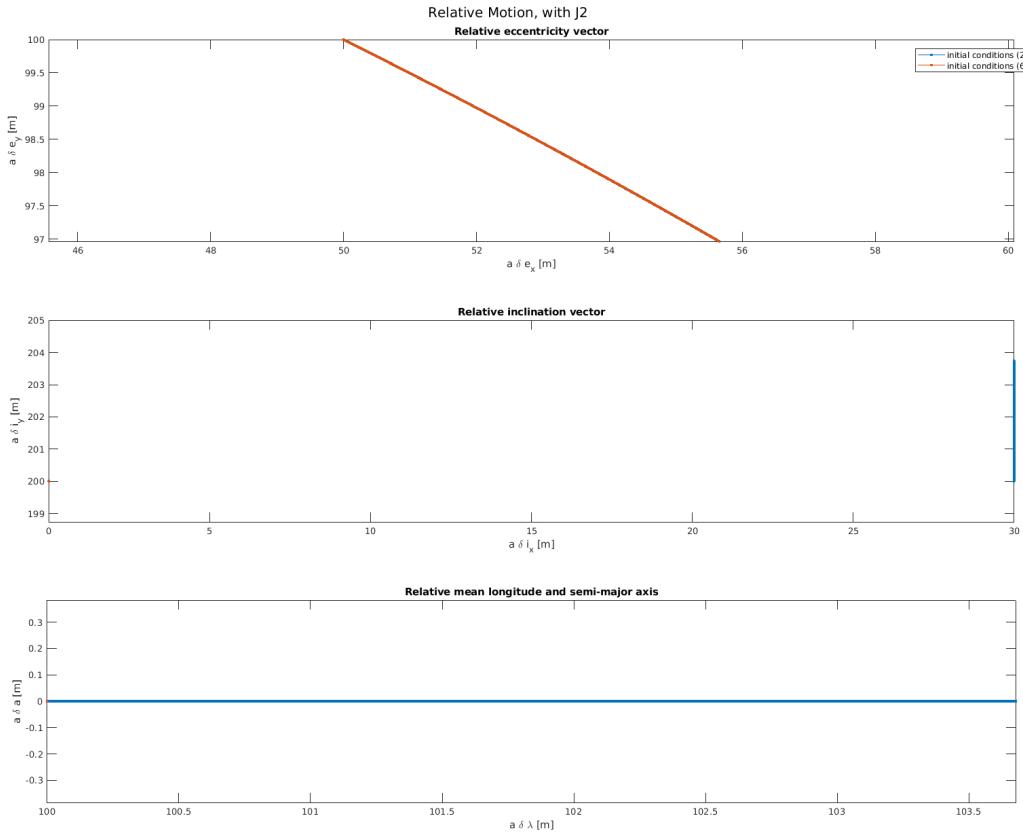


Figure 52: Orbital elements calculated using the STM with initial conditions from part 2 and part 6 ( $i_x = 0$ ).

It may be hard to see, so to be extra clear, there is an orange dot on the relative inclination vector plot at (0, 200), as well as on the relative mean longitude vs. semi-major axis plot at (100, 0).

Yes, these results are consistent with our numerical integration simulation.

## 5 Problem Set 5: Implement impulsive control law

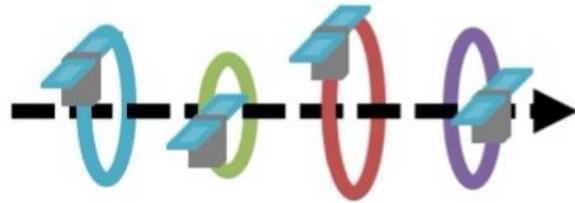
### 5.1 Problem 1. What are the control orbits?

#### 5.1.1 Significant operational modes.

1. In-train mode (deployment configuration).
2. Passive safety ellipse mode (tech demo mode).



Phase 1: In-Train String of Pearls  
Credits: NASA



Phase 2: Passive Safety Ellipses  
Credits: NASA

Figure 53: A diagram of the two orbital modes.

#### 5.1.2 Mode definitions .

1. In-train mode: All 4 satellites travel “in-train” like a “string-of-pears” [1]. This occurs right after deployment, and is the default mode when not demonstrating new software for tech-demo. Each satellite will solely have a non-zero  $\delta\lambda$ , all other relative orbital elements will be zero. Theoretically, no formation keeping will be necessary, but adjusting for small errors in orbit insertion or higher order perturbations may be necessary.  $\delta\lambda$  will range from about 10 km to 200 km for each spacecraft.
2. Safety ellipse mode: maintain constant  $\delta\lambda$ , and relative e-i vector separation, for the purposes of technology demonstration of special software. Since J2 will cause a precession of the relative eccentricity vector, we will need all deputies to perform formation keeping delta-v burns to keep the relative eccentricity vector within a dead-band of parallel to the relative inclination vector. Furthermore, J2 will cause a secular vertical drift (along  $\delta i_y$ ) of the inclination vector if  $\delta i_x \neq 0$ . Therefore we choose to force  $\delta i_x = 0$  for all deputies. The final constraint on relative motion is that each satellite must be able to be picked up by its neighbors’ star trackers, so the passive safety ellipses can’t be so large that the spacecraft go out of the field-of-view of the star trackers, and also can’t be so small that the relative motion is indiscernible.

### 5.1.3 Controls while operating in mode.

1. Controls for formation keeping in in-train mode: Under subject of higher order perturbation, tangential burns may be necessary to keep  $\delta\lambda$  within the allowable range of about  $10 - 250\text{km}$ .
2. Controls for formation keeping in passive safety ellipse mode: As mentioned in the part above, J2 will cause a precession of the relative eccentricity vector, so we will need to perform burns to keep our relative eccentricity vector parallel with our relative inclination vector, within some pre-defined dead band. As for the dead band, we should choose one that keeps our  $\delta\lambda$  within some allowable bounds (i.e.  $\pm 5\text{km}$ ).

Further J2 will cause a drift in  $\delta i_y$  if  $\delta i_x$  is non-zero, so we will force  $\delta i_x$  to be zero. If higher order perturbations cause  $\delta i_x$  to drift, we will need to perform burns to return the relative inclination vector back to its nominal value. Since the  $\delta i_y$  drift is non-secular, we will likely have a quite strict tolerance on keeping  $\delta i_x = 0$ .

### 5.1.4 Controls for switching between modes

1. Controls for in-train mode to safety ellipse mode: 1 (or 2) out-of-plane burn(s) to increase  $\delta i_y$  to some finite value. And 2 (or 3) in-plane burns to set  $\delta \vec{e}$  to some finite value parallel to  $\delta \vec{i}$ . No major requirements on reconfiguration time, but we would like to be able to do it over the course of a few orbits at most (so about half a day or so). Another main constraint is that we want to keep  $\delta\lambda$  within the  $10 - 250\text{km}$  bound throughout the whole process.
2. Controls for safety ellipse mode to in-train mode: Simply the reverse of above.

### 5.1.5 What actuators are you considering?

For the sake of the p-set, we are assuming that Starling has impulsive thrusters where  $\Delta v$  impulses can be modeled as instantaneous.

### 5.1.6 Given the control requirement, what absolute or relative orbit dynamics models are needed?

We also plan to propagate the absolute dynamics in Cartesian components of the chief satellite in order to maintain knowledge of the absolute orbital elements of the chief. This model will be fully non-linear and incorporate J2.

We also plan to use the linear STM for Quasi Nonsingular ROE with J2 to model our relative orbital dynamics of the deputy and plan control requirements.

We will only consider the two-body problem, since the effect of any third body on our system is negligible.

We would also like to propagate a non-linear, non-circular, with J2 relative model of the satellite to check for compounding errors with the simpler linearized relative model.

## 5.2 Problem 2. Impulsive Control Law.

### 5.2.1 Implementation and Justification

- (a) For ground truth, we chose to simulate the fundamental orbital differential equation for absolute orbits in position and velocity in the ECI frame, propagated using MATLAB's ode113. However, we ran into numerical precision errors compounding and leading to a drift in our  $\delta\lambda$ . This was seen in the previous section (PSet 4). We actually found that our STM with J2 propagator for relative motion was more accurate for our use case than the non-linear ECI propagator. Therefore, we used the STM for the ground truth of the deputy (and used our intuition to ensure our results are reasonable). If we include higher order perturbation s(drag, third-body, etc.) in the future, we will likely return to propagating the FODE in position and velocity in ECI as our ground truth.
- (b) For our dynamics and maneuver state representation, we relied on the absolute Keplerian elements and ECI vectors for the chief generated with a non-linear simulator with J2 in ECI. For the position of our deputy, we used relative quasi-non-singular orbital elements which we propagated with our STM which includes J2 perturbations.
- (c) We implemented our burns as instantaneous impulsive delta-v injections. The simulation was “paused” at each burns’ timestep, the velocity of the deputy was instantaneously changed, and then the simulation was “resumed.” This is far from realistic, but a reasonable estimation for our purposes.
- (d) We did not explicitly include uncertainties, however, we do implement active formation keeping control. Currently, this formation keeping solely controls the dead-band for  $\delta e_x$ , but it is written in such a way that it can easily be adapted to correct for drifts in any of the relative orbital elements. A few code changes could allow us to use formation keeping to correct for any uncertainties that propagate errors, at the cost of delta-v.

### 5.2.2 Impulsive control laws for formation keeping and reconfiguration.

#### 5.2.3 Formation reconfiguration

As in subsequent sections, the orbital elements of the chief are:

$$\begin{bmatrix} a \\ e \\ i \\ \Omega \\ \omega \\ \nu \end{bmatrix} = \begin{bmatrix} 7000 \text{ km} \\ 0.001 \\ 98^\circ \\ 0 \\ 90^\circ \\ 0 \end{bmatrix}.$$

We began in the in-train mode where the only non-zero relative orbital element is mean longitude.

$$\begin{bmatrix} \delta_a \\ \delta_\lambda \\ \delta e_x \\ \delta e_y \\ \delta i_x \\ \delta i_y \end{bmatrix} = \begin{bmatrix} 0 \\ 100 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \text{ km.}$$

After our mode change to the passive safety ellipse mode, our relative orbital elements are:

$$\begin{bmatrix} \delta_a \\ \delta_\lambda \\ \delta e_x \\ \delta e_y \\ \delta i_x \\ \delta i_y \end{bmatrix} = \begin{bmatrix} 0 \\ 100 \\ 0 \\ 30 \\ 0 \\ 30 \end{bmatrix} \text{ km.}$$

Our planned duration of the mode changing maneuver is one orbit.

#### 5.2.4 Least squares Reconfiguration Maneuver

Our first step is to calculate a maneuver to go from an in-train mode to passive safety ellipse mode where the  $\delta\lambda$  stays constant while our  $\delta e_y$  and  $\delta i_y$  both increase by 30 km. To do this, we rely on a least squared solution with our delta velocities occurring at mean arguments of latitude equal to  $[0, \pi/2, \pi]$ .

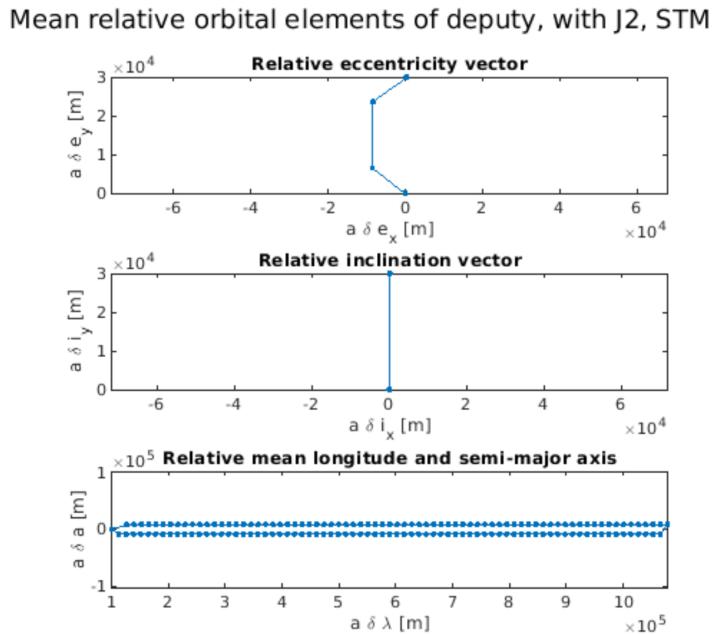


Figure 54: A diagram of the relative orbital elements of the deputy spacecraft as we maneuver from in-train mode to passive safety ellipse mode.

The delta-vs calculated by our least squares solver are:

$V_R$	-0.0069	-2.1869e-05	0.00694327734352689	km/s
$V_T$	-0.0046	0.0092	-0.0046	
$V_N$	-6.723e-05	0.0323	-6.72396e-05	

We can see that the first and last burn are primarily in-plane while the second burn is primarily out-of-plane.

### 5.2.5 Formation Keeping

We implemented an allowable “dead-band” around the relative eccentricity vector, almost exactly as outlined in slide 11 of Lecture 9. Because we are in a sun-synchronous orbit (retrograde), we know our relative eccentricity vector will precess clockwise. Starling’s relative eccentricity requirements are not published, so we chose  $a\delta e_{x,max} = 10km$  for the purposes of visualizing the effect.

Once, we are in passive safety ellipse mode, we check to see if  $\delta e_x$  has gone beyond our maximum value. If it has, we solve the least squares maneuver solution to rotate our relative eccentricity vector to  $\delta e_x = -\delta e_{x,max}$ . Because we know the closed form solution for near-circular orbits ahead of time, we choose to perform two burns half an orbit apart. We allow the least squares solver an extra burn a full orbit later in order to better zero  $\delta a$ .

Due to perturbations, and imperfect maneuvers, we also implement another dead band around relative mean longitude. We allow  $\delta\lambda$  to drift up to 50km within our dead band.

If either dead band is triggered, we choose to perform a maneuver that will reset both dead bands. Note that for this dead band, we do aim to burn such that we set  $\delta\lambda$  to the desired quantity, rather than pushing it to the other side of the dead band. We do this because, during this maneuver, we also perform the chose to zero out as much  $\delta a$  as possible, so we do not want  $\delta\lambda$  to drift further.

Our delta velocities for this formation keeping as computed by our least squares solver are:

$V_R$	-0.1334	-0.26808	-0.1334	km/s
$V_T$	0.002754	-0.00551	0.002754	
$V_N$	0.000151	0.000211	0.000151	

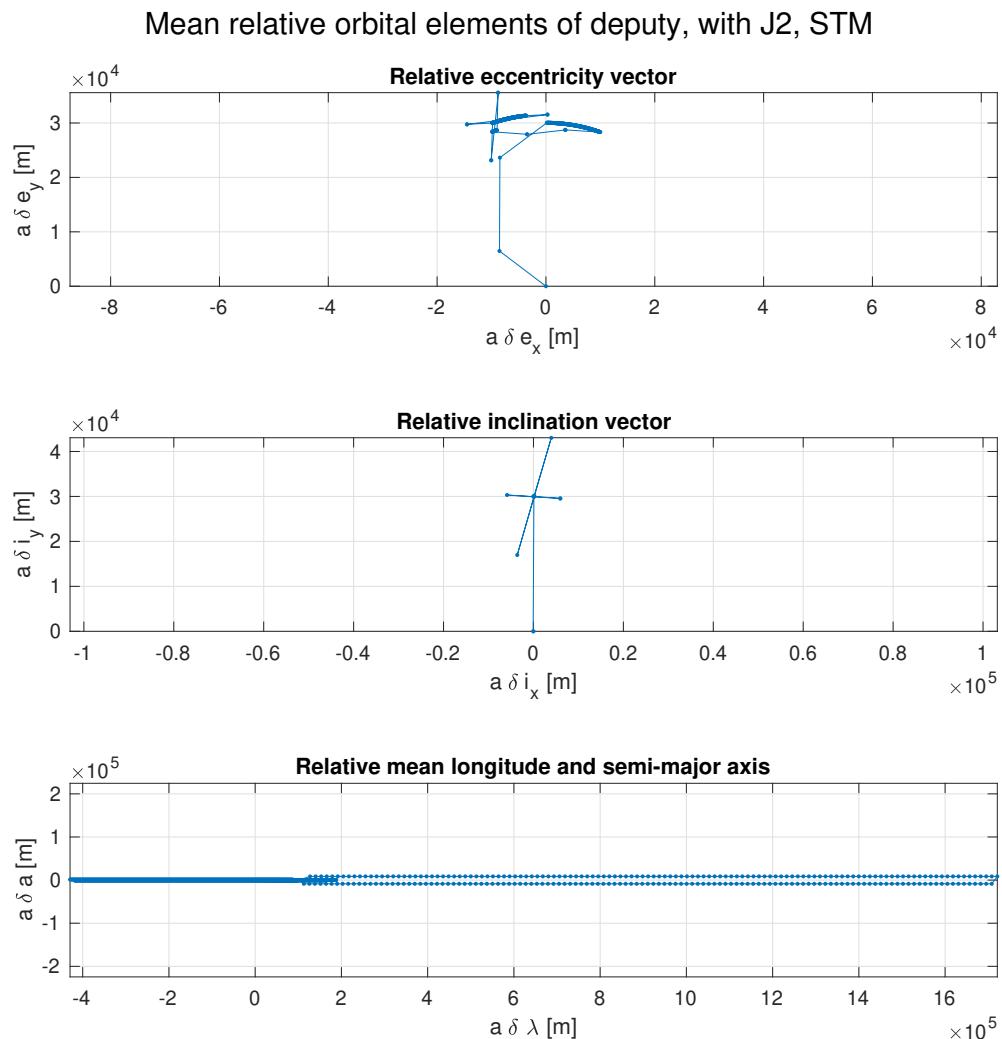


Figure 55: A diagram of the relative orbital elements of the deputy spacecraft as we maneuver from in-train mode to passive safety ellipse mode, and then perform formation keeping to keep  $\delta e_x \approx 0 \pm \delta e_{x,max}$  over the course of 200 orbits.

### 5.2.6 RTN visualization

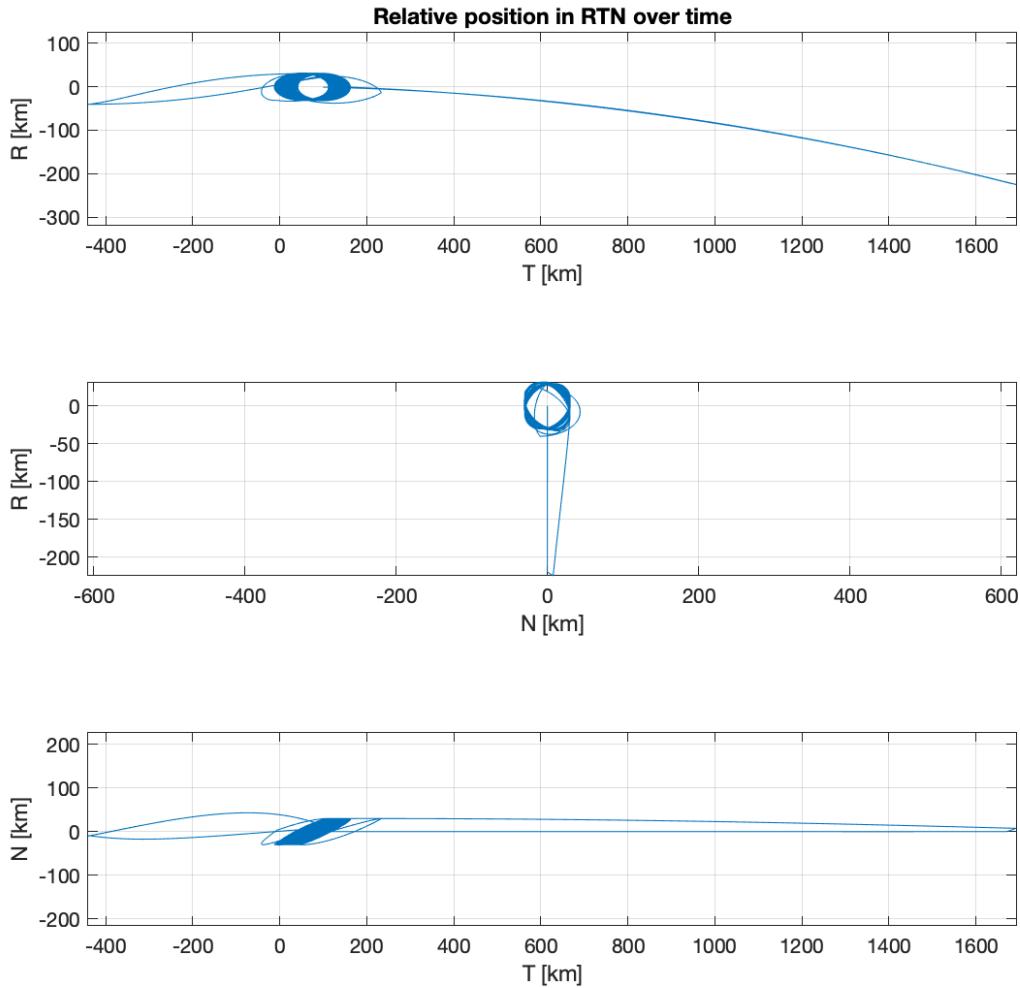


Figure 56: A diagram of our position in RTN over 200 orbits. There is initially a large change in the along-track, but this is solely due to the reconfiguration maneuver. During the formation keeping maneuvers, the along-track drift is kept in check, within a dead band.

### 5.2.7 Delta-V Closed Form Solutions and Lower Bound and Comparisons

For our formation keeping, the closed form solution is given by D'Amico as:

$$\begin{aligned}\delta v_{T_1} &= \frac{na}{4} \|\Delta \delta \mathbf{e}\| \\ \delta v_{T_2} &= -\frac{na}{4} \|\Delta \delta \mathbf{e}\| \\ u_{M_1} &= \arctan_2\left(\frac{\Delta \delta e_y}{\Delta \delta e_x}\right).\end{aligned}$$

This gives us:

$$\begin{aligned}\delta v_{T_1} &= \frac{na}{4} \left\| \begin{bmatrix} -2 * 0.00142857 \\ 0 \end{bmatrix} \right\| = 0.005390035 km/s \approx 5.4 m/s \\ \delta v_{T_2} &= -\frac{na}{4} \left\| \begin{bmatrix} -2 * 0.00142857 \\ 0 \end{bmatrix} \right\| = -0.005390035 km/s \approx -5.4 m/s \\ u_{M_1} &= \arctan_2\left(\frac{0}{-2 * 0.00142857}\right) = \pi.\end{aligned}$$

And,  $u_{M_2}$  occurs half an orbit after  $u_{M_1}$ .

We also perform a burn to zero out any relative semi-major axis. We can use

$$\begin{aligned}a\delta a &= 2 \frac{\delta v_T}{n} \\ \delta v_T &= \frac{na\delta a}{2}.\end{aligned}$$

We notice that in the worse case our  $a\delta a$  is off by no more than about  $10m$  (from zooming in on 60), so we can estimate the burn to zero out  $\delta a$  will cost at most  $\approx 0.5 cm/s$ . This is negligible compared to the rest of the burns.

So, our lower bound for formation keeping should be about  $11m/s$  for each maneuver necessary. And, based off how long it takes our relative eccentricity vector to rotate, we see that we need to perform one of these formation keeping maneuvers about every 180 orbits to stay within our bounds.

Comparing this to our least squares solved delta-v's actually implemented in our code, we see that our solver optimally keeps  $\Delta v_N \approx 0$ , but non-optimally chooses to perform significant delta-v in the radial direction, rather than the along-track direction. Here, we spend a total of about  $21.3 m/s$  of delta-v (this is computed by taking the norm of burn at the same argument of latitude, and then the total sum of each burn), rather than the optimal  $11 m/s$ . So, we are spending a little more than twice as much delta-v as we optimally could be.

**For our reconfiguration maneuver**, we perform the same analysis for in-plane, and also perform an out-of-plane analysis as follows:

$$\begin{aligned}\delta v_{N_1} &= na \|\Delta \delta \mathbf{i}\| \\ u_{M_1} &= \arctan_2\left(\frac{\Delta \delta i_y}{\Delta \delta i_x}\right).\end{aligned}$$

So, the whole reconfiguration maneuver gives us the following for in-plane:

$$\delta v_{T_1} = \frac{na}{4} \left\| \begin{bmatrix} 0.0042857142 \\ 0 \end{bmatrix} \right\| = 0.0080850526 km/s \approx 8.1 m/s$$

$$\delta v_{T_2} = -\frac{na}{4} \left\| \begin{bmatrix} 0 \\ 0.0042857142 \end{bmatrix} \right\| = 0.0080850526 km/s \approx -8.1 m/s$$

$$u_{M_1} = \arctan_2\left(\frac{0.0042857142}{0}\right) = \frac{\pi}{2},$$

and the following for out-of-plane:

$$\delta v_{N_1} = na \|0.0042857142\| = 0.03234021 km/s \approx 32.3 m/s$$

$$u_{M_1} = \arctan_2\left(\frac{0.0042857142}{0}\right) = \frac{\pi}{2}.$$

So, our lower bound on the full reconfiguration maneuver (in-train to passive safety ellipse) is about  $41.4 m/s$ , which we will round up to  $42 m/s$ . This is computed by taking the norm of burn at the same argument of latitude, and then the total sum of each burn.

Notice: Optimally, we would replace burns in radial direction with less expensive burns in the tangential direction. Also, the out-of-plane maneuver is twice as expensive as the in-plane maneuver.

Comparing this to our least squares solved delta-v's actually implemented in our code, we see that again we mostly decompose the cross-track maneuver, but we still see that we favor radial and along-track maneuvers instead of pure along-track maneuvers. Here, we spend a total of about  $50 m/s$ , which is about 20% larger than optimal. (Honestly, not as bad as we expected.)

### 5.2.8 Control systems over time

1. Control tracking error, maneuver scheduling, and delta-v over time

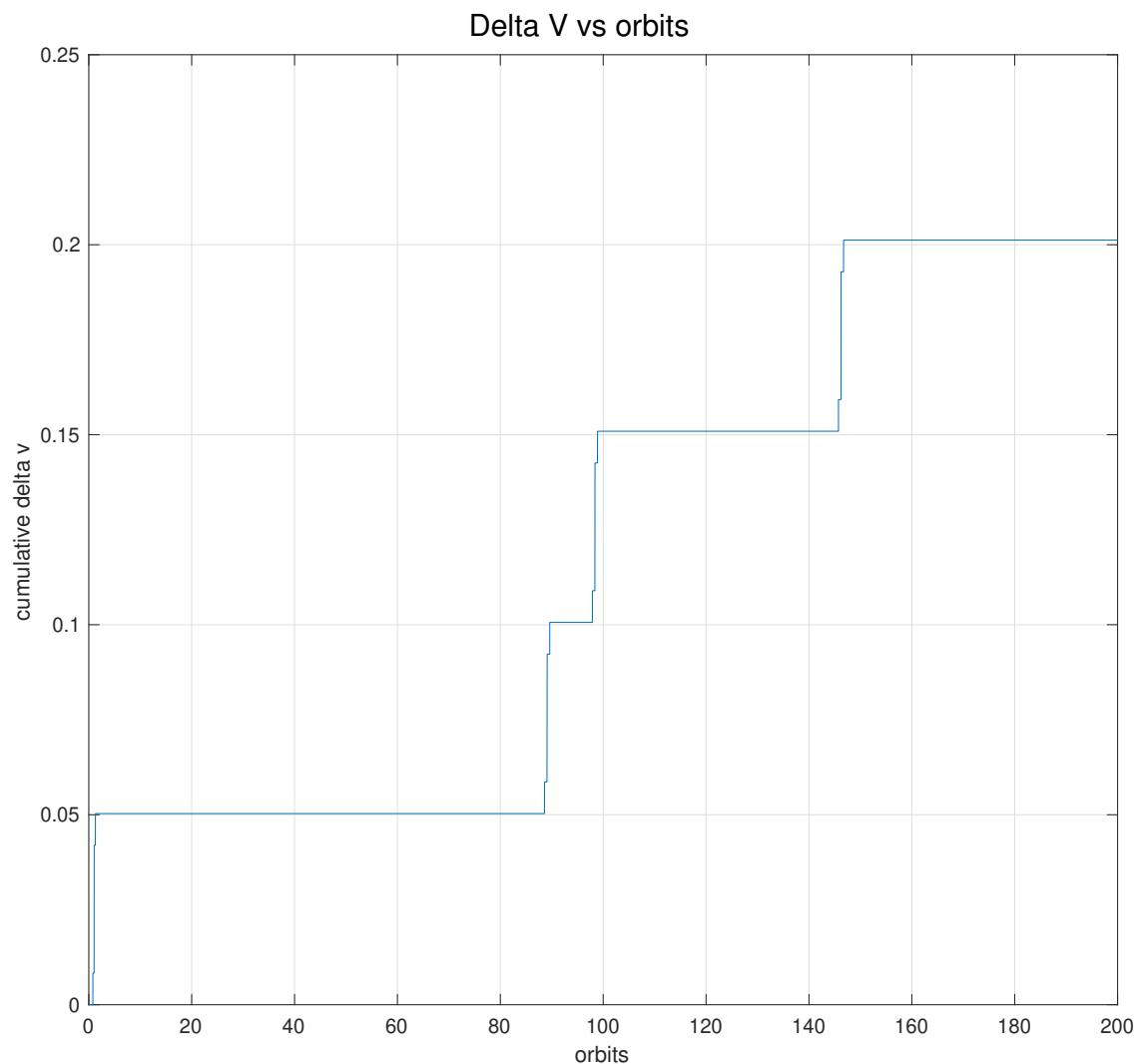


Figure 57: Our cumulative delta v for both the initial maneuver from in-train to a passive safety ellipse and subsequent station keeping maneuvers over the course of 200 orbits.

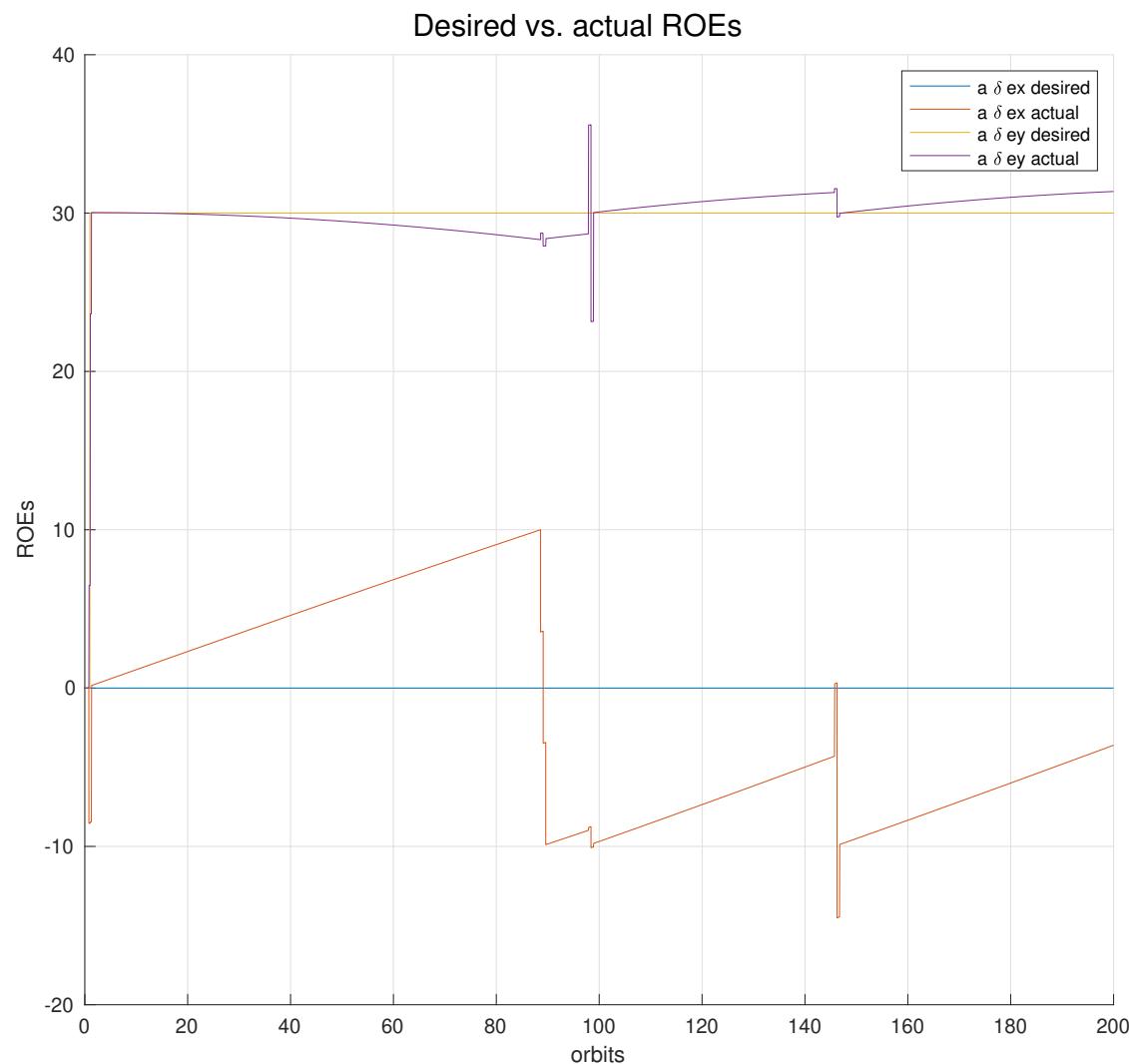


Figure 58: Relative eccentricity vector tracking over time. We can see that our relative eccentricity vector begins to drift due to J2 effects and is reset by our impulsive maneuvers.

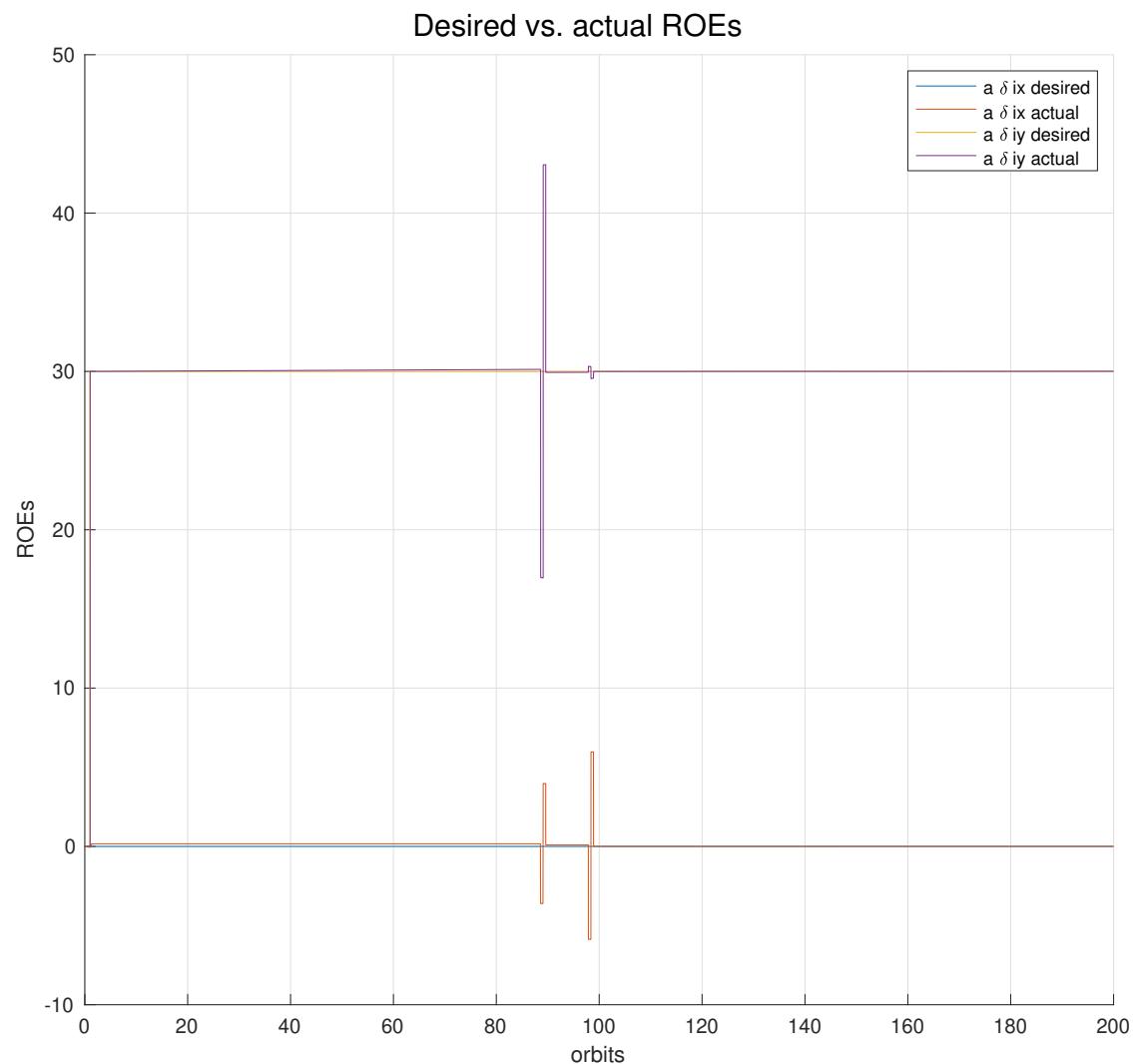


Figure 59: Relative inclination vector tracking over time. It remains largely unaffected by the J2 perturbation.

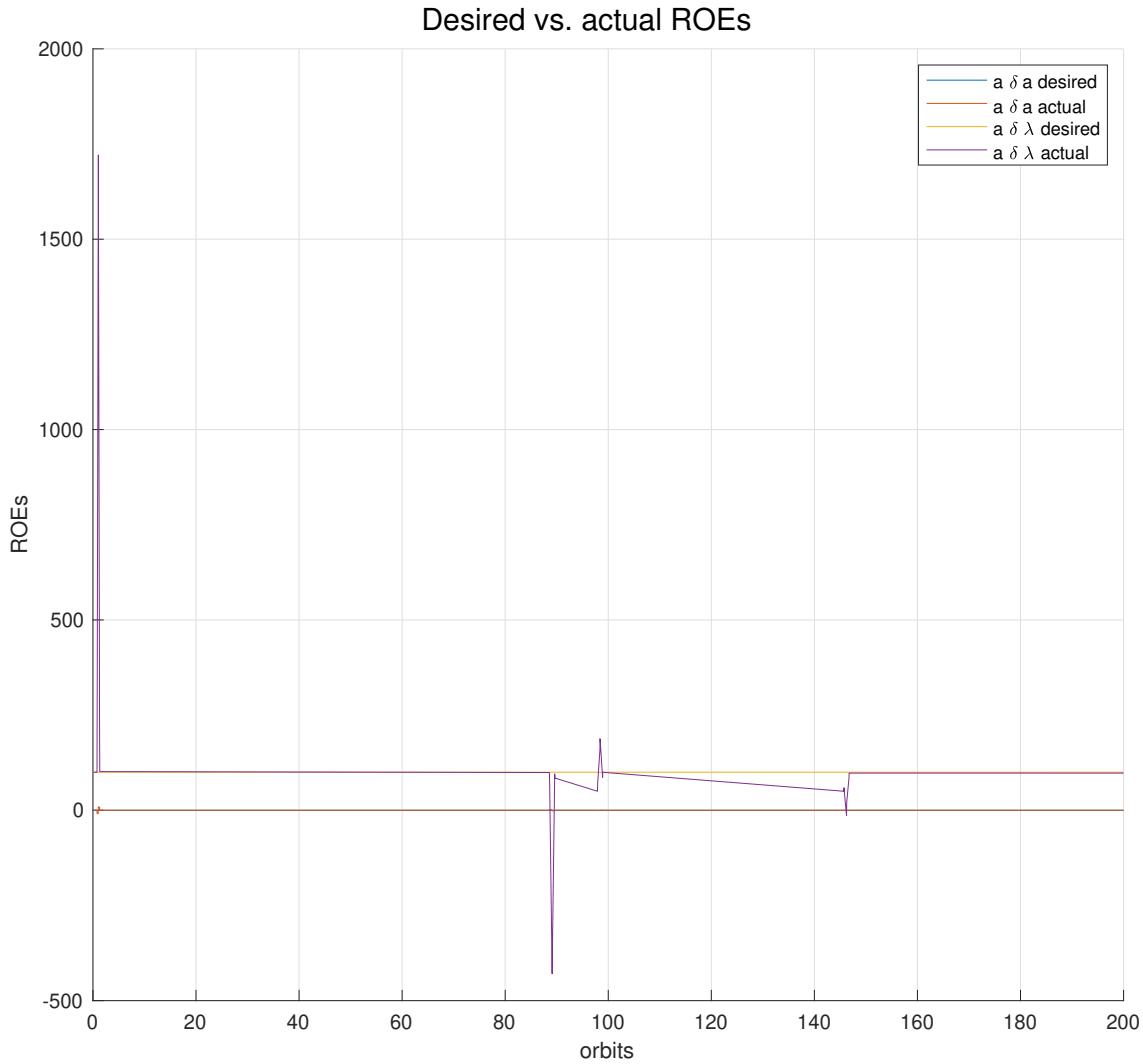


Figure 60: Relative semi-major axis and relative mean longitude tracking over time. We see a poor maneuver cause a non-zero  $\delta a$  causing a drift in  $\delta \lambda$ . However, this gets fixed in the following formation keeping maneuver.

In the above graph, we can see that some formation keeping maneuvers are poorly executed and result in a non-zero  $\delta a$  and thus a drifting  $\delta \lambda$ , but they get corrected for in the following formation keeping maneuver. We suspect this is due to poor solving of the least squares solution. It can be better mitigated with more frequent formation keeping, but is not necessary to meet the mission requirements.

## 2. Strengths and weaknesses of our implementation:

The major strengths of our implementation are it's speed and adaptability. It is fast because we are using the linearized STM with J2 to propagate mean relative orbital elements, and pre-compute the chief's trajectory in absolute ECI position and velocity. This is surprisingly accurate, and allows

us to easily perform deputy maneuvers in mean element space. Our implementation is adaptable because our least squares solver can solve for any maneuver as long as we provide it a sufficient number of burns (2 for in-plane, 1 for out-of-plane). If we have doubts about where burns should be placed, we can add more burns at multiple locations (3-4 for in-plane, 2-3 for out-of-plane), and our least-squares solver should return a feasible solution.

The major weakness of our implementation is that it provides relatively sub-optimal solutions for delta-v burns during our maneuvers. We waste quite a bit of delta-v relative to the theoretical lower bound. Perhaps we could mitigate this by changing the cost function to penalize our solver for delta-v in the radial direction. The other weakness of our implementation is it's simplicity. We aren't modeling higher order perturbations, such as drag, third-body, etc., including uncertainties, or applying delta-v's over a realistic time intervals.

3. Performance against expectation: See section 5.2.6 Delta-V Closed Form Solutions and Lower Bound and Comparisons. In summary, our reconfiguration maneuver costs about 30% more than optimal, and our formation keeping costs about twice as much as the optimal solution.
4. Outstanding problems: One outstanding problem is that our least squares solver can perform so poorly that we end up with a significant  $\delta a$ , which leads to drift in  $\delta \lambda$ . This gets mitigated in following formation keeping maneuvers, and can be further mitigated with more frequent burns, but remains a core issue with the least squares approach.

## 6 Problem Set 6: Implement continuous control law

### 6.1 Problem 1: Continuous Control Law

We implement a continuous Lyapunov controller to control both reconfiguration and formation keeping.

#### 6.1.1 Implementation and justification

1. Define formation reconfiguration

As in subsequent sections, the orbital elements of the chief are:

$$\begin{bmatrix} a \\ e \\ i \\ \Omega \\ \omega \\ \nu \end{bmatrix} = \begin{bmatrix} 7000 \text{ km} \\ 0.001 \\ 98^\circ \\ 0 \\ 90^\circ \\ -15^\circ \end{bmatrix}.$$

We began in the in-train mode where the only non-zero relative orbital element is mean longitude.

$$\begin{bmatrix} \delta_a \\ \delta_\lambda \\ \delta e_x \\ \delta e_y \\ \delta i_x \\ \delta i_y \end{bmatrix} = \begin{bmatrix} 0 \\ 100 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \text{ km.}$$

After our mode change to the passive safety ellipse mode, our relative orbital elements are:

$$\begin{bmatrix} \delta_a \\ \delta_\lambda \\ \delta e_x \\ \delta e_y \\ \delta i_x \\ \delta i_y \end{bmatrix} = \begin{bmatrix} 0 \\ 100 \\ 0 \\ 30 \\ 0 \\ 30 \end{bmatrix} \text{ km.}$$

We then monitor relative orbit elements and provide delta-v impulses as J2 perturbations move us away from our ROE target.

2. Dynamics model for ground truth simulation.

Similar to the previous problem set, we model the ground truth of the chief spacecraft by solving the non-linear differential equations that factors in J2 perturbations. For the deputy spacecraft, we continue to use the STM propagator as the ground truth because, we noted in the previous problem set, the STM propagator appears to suffer from fewer or smaller numerical issues relative to the non-linear model.

3. Dynamics model for state representation of controller/maneuver planner.

Also similar to the previous problem set, we continue to primarily work in the orbital elements of the chief (both mean and osculating) and the relative orbit elements of the deputy.

4. Apply Lyapunov control theory with or without constraints.

We choose to apply Lyapunov control theory without constraints because of its simplicity, computational efficiency, and because our mission does not require tight constraints on the reconfiguration maneuver. This controller provides local stability guarantees, but not global.

We use the continuous Lyapunov controller as described in Lecture 10. We use the reduced linear model for circular orbits where  $\delta\lambda$  is not directly controlled, but is indirectly controlled through  $\delta a$ . We control based off of the control tracking error  $\Delta\delta oe = \delta oe - \delta oe_{desired}$ . Our Lyapunov function is then defined as  $V = \frac{1}{2}\Delta\delta oe^T\Delta\delta oe$ , and thus our control law is defined as

$$u = -B^*(A\delta oe + P\Delta\delta oe),$$

where

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ \frac{7}{2}\kappa e_{yf}Q\tau & \cos(\dot{\omega}) - 4\kappa e_{xi}e_{yf}GQ\tau & \sin(\dot{\omega}) - 4\kappa e_{yi}e_{yf}GQ\tau & 5\kappa e_{yf}S\tau & 0 \\ -\frac{7}{2}\kappa e_{xf}Q\tau & \sin(\dot{\omega}) + 4\kappa e_{xi}e_{xf}GQ\tau & \sin(\dot{\omega}) + 4\kappa e_{yi}e_{xf}GQ\tau & 5\kappa e_{xf}S\tau & 0 \\ 0 & 0 & 0 & 1 & 0 \\ \frac{7}{2}\kappa S\tau & -4\kappa e_{xi}GS\tau & -4\kappa e_{yi}GS\tau & 2\kappa T\tau & 1 \end{bmatrix},$$

$$B = \frac{1}{na} \begin{bmatrix} 0 & 2 & 0 \\ \sin(u) & 2\cos(u) & 0 \\ -\cos(u) & 2\sin(u) & 0 \\ 0 & 0 & \cos(u) \\ 0 & 0 & \sin(u) \end{bmatrix}.$$

And,

$$P = \frac{1}{k} \begin{bmatrix} \cos(J)^N & 0 & 0 & 0 & 0 \\ 0 & \cos(J)^N & 0 & 0 & 0 \\ 0 & 0 & \cos(J)^N & 0 & 0 \\ 0 & 0 & 0 & \cos(H)^N & 0 \\ 0 & 0 & 0 & 0 & \cos(H)^N \end{bmatrix},$$

where we use  $k = 1$ ,  $N = 10$ , and  $J$  and  $H$  are the in-plane and out-of-plane phase differences between the optimal orbit location of the burn and current location (as defined in Lecture 10), respectively.

5. Compare with impulsive control for delta-v lower bound.

In the previous problem set, we found that the theoretical lower bound for impulsive maneuvers is about  $42m/s$  for the full configuration maneuver (in-train to passive safety ellipse). Based on our Figure 62, we see that applying a continuous maneuver results in a delta-v of about  $52m/s$  or about 23% more than the optimal impulsive maneuver. Relative to our actual impulsive maneuver applied in the previous problem set (delta-v =  $50m/s$ ), the delta-v of in the continuous case is only 4 percent worse.

6. Actuator implementation. Continuous delta-vs could be achieved on the satellite with any high specific impulse, low thrust engine such as an ion-thruster. We have variables that allow us to set a lower and upper bound of thruster actuation, which better reflects the reality of thrusters. Currently, they have been chosen somewhat arbitrarily, based off our rough estimates of the problem, but they can be updated in the future to better reflect a particular thruster.

7. Inclusion of uncertainty. Similar to the previous problem set, in order to keep our models simple and easy to work with we are not including uncertainties in the state nor the actuations. Please see PSet 9 for the inclusion of noise and uncertainty in the state inputs to our continuous controller. Unfortunately, we see that noise significantly affects the performance of our controller. Again, please reference PSet 9. Furthermore, we never model uncertainty in the actuations. Perhaps this can be added to future work.

### 6.1.2 Discuss performance and visualize results.

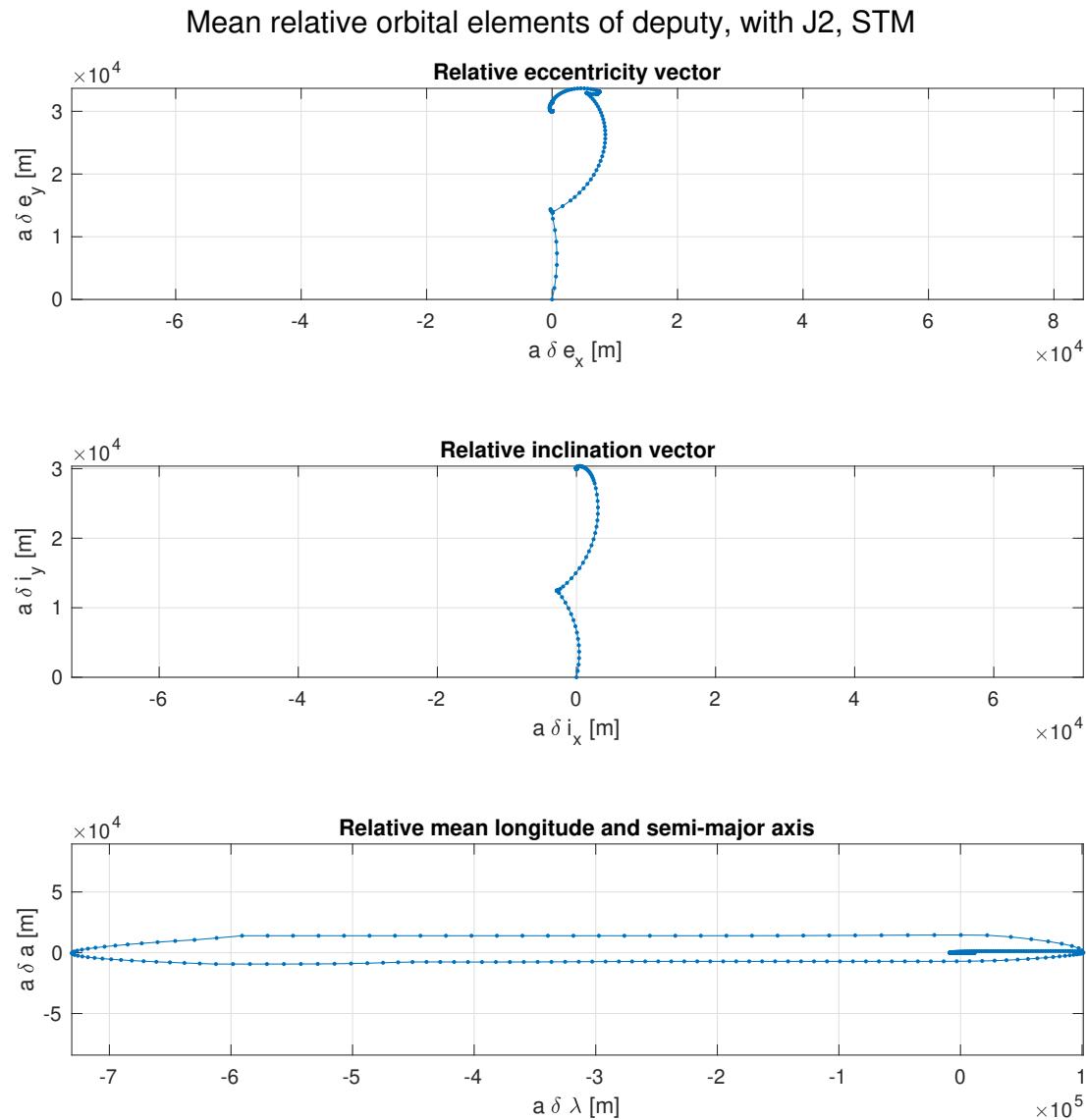


Figure 61: A diagram of the relative orbital elements of the deputy spacecraft as we maneuver from in-train mode to passive safety ellipse mode and then maintain the safety ellipse mode.

### Cumulative delta-v vs orbits passed

Cumulative delta-v vs number of orbits passed

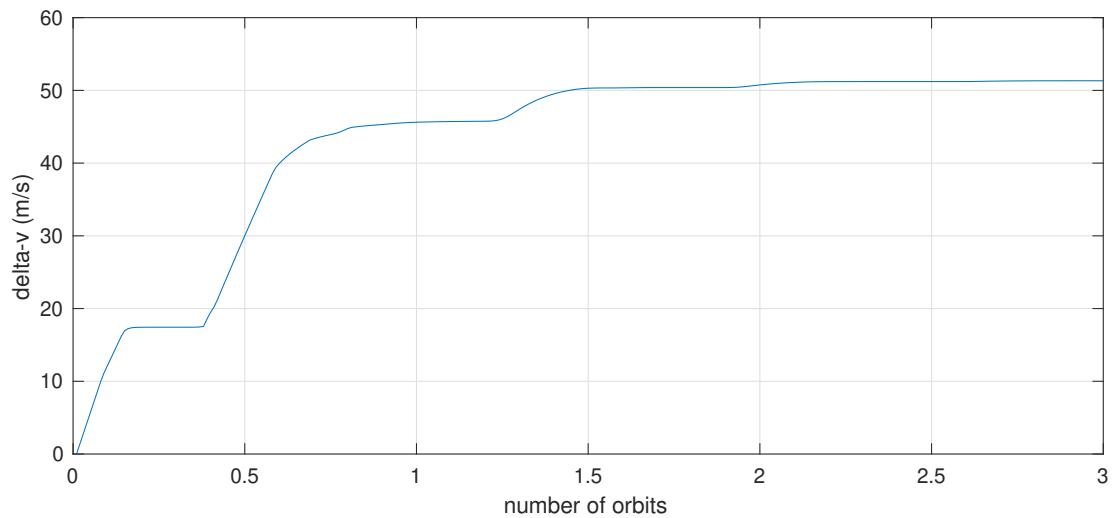
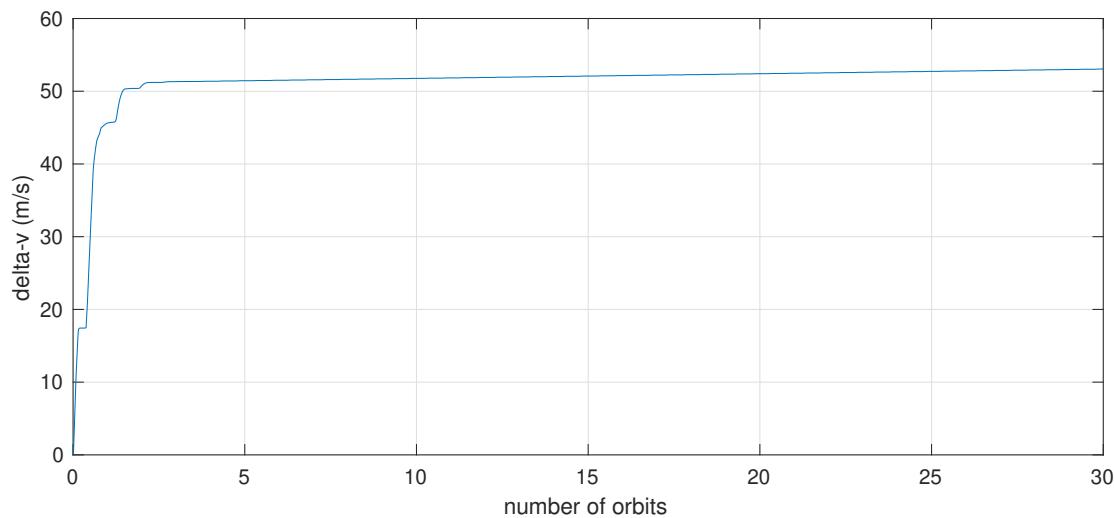


Figure 62: A diagram of the cumulative delta-v applied over time (measured in number of orbits). Note that the bottom subplot is a zoomed-in plot for the first 3 orbits. We can see that the largest delta-v is applied in the first couple orbits to move the deputy into the passive safety ellipse mode and then a consistent amount of delta-v is required per orbit to oppose the J2 effects on the relative eccentricity vector.

### Delta-v vs orbits passed

#### Delta-v components vs number of orbits passed

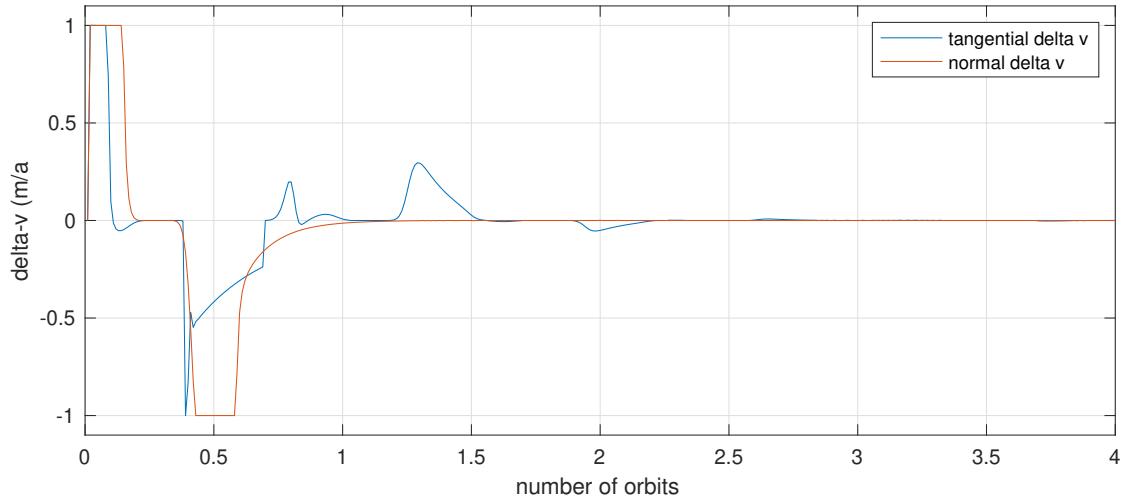
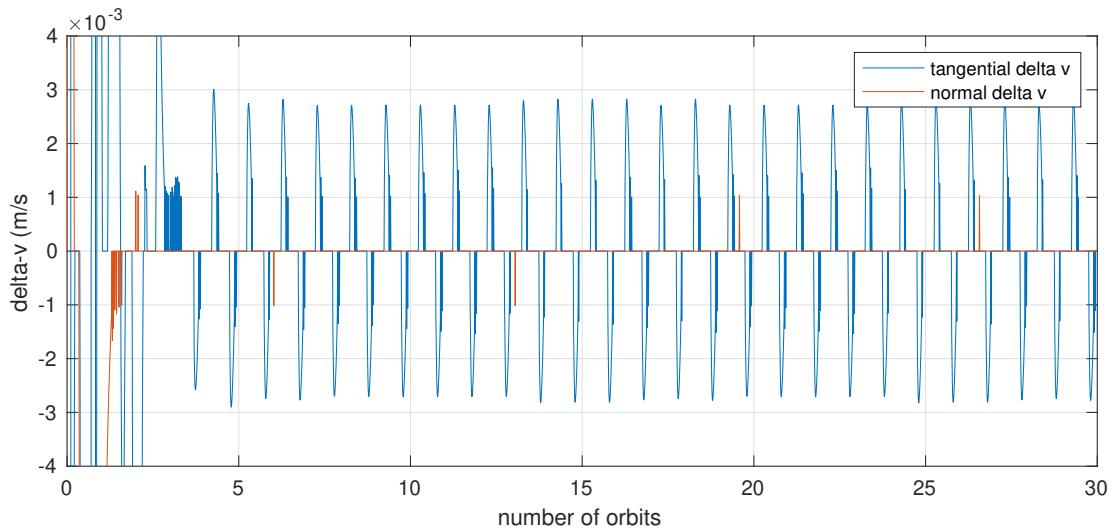


Figure 63: A diagram of the tangential and normal components of delta-v over time. Note that the top subplot is zoomed in on the y-axis in order to show the small bursts of delta-v for formation keeping, while the bottom subplot is zoomed in on the x-axis to show the large bursts of delta-v in the first few orbits. We can see that the majority of the delta-v is applied in the first couple of orbits (both in the tangential and radially components), then a pattern emerges of very small bursts to combat error buildup due to perturbations. Notice that the high frequency pattern of normal burns has a lower frequency than that of tangential burns.

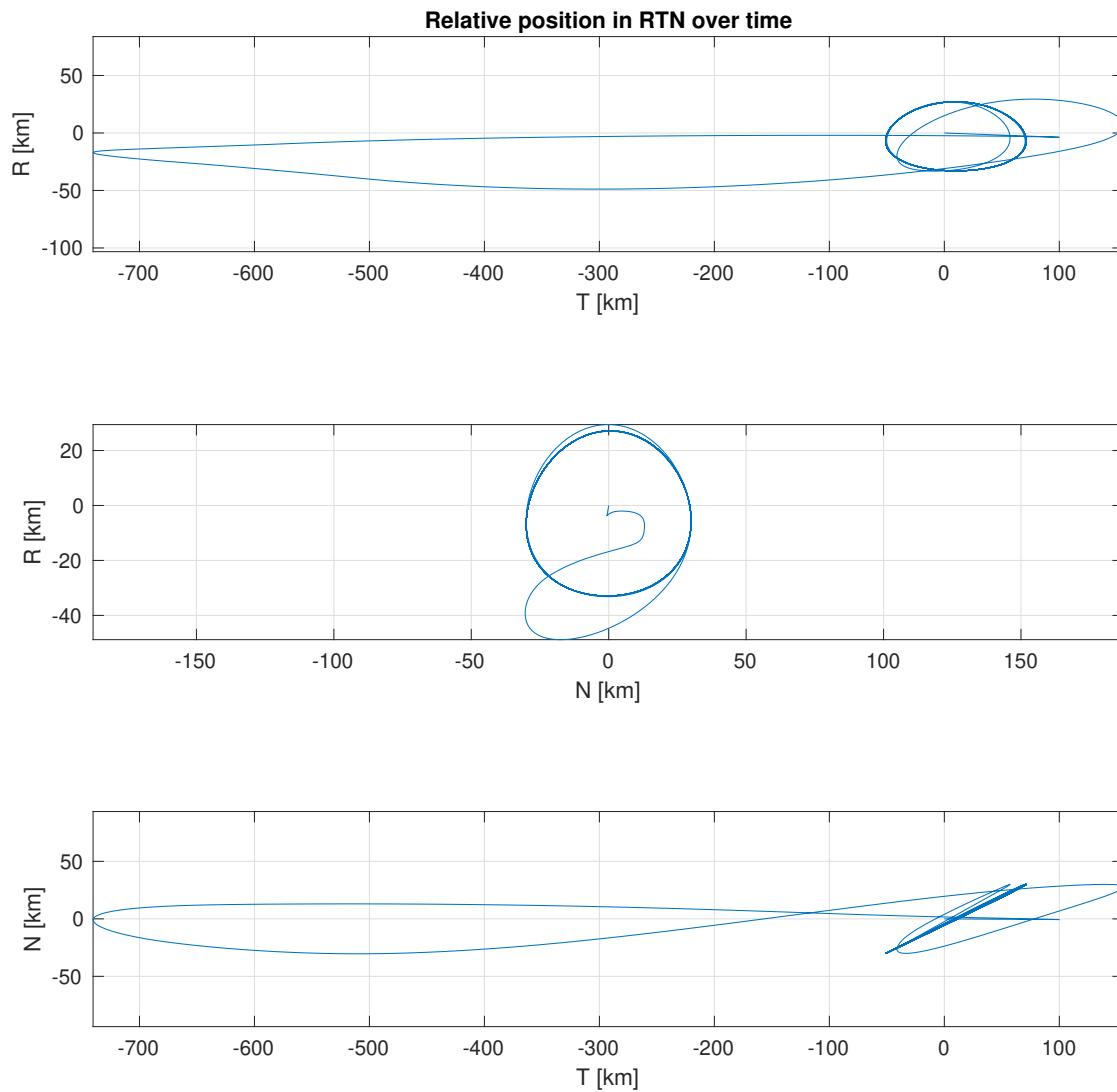


Figure 64: A diagram of the RTN position of the deputy with respect to the chief over time. We see a large along-track separation at the beginning, which is for the reconfiguration maneuver, but the controller very quickly reduces this and the deputy ends up in the desired cyclical motion of the passive safety ellipse.

### ROE error vs time

Change in ROEs vs number of orbits passed

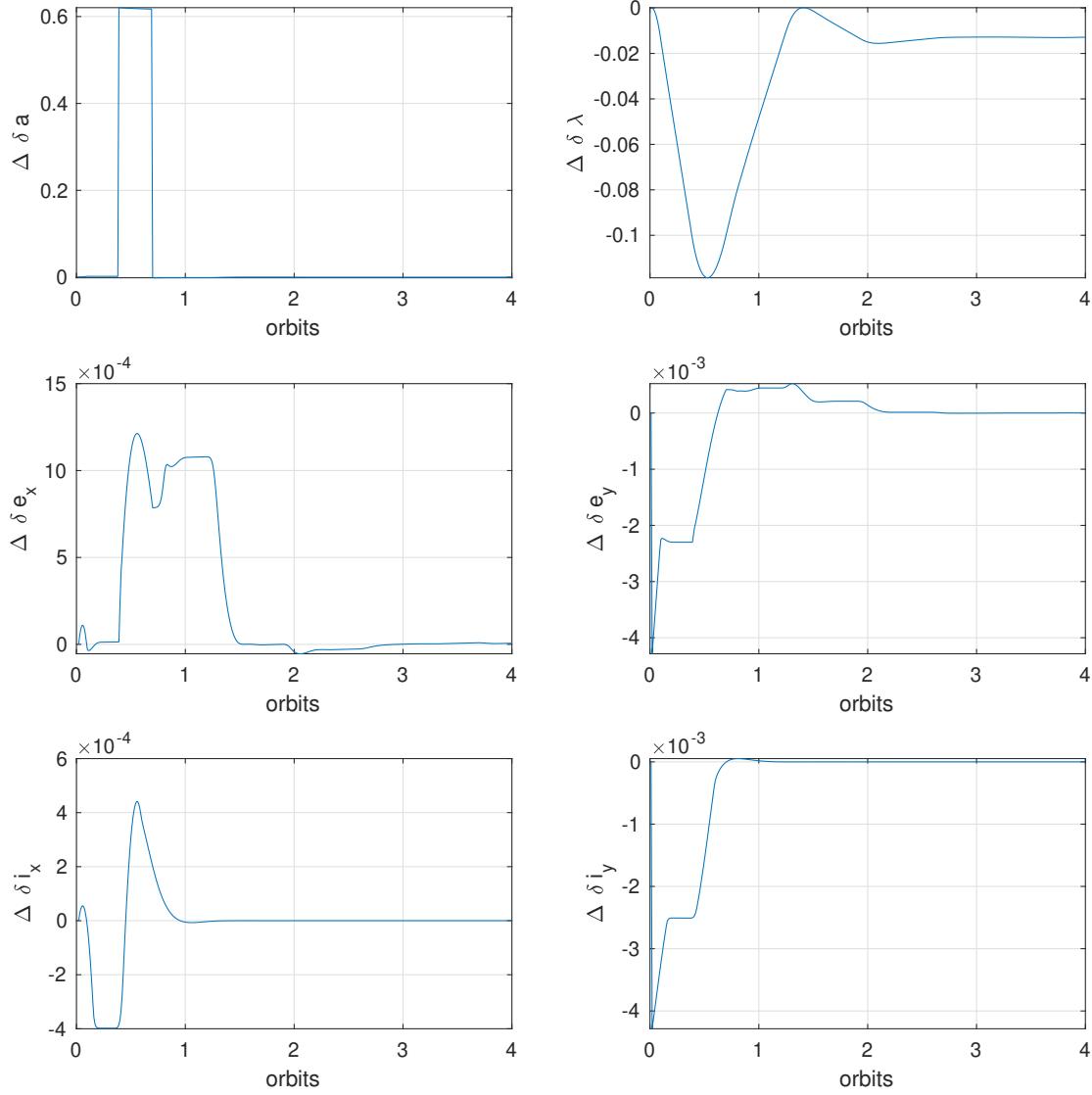


Figure 65: A diagram of the  $\Delta$  ROEs over time. The initial maneuver to go from in-train to passive safety ellipse mode takes only a couple of orbits. Then, after that all errors are close to zero for the formation keeping portion of the controller. Interestingly,  $\Delta\delta\lambda$  appears to maintain at a small constant.

1. Control tracking error, maneuver scheduling, and delta-v over time.

With the continuous Lyapunov model, the largest error in our relative orbit elements is at the first time step and quickly drops to zero for all the relative orbit elements except  $\Delta\delta\lambda$  which is not part of the reduced model we are using. Similarly, the delta-v is applied mostly in the first couple of orbits after which the delta-v grows at a linear rate to maintain the passive safety ellipse mode (figure 62).

2. Compare strengths and weaknesses of your implementation. Relative to the previous problem set, using continuous (rather than impulsive control) allows us to stay much closer to the desired relative orbit elements. Instead of allowing the relative eccentricity to drift before making applying a large delta-v, we are able to apply smaller delta-vs more frequently and thus maintain a tighter bound on our orbital elements. The primary weakness of our model is simplicity. For example, because we are using the reduce model, our delta-vs do not correct for relative mean longitude drifts. In order to not allow the  $\Delta\delta\lambda$  grow to large, we added a conditional statement that modifies our desired  $\delta a$  if  $\Delta\delta\lambda$  is greater than a certain threshold. This bounds the along track drift.
3. Compare performance against expectations (delta v budget, delta-v lower bound, frequency of maneuvers)

The frequency of maneuvers is much higher than in the impulsive case (obviously). We also saw a slightly higher delta-v with continuous burns rather than impulsive burns. This was expected as in the continuous case the burns inevitably do not always take place at the most efficient point in the orbit. Because our orbit is largely circular this effect is not very large.

## 7 Problem Set 7: Finalize relative orbit control laws, design, and start implementing relative navigation system.

### 7.1 Finalize relative orbit control

We plan to use continuous control for both the reconfiguration maneuver and the formation keeping.

Proof of functionality: See the previous two PSets. Both are functional.

Performance: The specifics of our performance characteristics are defined in the previous two PSets. We will talk generally about them here. Both the continuous and impulsive control cases are within 25% of the optimal lower bound. Better solutions could be implemented to get closer, but we value the simplicity of our solution.

Lessons Learned: We found the continuous control to be surprisingly straightforward to implement and, given our near circular orbit, comparable in delta-v to the impulsive maneuvers optimized with least squares. We also learned how important it can be to have the knowledge to remove delta-vs in the radial direction, due to its sub-optimality.

### 7.2 Navigation System Design

We plan to implement a UKF to estimate relative state of the deputy. We will go further into detail as to why we chose this in later (sub)sections. The UKF, unlike the EKF, does not require linearization. Instead, the UKF propagates a set of points (sigma points) distributed about the mean of the current estimate with a covariance that matches the covariance of the current estimate. After propagation, the set of sigma points is combined in a weighted sum to form the single state estimate at the next time step.

#### 7.2.1 State Representation

We plan to have our filter estimate the relative motion of the deputy spacecraft, thus we choose our state representation to be quasi non-singular relative orbital elements of the deputy, with respect to the chief. We will assume that our chief's absolute position and velocity (and thus absolute orbital elements) are well known (we may add noise to simulate how these absolute elements would likely come from its own filtering scheme). We choose this state representation because we only care about relative motion for this course, and quasi non-singular ROEs are the best choice for our application, and we have been using them this whole quarter.

#### 7.2.2 Dynamics Model

For our dynamics model, we plan to use the STM for quasi non-singular ROEs, including J2 effects. This is a linear model, and can be computed quickly on board of the spacecraft. We choose this model primarily for its simplicity.

Similar to previous problem sets, we plan to treat our chief ground truth as a non-linear simulation with J2 perturbations of the ECI state. The ground truth of the deputy spacecraft is also calculated using a non-linear simulation with J2 perturbations in ECI (and input controls). These simulated "true" values

plus Gaussian noise are the source of our measurement data, they are not used in the dynamics model of our filtering scheme.

### 7.2.3 Linearized Dynamics Model

Since our dynamics model is already linear, we do not need to perform further linearization.  $A$ , the linearized transition model, is the same STM with J2 perturbations that we have used on many of the recent problem sets.

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ -\left(\frac{3}{n} + \frac{7}{\kappa} EP\right)\tau & 1 & \kappa e_{xi} FGP\tau & \kappa e_{yi} FGP\tau & -\kappa FS\tau & 0 \\ \frac{7}{2}\kappa e_{yf} Q\tau & 0 & \cos(\dot{\omega}) - 4\kappa e_{xi} e_{yf} GQ\tau & \sin(\dot{\omega}) - 4\kappa e_{yi} e_{yf} GQ\tau & 5\kappa e_{yf} S\tau & 0 \\ -\frac{7}{2}\kappa e_{xf} Q\tau & 0 & \sin(\dot{\omega}) + 4\kappa e_{xi} e_{xf} GQ\tau & \sin(\dot{\omega}) + 4\kappa e_{yi} e_{xf} GQ\tau & 5\kappa e_{xf} S\tau & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ \frac{7}{2}\kappa S\tau & 0 & -4\kappa e_{xi} GS\tau & -4\kappa e_{yi} GS\tau & 2\kappa T\tau & 1 \end{bmatrix}$$

$B$  the control input matrix is the same  $B$  matrix in the reduced model we used in the previous problem set for continuous controls. Our control inputs will be delta-vs in RTN of the deputy at each timestep, thus the  $B$  matrix will be the matrix that maps these delta-vs in RTN to delta ROEs, which we have seen as  $\Gamma$  in the previous PSets. We will make the circular orbit assumption, so that our  $B = \Gamma_{circular}$ .

$$B = \frac{1}{na} \begin{bmatrix} 0 & 2 & 0 \\ \sin(u) & 2\cos(u) & 0 \\ -\cos(u) & 2\sin(u) & 0 \\ 0 & 0 & \cos(u) \\ 0 & 0 & \sin(u) \end{bmatrix}$$

### 7.2.4 Onboard sensors

For our measurements, we will propagate both the absolute positions and velocities of the chief and deputy in ECI, then add Gaussian White Noise to these values to generate simulated GPS measurements of both the chief and deputy. The standard deviation of the sensor is  $\frac{10}{\sqrt{3}}m$  for each component of position and  $2cm/s$  for each component of velocity. Thus, our measurements will be absolute GPS measurements (in ECI). Because of our large separation distance, so we cannot take advantage of closeness assumptions to only store relative position and velocity.

Note: GPS sensors actually provide output in ECEF. We choose to make the simplifying assumption that the GPS provides output in ECI, so that we do not have to deal with Earth's precession/nutation or complicated time synchronization.

### 7.2.5 Nonlinear measurement model

As we have discussed, our measurements are the ECI state of the chief and the deputy spacecraft. The state we propagate in the UKF is a state of relative orbital elements of the deputy. Thus, we need a mapping between ROEs of the deputy and absolute position and velocity of the deputy. Since we assume

the chief's absolute position and velocity "measurements" are well known, we can develop the following mapping:

$$\mathbf{r}, \mathbf{v} \text{ of chief and deputy in ECI} \leftrightarrow \text{OE of chief and deputy} \leftrightarrow \text{ROE of deputy}$$

So, our nonlinear measurement model  $h(x)$  shall map ROE of deputy to absolute position and velocity of the chief and deputy in ECI. In order to convert from ROE to ECI, we use the OE of the chief from our non-linear ground truth propagator. Thus,  $h(x)$  takes the following form:

$$h(x) : \text{ROE of deputy and OE of chief} \rightarrow \text{OE of deputy and chief} \rightarrow \mathbf{r}, \mathbf{v} \text{ of deputy and chief in ECI}$$

### 7.2.6 Associated sensitivity matrix which you would use to perform the EKF

Since there are many strong non-linearities in our measurement model, finding the linearized matrix mapping from state to measurement would be difficult and inaccurate. Thus, we choose to use a UKF in order to not have to derive this linearization. So will not have a sensitivity matrix, and instead we will use the full non-linear mapping from the state (ROE) to the measurement space (ECI state of both chief and deputy). This mapping is defined in the subsection above. (Shane said this was okay.)

However, for sake of some completeness, if we were to derive the sensitivity matrix, our process would be to take the Jacobian of  $h(x)$  with respect to the state. If we call this Jacobian  $H$ , then our linearized measurement model would be

$$Y_{k|k-1} = H X_{k|k-1}.$$

### 7.2.7 Our UKF Filtering Scheme

As mentioned, we plan to implement a UKF on the relative state, using linear dynamics and a nonlinear measurement model. We will use the following algorithm.

The UKF-A is initialized as prescribed in Alfriend:

$$\hat{x}_0 = E[x_0] \tag{1}$$

$$P_0 = E[(e_0 - \hat{x}_0)(x_0 - \hat{x})^T] \tag{2}$$

To propagate our dynamics, we do not need to use the sigma point method as our dynamics equation is a linear STM.

$$\hat{x}_k^- = Ax_{k-1} + Bu_{k-1} \tag{3}$$

Similarly our covariance matrix becomes

$$P_k^- = AP_{k-1}^+A^T + Q_{k-1} \tag{4}$$

However, to propagate our non-linear measurement model we do rely on sigma points.

$$X_{k|k-1} = [\hat{x}_k^-, \hat{x}_k^- + \gamma\sqrt{P_k^-}, \hat{x}_k^- - \gamma\sqrt{P_k^-}] \quad (5)$$

$$Y_{k|k-1} = h(X_{k|k-1}) \quad (6)$$

where  $h(x)$  maps ROE to ECI of both chief and deputy.

$$\hat{y}_k^- = \sum_{i=0}^{2n} W_i^{(m)} Y_{i,K|K-1} \quad (7)$$

$h(x)$  is our non-linear measurement model described in detail above.  $W$ , the scalar weights used to combine the sigma points, are given in the Alfriend textbook.

For the final step, we create the Kalman gain and perform the measurement update.

$$P_{\tilde{y}_k, \tilde{y}_k} = \sum_{i=0}^{2n} W_i^{(c)} (Y_{i,k|k-1} - \hat{y}_k^-) (Y_{i,k|k-1} - \hat{y}_k^-)^T + R \quad (8)$$

$$P_{x_k, x_k} = \sum_{i=0}^{2n} W_i^{(c)} (X_{i,k|k-1} - \hat{x}_k^-) (X_{i,k|k-1} - \hat{x}_k^-)^T \quad (9)$$

$$K_k = P_{x_k, y_k} P_{\tilde{y}_k, \tilde{y}_k}^{-1} \quad (10)$$

$$\hat{x}_k = \hat{x}_k^- + K_k (y_k - \hat{y}_k^-) \quad (11)$$

$$P_k = P_k^- - K_k P_{\tilde{y}_k, \tilde{y}_k} K_k^T \quad (12)$$

Finally, the process repeats at each timestep.

### 7.2.8 Simple test cases.

We have already performed much testing of our dynamics model, since we have been using it in previous PSets.

We have also already written and tested functions that perform the conversions from ROE to OE to ECI, which define our measurement model.

## 8 Problem Set 8: Implement navigation filter.

### 8.1 Ground truth generation

We use MATLAB's ode113 to numerically integrate the fundamental orbital differential equation in absolute position and velocity in ECI. This FODE includes J2 effects. We do this with both the chief and deputy.

### 8.2 Convert ground truths into measurements by adding Gaussian noise.

We sample both the chief's and deputy's ground truth and add gaussian white noise to create our measurements. The gaussian white noise is centered at zero and the covariance matrix is written in section 8.4.

### 8.3 Set initial estimate and covariance.

We choose our initial estimate of the deputy's relative position to be our estimate of the chief's position. (So the initial ROEs of the deputy are all zeros.) And we set the initial covariance to a diagonal matrix consistent with the precision of a common GPS sensor (as discussed in the previous PSet).

### 8.4 Define process and measurement noise covariance

Our measurement noise in  $a\delta\alpha e$  space is about 20 meters:

$$Q = \frac{0.02}{a} I_{6x6} \approx 3e-6 I_{6x6}$$

We choose the measurement noise based on our understandings of the limitations of GPS in low earth orbit. Specifically, we expected variance in position measurement of about 10 m and a variance in velocity measurements of about 2 cm/s.

$$R = \frac{1}{1e3} \text{Diag}([10, 10, 10, .02, .02, .02, 10, 10, 10, .02, .02, .02]).$$

### 8.5 Combine all the parameters to mechanize the Unscented Kalman Filter

See UKF implementation in code.

### 8.6 Functionality and performance of the UKF

Our first step was to analyze the true versus estimated ROEs and the confidence interval based on our sigma matrix.

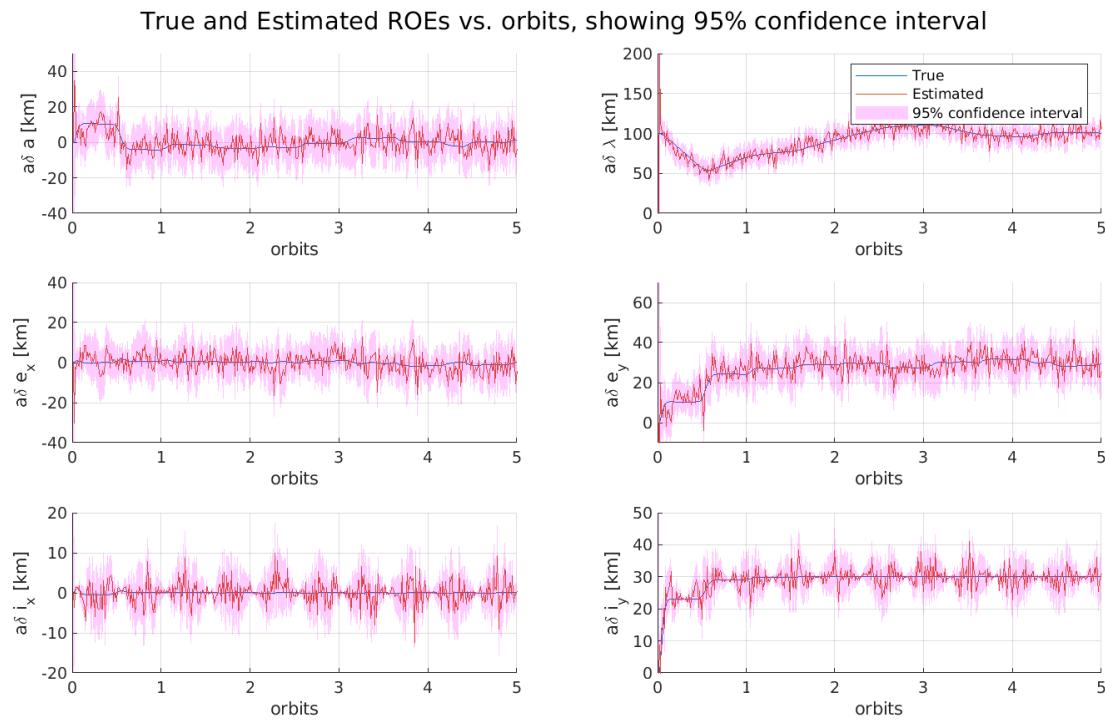


Figure 66: We can see that the estimated state is reasonable stable and close to the true state.

In addition to our error bounds, we plotted the pre and post-fit residuals.

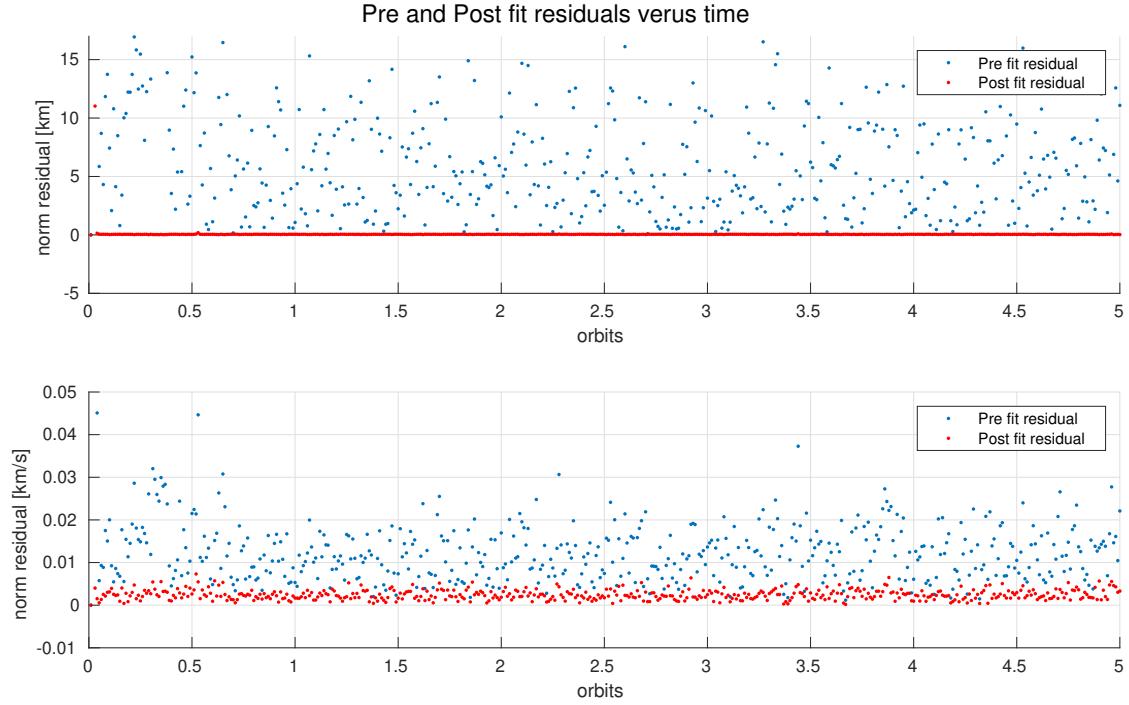


Figure 67: It is not too surprising that we are seeing a pre-fit with residual that is an order of magnitude greater than the post-fit residual because the measurement data is reasonable noisy when it comes in.

We also looked at our error statistics in steady state (for the last orbit).

	mean error ([km])	standard deviation of error ([km])
$a\delta a$	4.426	3.069
$a\delta\Omega$	5.252	4.049
$a\delta i_x$	3.472	2.604
$a\delta i_y$	3.644	2.832
$a\delta e_x$	1.974	1.676
$a\delta e_y$	2.016	1.872

These numbers were about as expected based on visually looking at the figure 66.

Further, we plotted the post fit residual against the 95th percentile of injected measurement noise.

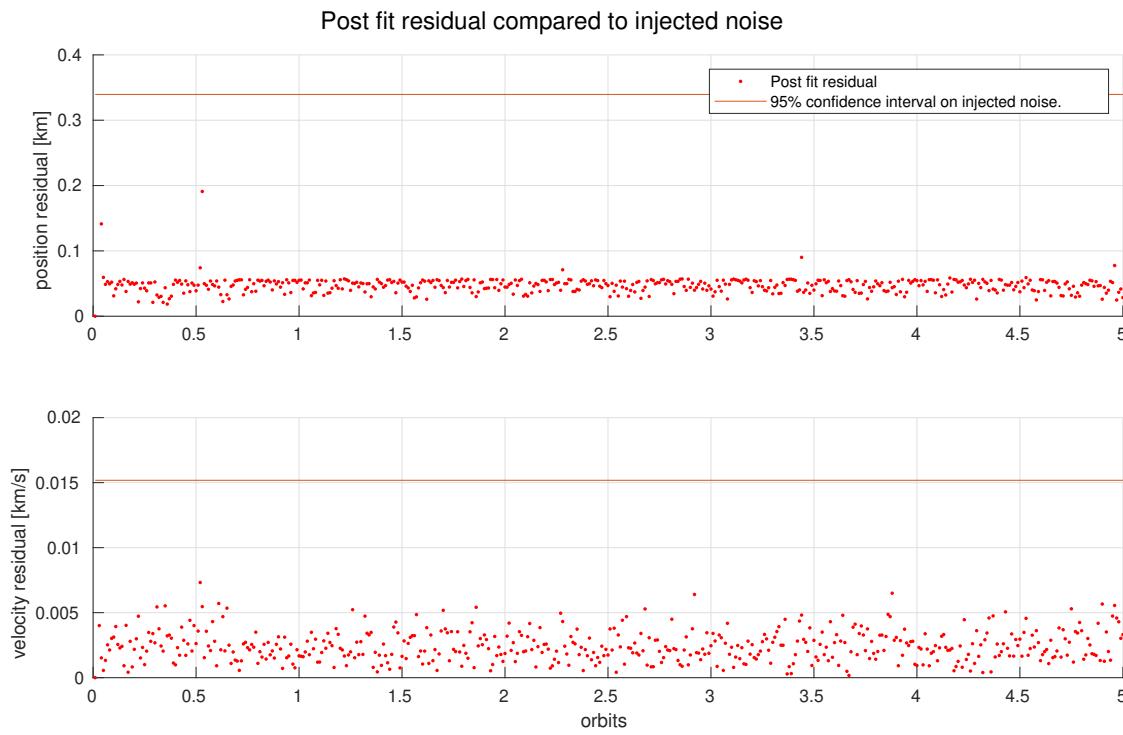


Figure 68: We can see here that the post fit residual of both position and velocity is less than the 95 percentile upper-bound of noise injected during the measurement model step. This reflects well on our model.

Lastly, we plotted the error of our model for each relative orbit element against the associated 95th upper confidence bound of Sigma to check that our errors were in the right range.

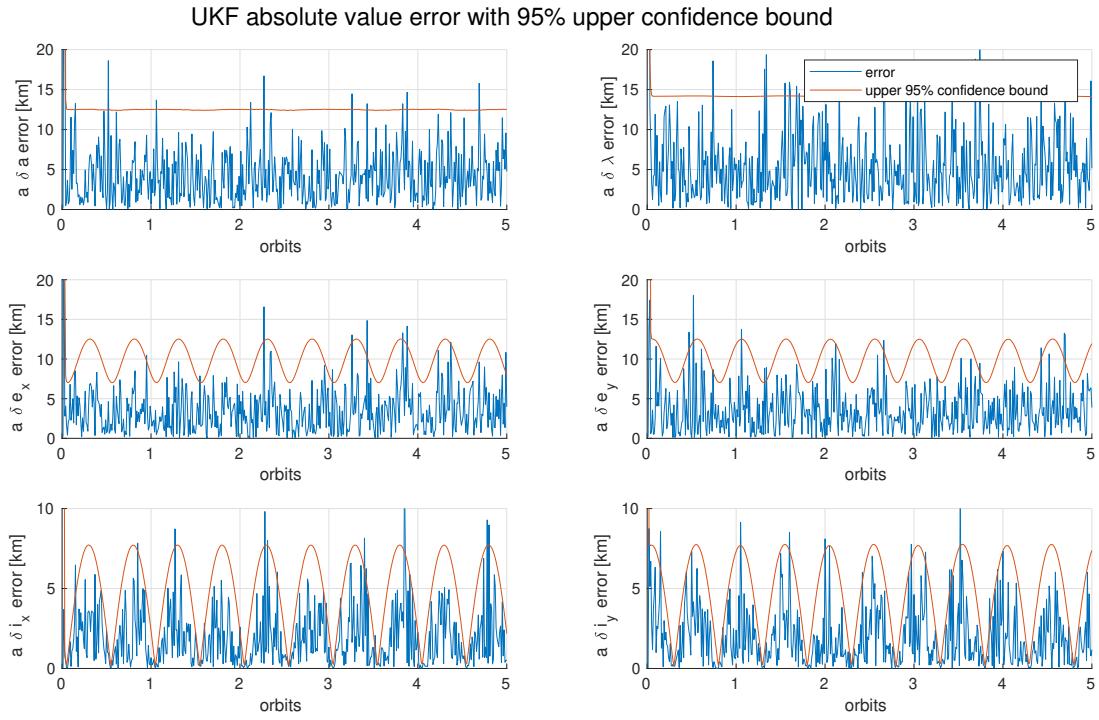


Figure 69: The state error is generally less than the 95 percentile upper confidence bound as one would expect. Further, the error terms are cyclical with the 95 percentile upper confidence bound.

## 9 Problem Set 9: Final Project and Closing the loop between Nav and Control

### 9.1 Problem 1: Catching Up

Below is an outline of updates we have made to previous problem sets.

#### 9.1.1 General

We have transitioned to not including our code in this document. Please find all our code here, or on GitHub using this link: <https://github.com/xosgood/AA279D>.

#### 9.1.2 PSet 1

We updated our graphs in problem set 1 for clarity by decreasing the number of orbits graphed from 100 to 30 and creating separate graphs for orbital elements. Additionally, we had made a mistake of propagating the keplarian orbit with mean orbital elements, so we changed our mean elements to osculating and propagated those instead. We additional had some typos and incorrect labeling of graphs that we fixed. One challenge that we attempted to address, but ultimately did not was the oscillation of our mechanical energy in the orbit propagation without J2. At first, we assumed there had been a typo in our code, but after looking more into the problem it was apparent that the variation in mechanical energy came from variation in the radius in the simulation without J2 appears to be a result of numerical noise in the simulation.

#### 9.1.3 PSet 2

We modified our delta-v calculation for the semi-major axis changing maneuver to occur in the RTN frame of the deputy rather than the RTN frame of the chief. As a result, after the maneuver both spacecraft have the same semi-major axis in a way that previously was not true.

#### 9.1.4 PSet 3

For PSet 3, we removed a surplus  $\delta e_z$  typo the existed in two places where we were defining ROE vectors. Further, we slightly modified our graphs of RTN position and velocity to include R on the y-axis rather than the x-axis.

#### 9.1.5 PSet 4

We again updated our notation of the ROE vectors to remove the  $\delta e_z$  typo. We further modified our delta-v calculation in section 4.1.6. Previously we had been off by an order of magnitude because of a mistaken placement of parentheses.

### 9.1.6 PSet 5: Impulsive control

We have updated our formation keeping control law to correct all compounded error due to perturbations. In particular, now our formation keeping burns correct for non-zero  $\delta a$  and drifting  $\delta \lambda$ , in order to eliminate along track drift. Following this, we have also added discussion of the optimal delta-v for this portion of the maneuver.

### 9.1.7 PSet 6: Continuous control

We updated plots to make them more readable. We liked the continuous controller a lot, and it worked really well, so we are keeping it. We also added further discussion of our Lyapunov controller and uncertainty.

### 9.1.8 PSet 7: Navigation filter setup

Fixed control conclusions as mentioned in PSet 5 and 6. Removed unnecessary comments.

### 9.1.9 PSet 8: UKF

For Pset 8, we re-did the true and estimated state plot to include 95th percentile confidence bars around the estimated state. Further, we plotted the pre and post-fit residuals against each other and plotted the post-fit residuals against the 95 percentile line of injected noise. We also added a table of the mean and standard deviation of the steady state solution which we had not included in the original submission. Lastly, we add the 95 percentile upper confidence bound to the UKF error plots based on the diagonal entries of Sigma.

## 9.2 Problem 2: Integration of Navigation and Control

In this problem, we close the loop between Nav and Control. Since our satellite never actually knows its ground truth state, it must be making control decisions based off of its state estimate.

### 9.2.1 Control law selection

We choose to use our continuous Lyapunov controller for reconfiguration and formation keeping.

### 9.2.2 Baseline

For this PSet, we have slightly updated the parameters for our Lyapunov controller from PSets 6 and 8. To provide a baseline, here is the performance of our simulation without closing the loop between navigation and control, and without injecting noise into the control input computations.

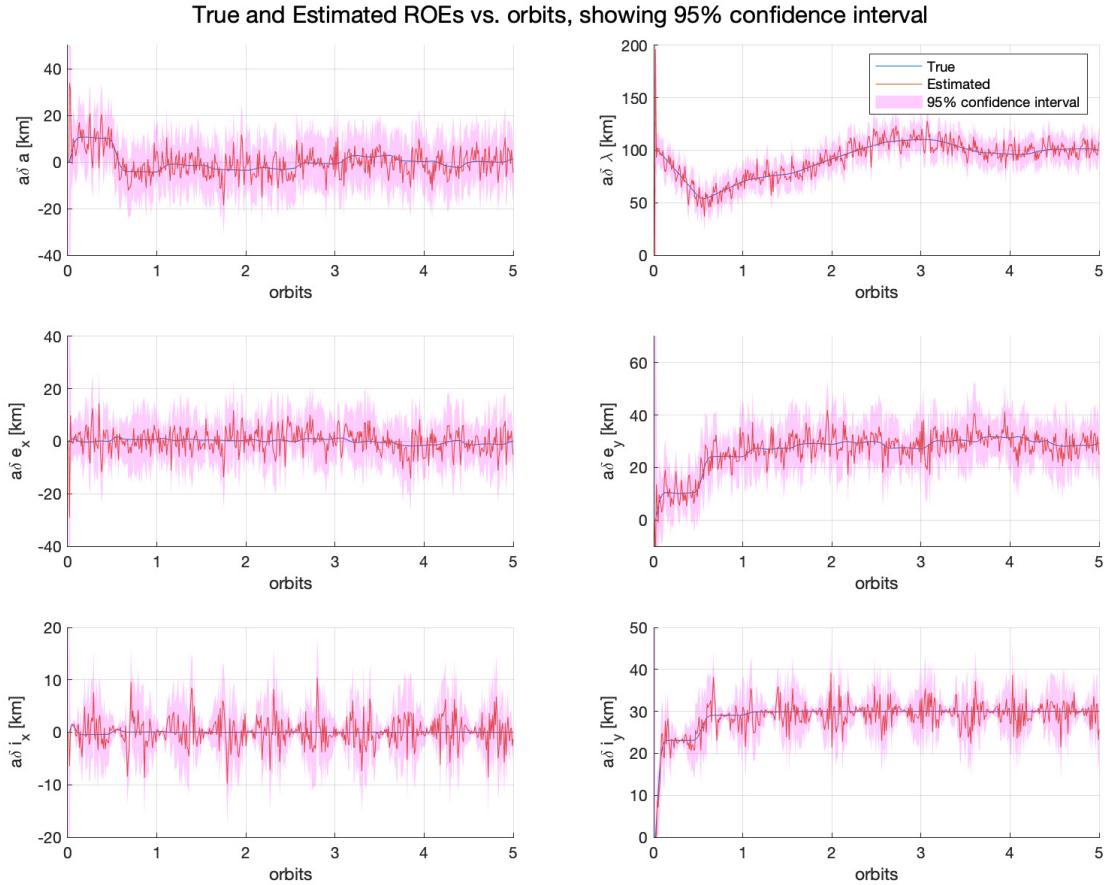


Figure 70: Baseline performance. Plot showing scaled ROEs vs number of orbits passed. This is without closing the loop between navigation and control, and without injecting noise into the controller inputs.

We see most of the reconfiguration maneuver occur in the first orbit, then see  $\delta\lambda$ ,  $\delta e_y$ , and  $\delta i_y$  settle to their desired values. There is initial variation in  $\delta\lambda$ , and we control  $\delta a$  in order to slowly drive  $\delta\lambda$  to the desired value.

Our filter (UKF) tracks the ground truth well, and is not overconfident. We do see the covariance measurements sinusoidally oscillating between larger and smaller, likely due to the periodic relative motion introduced by the maneuver as well as J2 perturbations.

Below is another plot showing the steady state performance up to 15 orbits:

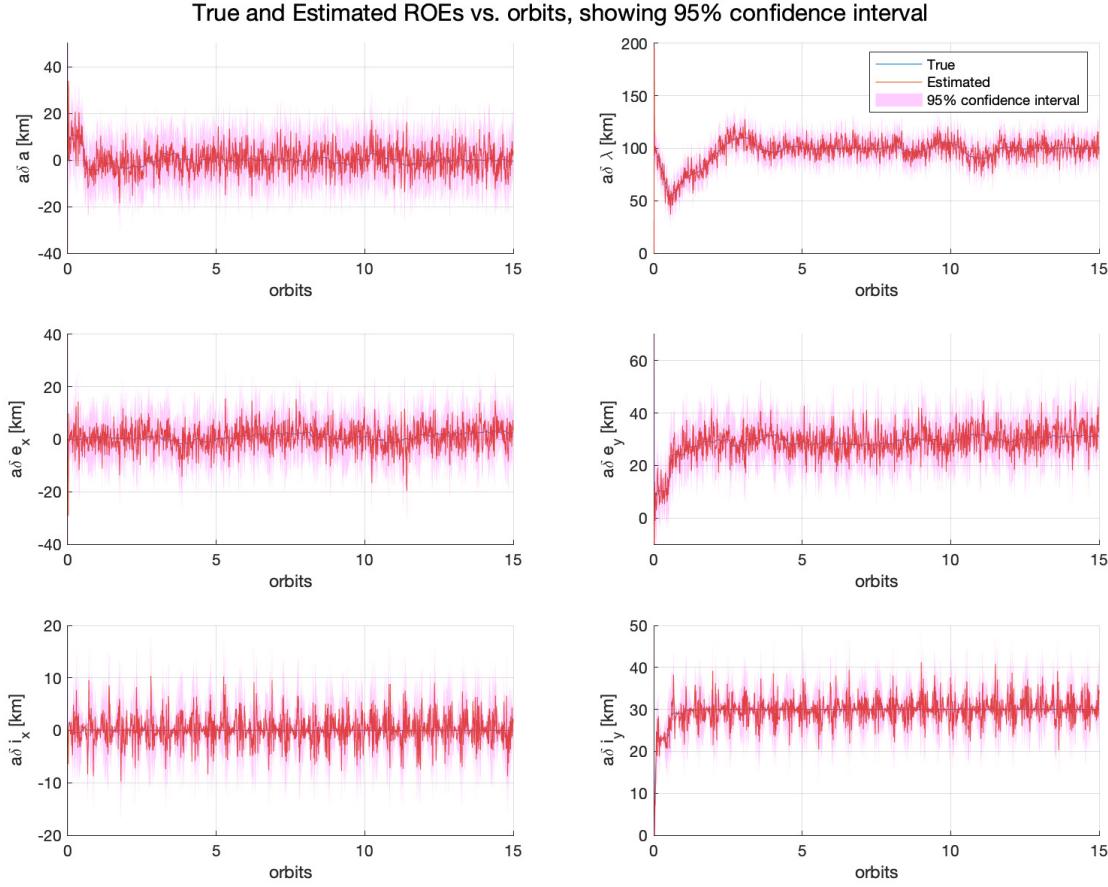


Figure 71: Baseline performance over longer time. Plot showing scaled ROEs vs number of orbits passed. This is without closing the loop between navigation and control, and without injecting noise into the controller inputs.

### 9.2.3 Corrupt ground truth for controller

Before fully closing the loop between Nav and Controls, we want to see how the controller performs with a noisy ground truth input. We chose to add Gaussian white noise with covariance equivalent to our process noise (as defined in PSet 8) to the ground truth state (in the ROE space) that we send to the control law.

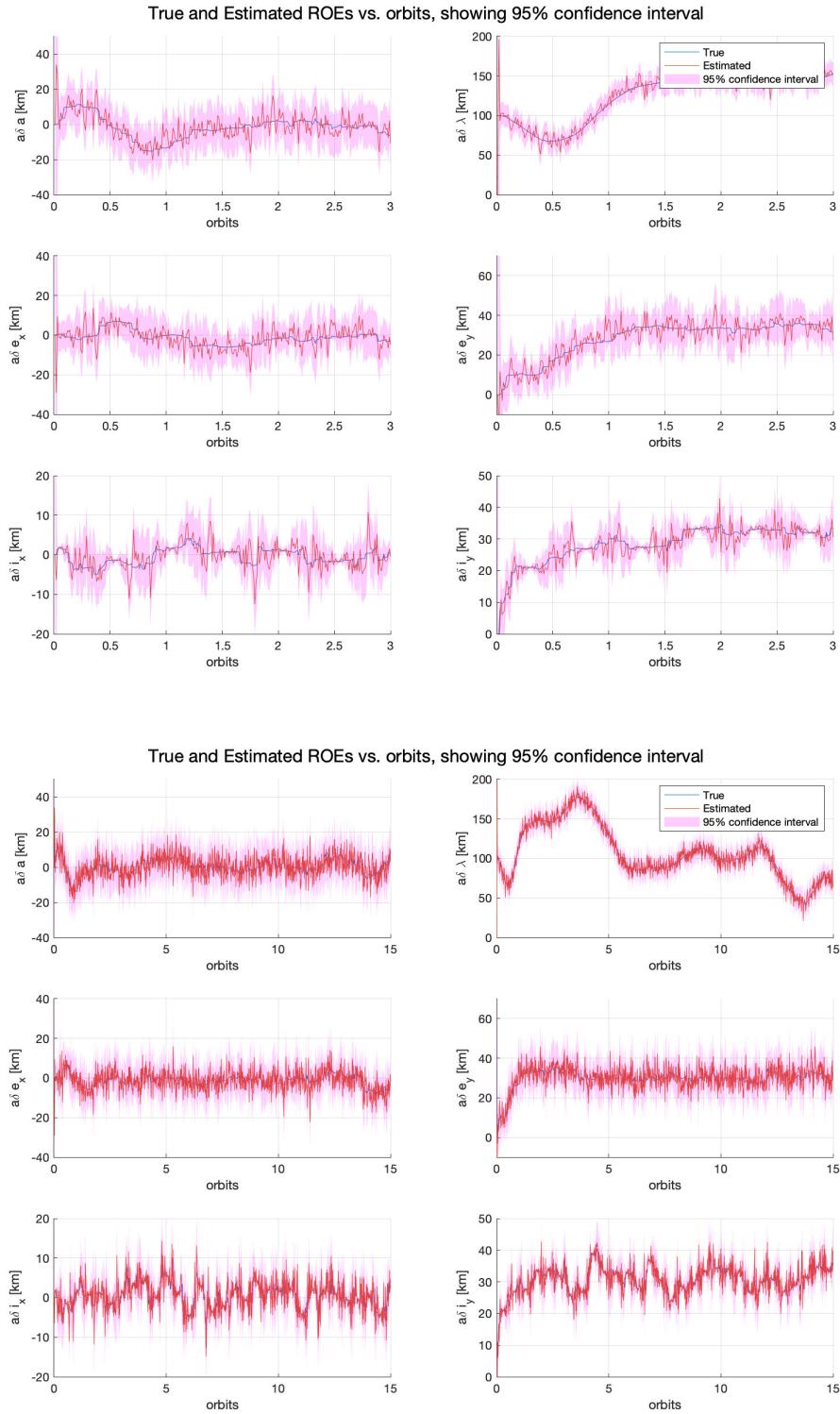


Figure 72: Performance with noisy controller. Plot showing scaled ROEs vs number of orbits passed. The top plot is zoomed in over the first few orbits, and the bottom plot shows up to 15 orbits.

In this case, the controller unsurprisingly performs much worse than with the ground truth inputs. It does a very poor job controlling  $\delta\lambda$ , but does okay at controlling the rest of the relative orbital elements. Of course, it is still very noisy for all elements. Looking at the longer plot, we see that the controller may not necessarily converge, especially for  $\delta\lambda$ . Perhaps one could tune the controller a bit better to reject disturbances, but our relatively crude results in a mediocre controller.

However, our filter (UKF) still performs quite well, never getting over confident, or diverging from the ground truth (although it is still just as noisy). This is as expected because the filter still gets the proper measurements and still has knowledge of the noisy control inputs.

#### 9.2.4 Closing the loop between Navigation and Controller (with analysis)

Now, we use our UKF's estimate as inputs for the controller to compute control inputs for the next timestep. Here is the result:

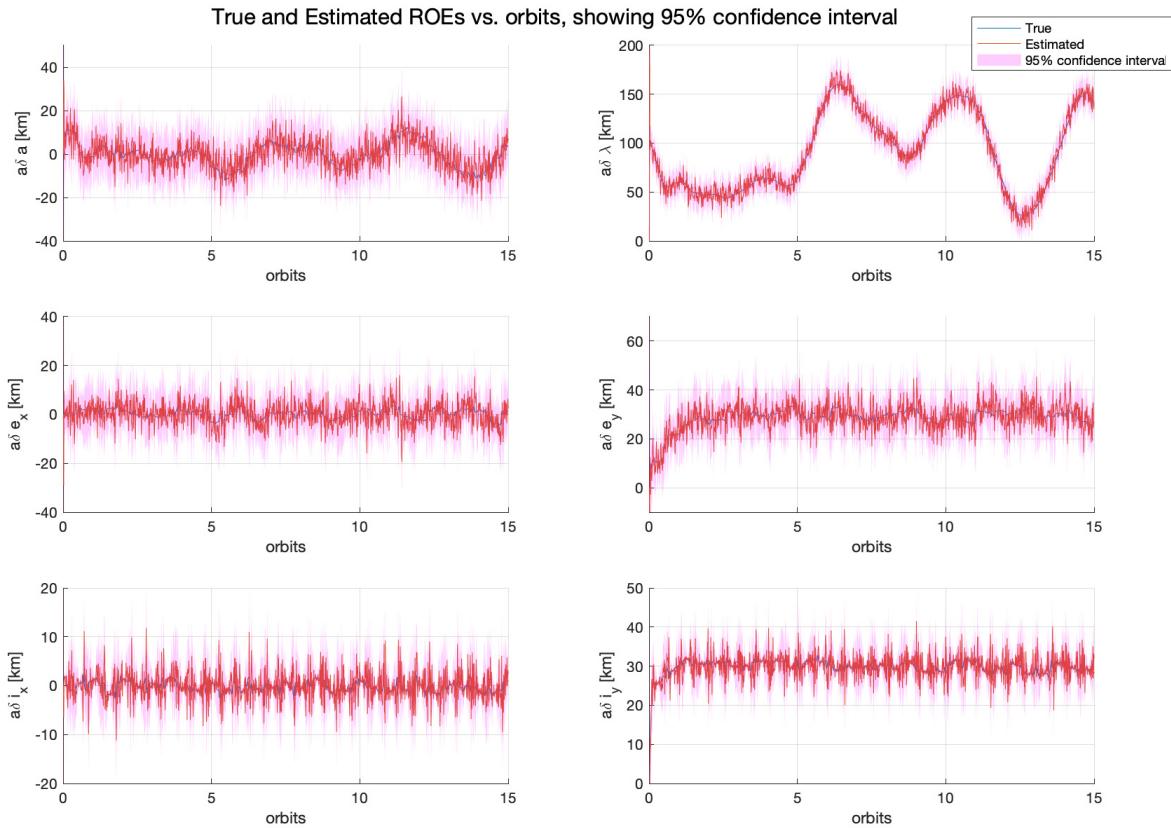


Figure 73: Closed loop performance. Plot shows scaled ROEs vs number of orbits passed. In this case, the control law is computed using the UKF's state estimate.

Our filter continues to predict state reasonably well, but there is a lot of noise in the filter's estimate. Our controller performs poorly, especially in the  $\delta\lambda$  component. We can even see it going unstable along with  $\delta a$ . All the other relative orbital elements seem stable, but have high frequency steady state oscillations.

Thus, it is obvious that we need to better tune our controller to reject disturbances (those disturbances being noise in the filter's estimation).

We can tune our controller by adjusting  $k$  (essentially the overall gain),  $N$  (affecting the impulsivity of the controller), as well as the parameters that set our maximum and minimum desired  $\delta a$  and  $\delta \lambda$ . After a few hours of tuning to no avail, we found that this controller matched with the noise of the filter was insufficient to converge in the  $\delta \lambda$  component. So, we introduced more granularity to our control by changing the  $P$  matrix to

$$P = \begin{bmatrix} \frac{1}{k_{IP}} \cos(J)^N & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{k_{IP}} \cos(J)^N & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{k_{IP}} \cos(J)^N & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{k_{OOP}} \cos(H)^N & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{k_{OOP}} \cos(H)^N \end{bmatrix}$$

where  $k_{IP}$  and  $k_{OOP}$  are arbitrary scalars valued similar to  $k$ . We ended up using  $k_{IP} = 10$  and  $k_{OOP} = 6$ . This should allow us to further decouple the in-plane and out-of-plane control, however it was insufficient to fix our controller issues.

Furthermore, we tried adding a low pass filter (moving average) to the input to the controller. We ended up using a moving average over the previous 4 timesteps (about 4 minutes of an orbit). This helped, but the controller still was unstable in the  $\delta \lambda$  component.

At this point, we concluded that either our control law was fundamentally insufficient for the task of our problem, or our measurements were insufficient for the granularity of control we desired. Or both. Since designing a whole new controller (see potential future work) seemed out of the scope of this project, we decided that the path forward would be decrease the measurement noise (to something still reasonable, but definitely better than basic GPS measurements). We decreased the measurement noise covariance by a factor of 2 such that

$$R = \frac{1}{2e3} \mathbf{Diag}([10, 10, 10, .02, .02, .02, 10, 10, 10, .02, .02, .02]).$$

Here are the results following these changes:

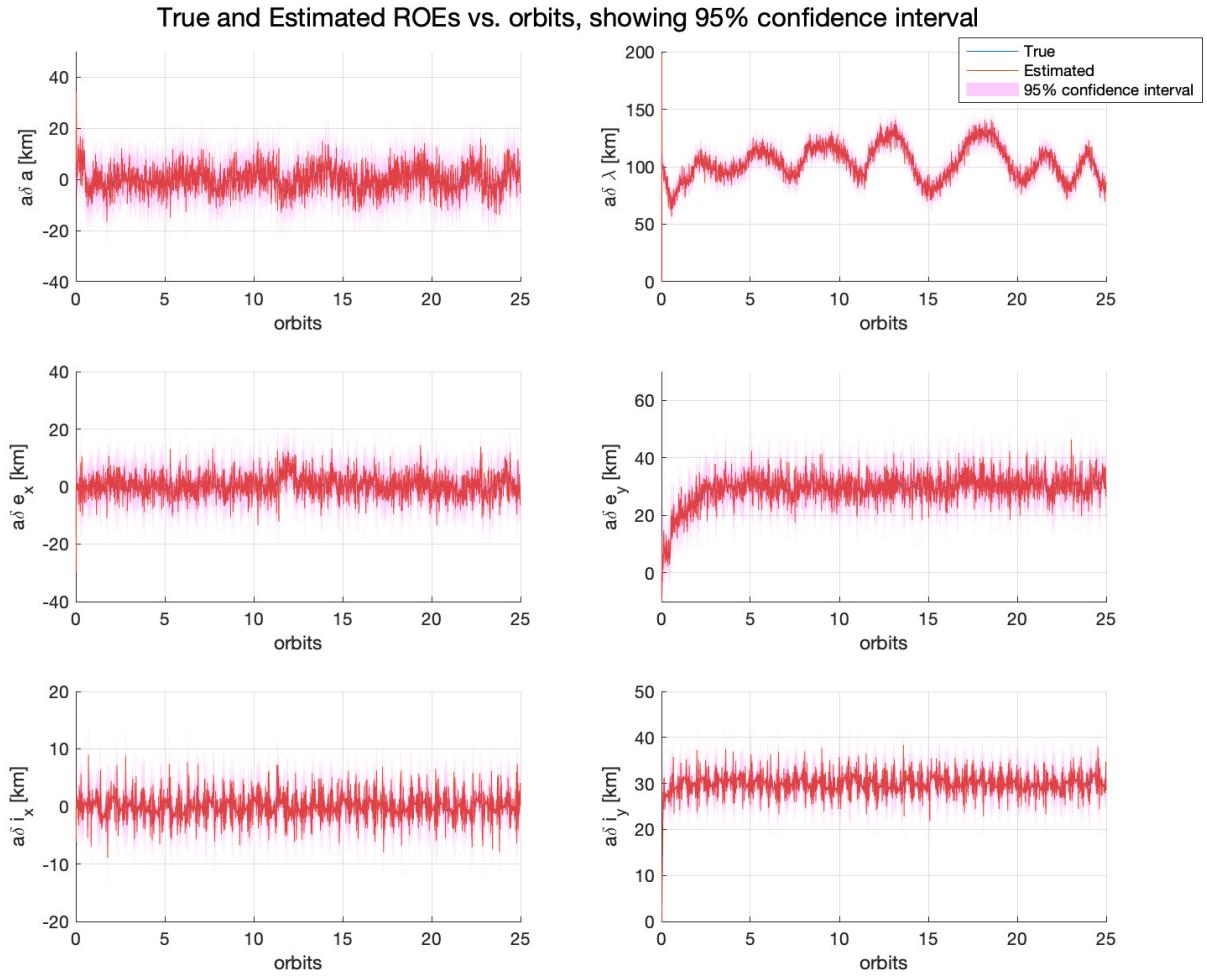


Figure 74: Closed loop performance. Plot shows scaled ROEs vs number of orbits passed. With updated controller parameters, and decreased measurement noise.

We still do not see convergence in the  $\delta\lambda$  direction, but it is at least not unstable.

If we decrease  $R$  by a full order to magnitude, we get the following results:

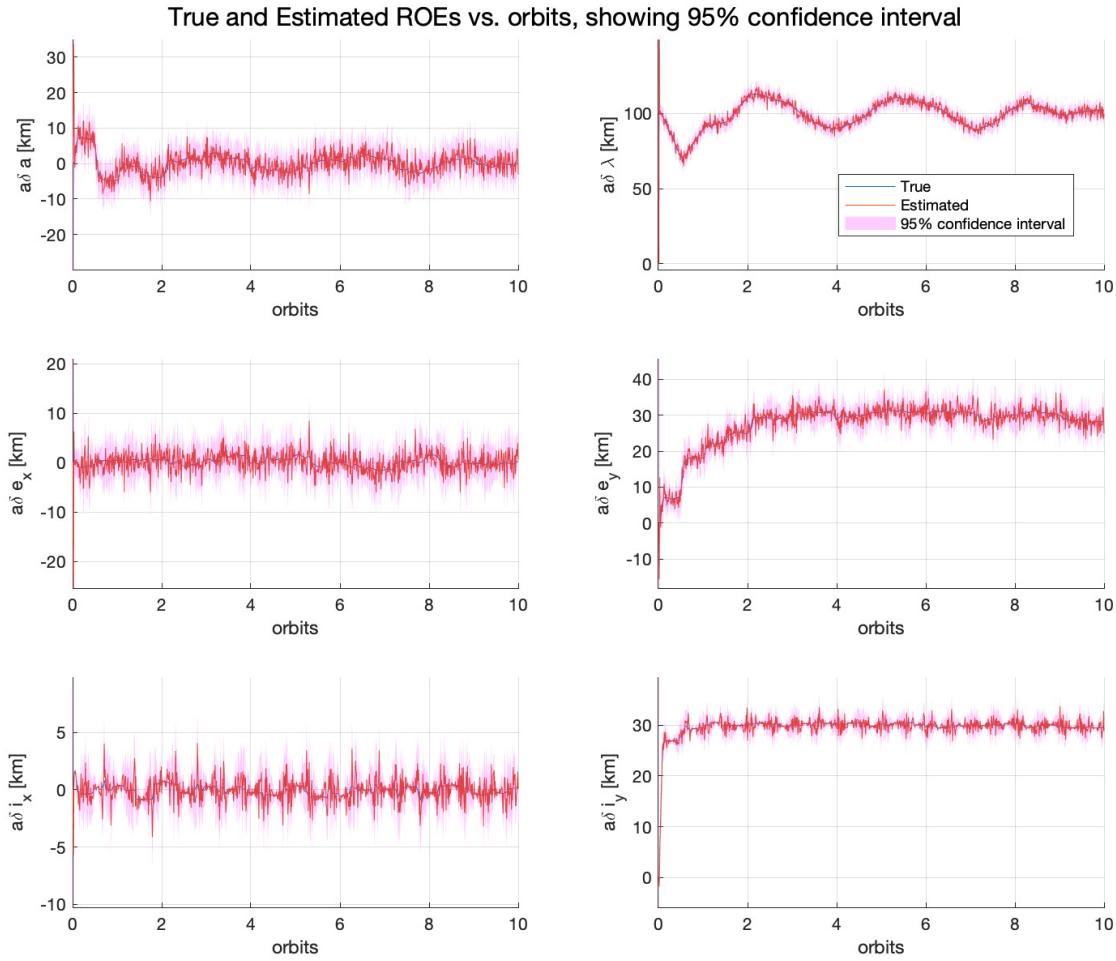


Figure 75: Closed loop performance. Plot shows scaled ROEs vs number of orbits passed. With measurement noise covariance decreased by an order of magnitude.

Here, we finally see the controller converge to tighter bounds which are reasonable for our mission.

Both of the previous simulations show better controller performance than in part (b) when random noise on top of the ground truth is input to the controller, rather than the state estimate. This makes sense, as our UKF should do a better job estimating the state than random noise on top of the ground truth.

### 9.2.5 Performance metrics

The performance metrics below are for the closed loop simulation with the most updated control parameters and the lowest magnitude measurement noise covariance.

General performance over 10 orbits:

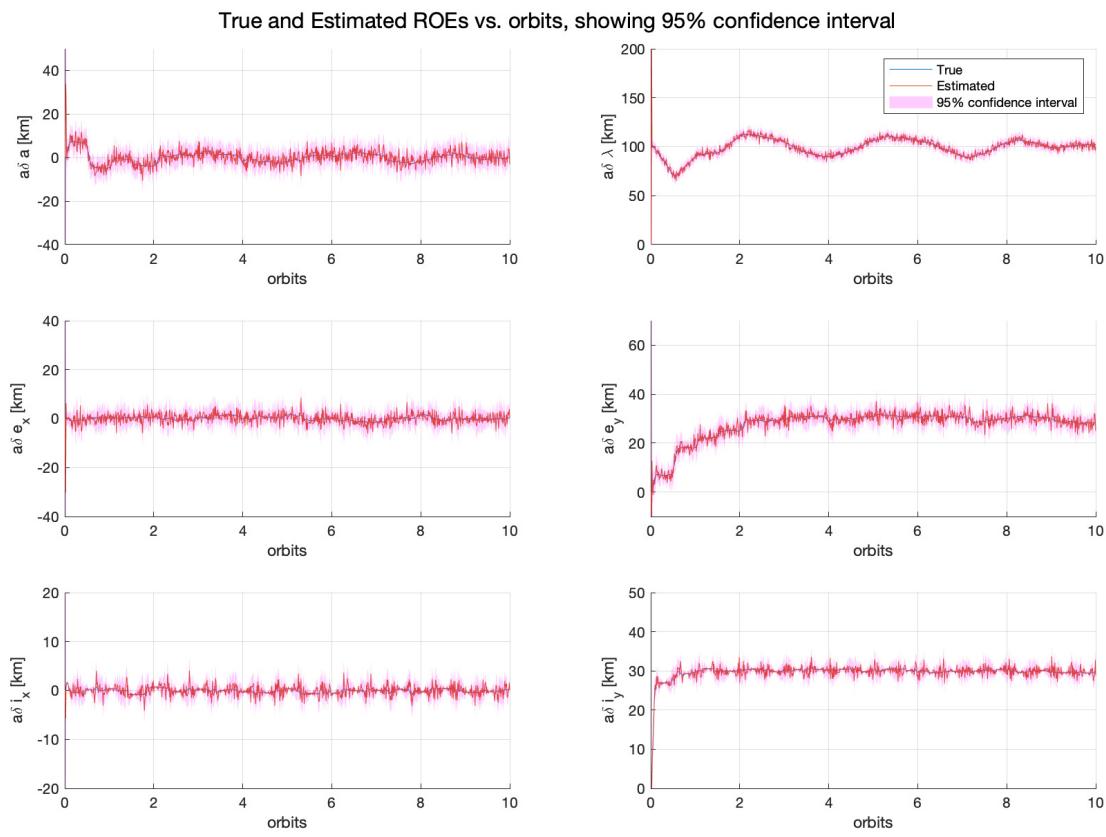


Figure 76: True and estimated ROEs over 10 orbits, with 95% confidence interval.

Pre-fit and post-fit residuals:

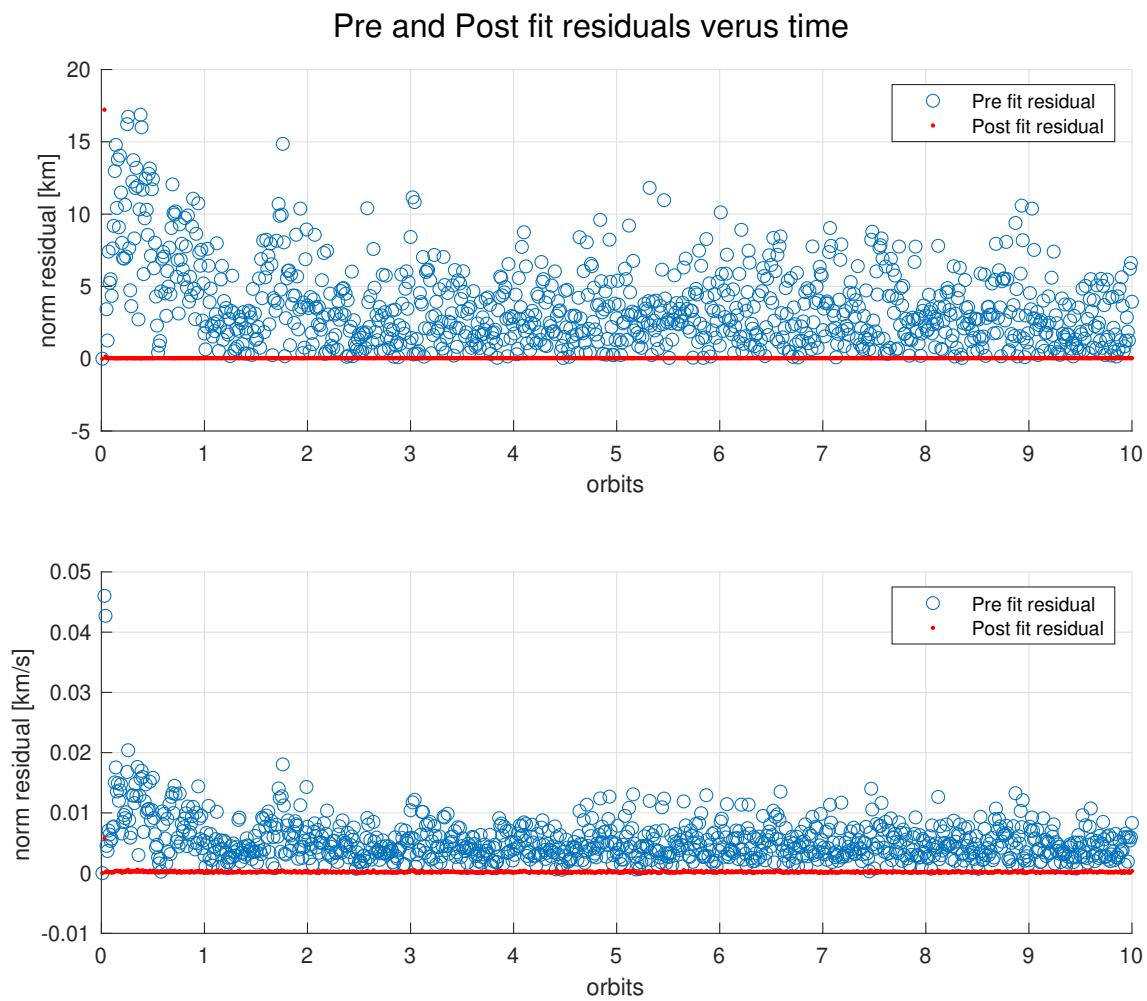


Figure 77: Pre-fit and post-fit residuals.

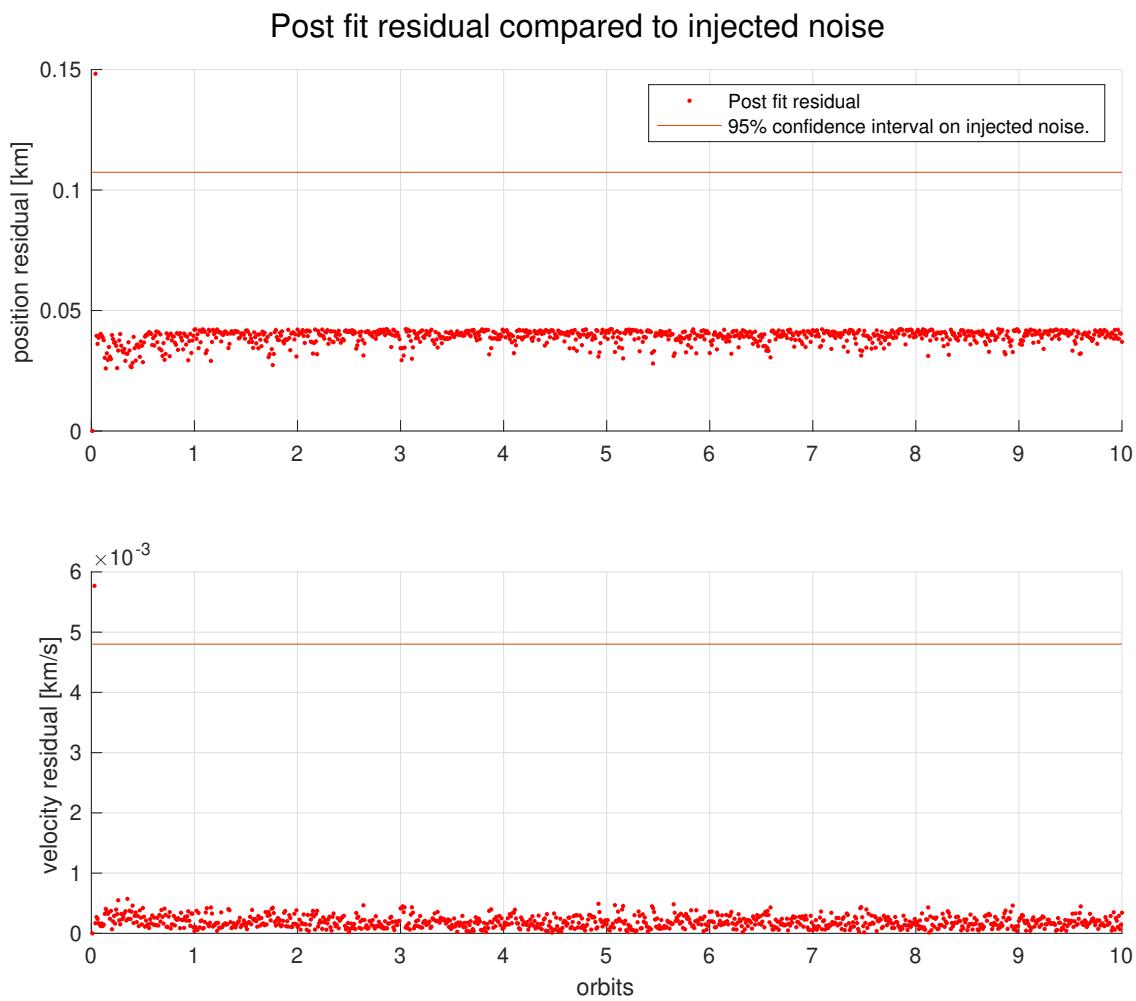


Figure 78: Post-fit residuals with 95% line from measurement noise covariance matrix.

Errors:

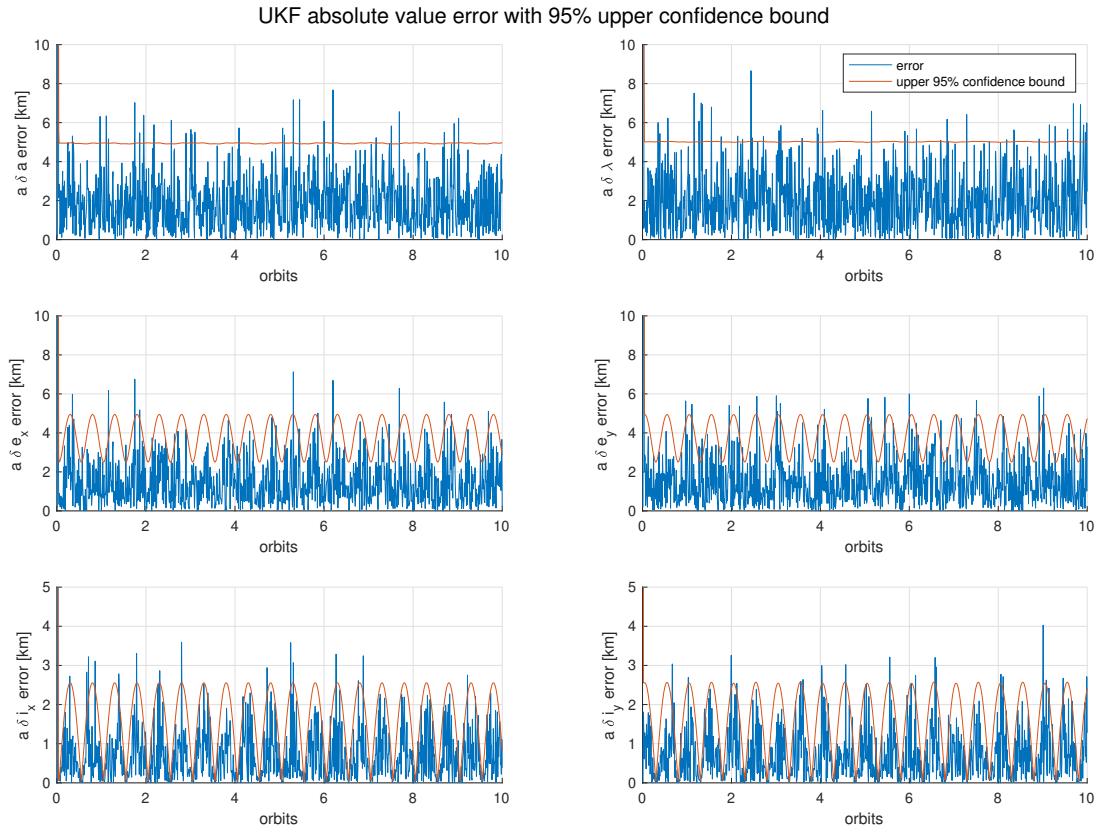


Figure 79: UKF state estimate errors.

Steady state (last orbit) true error metrics:

	mean error ([km])	standard deviation of error ([km])
$a\delta a$	1.829	1.34
$a\delta \Omega$	2.263	1.70
$a\delta i_x$	1.396	1.06
$a\delta i_y$	1.5659	1.26
$a\delta e_x$	0.6810	0.60
$a\delta e_y$	0.7829	0.76

Control inputs:

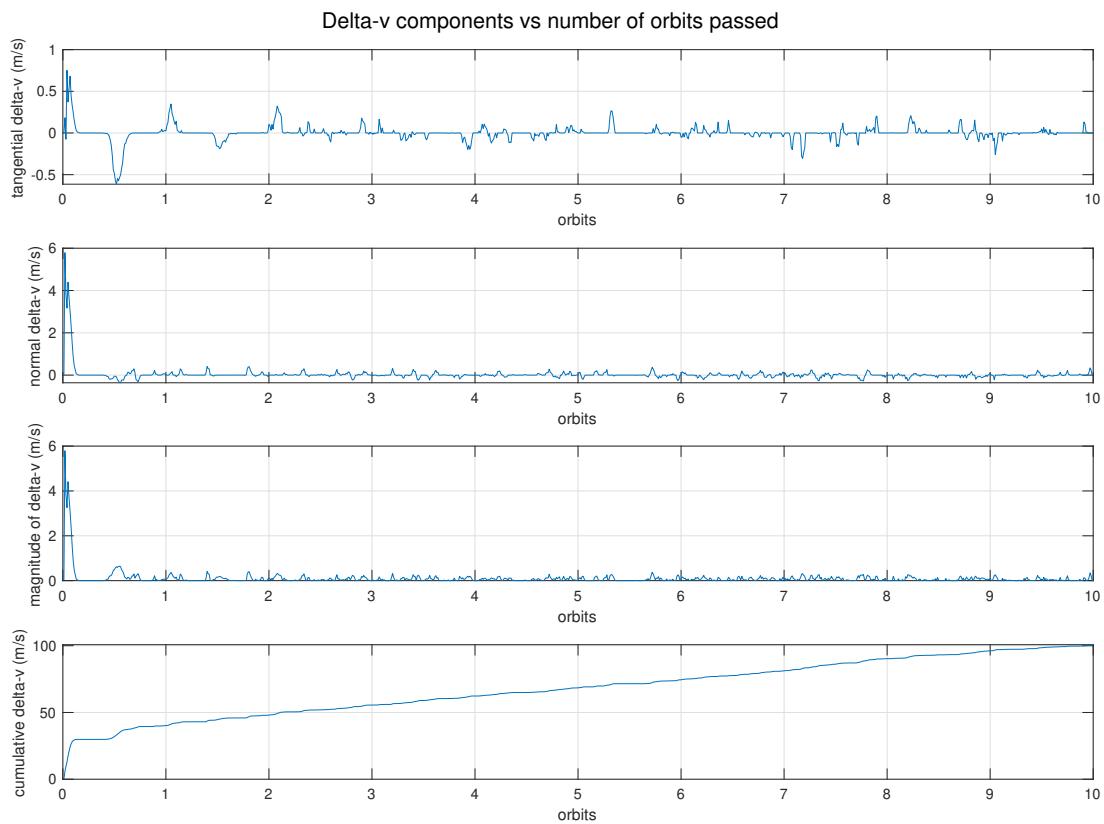


Figure 80: Control input history in each direction, magnitude, and cumulative.

Control tracking error:

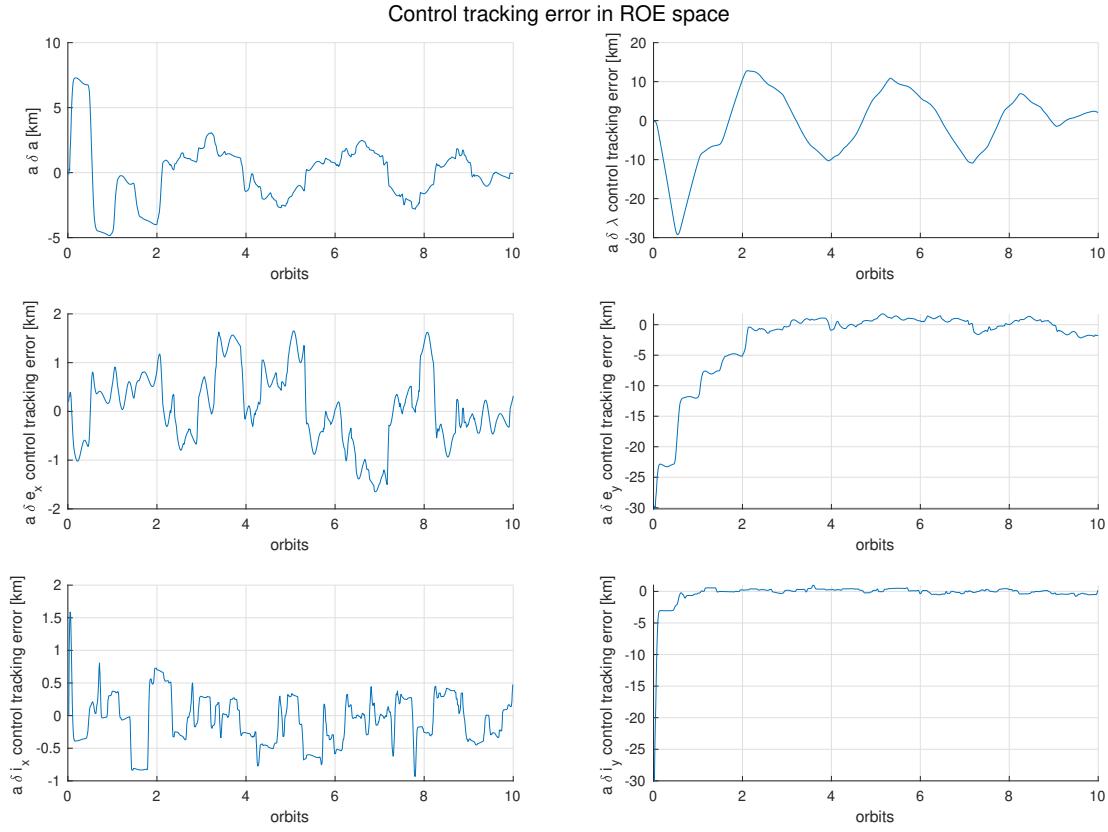


Figure 81: Control tracking error.

All of these performance metrics meet expectations. Mainly, the control tracking error in  $\delta\lambda$  is the worst. It appears one may need a better, more complex continuous controller in order to improve on the control tracking error.

The convergence time for the out-of-plane relative elements is about 1 orbit. The convergence time for the in-plane elements is much slower. For  $\delta e_x$  is it immediate, because we start in the desired state. For  $\delta e_y$ , it is about 2 orbits. For  $a\delta a$  and  $a\delta\lambda$ , it takes about 2 orbits to get to a small periodic ringing around the desired state, but it takes about 9-10 orbits to settle within about  $\pm 2\text{km}$  of the desired value.

### 9.3 Problem 3: Extension of our choosing

We have opted not to fully do this optional problem.

However, we did play around with adding actuator uncertainties to the controller. We just added random noise to the computed RTN delta-vs, with covariance of  $\text{diag}([0, 2\text{e-}7, 1\text{e-}7])$ .

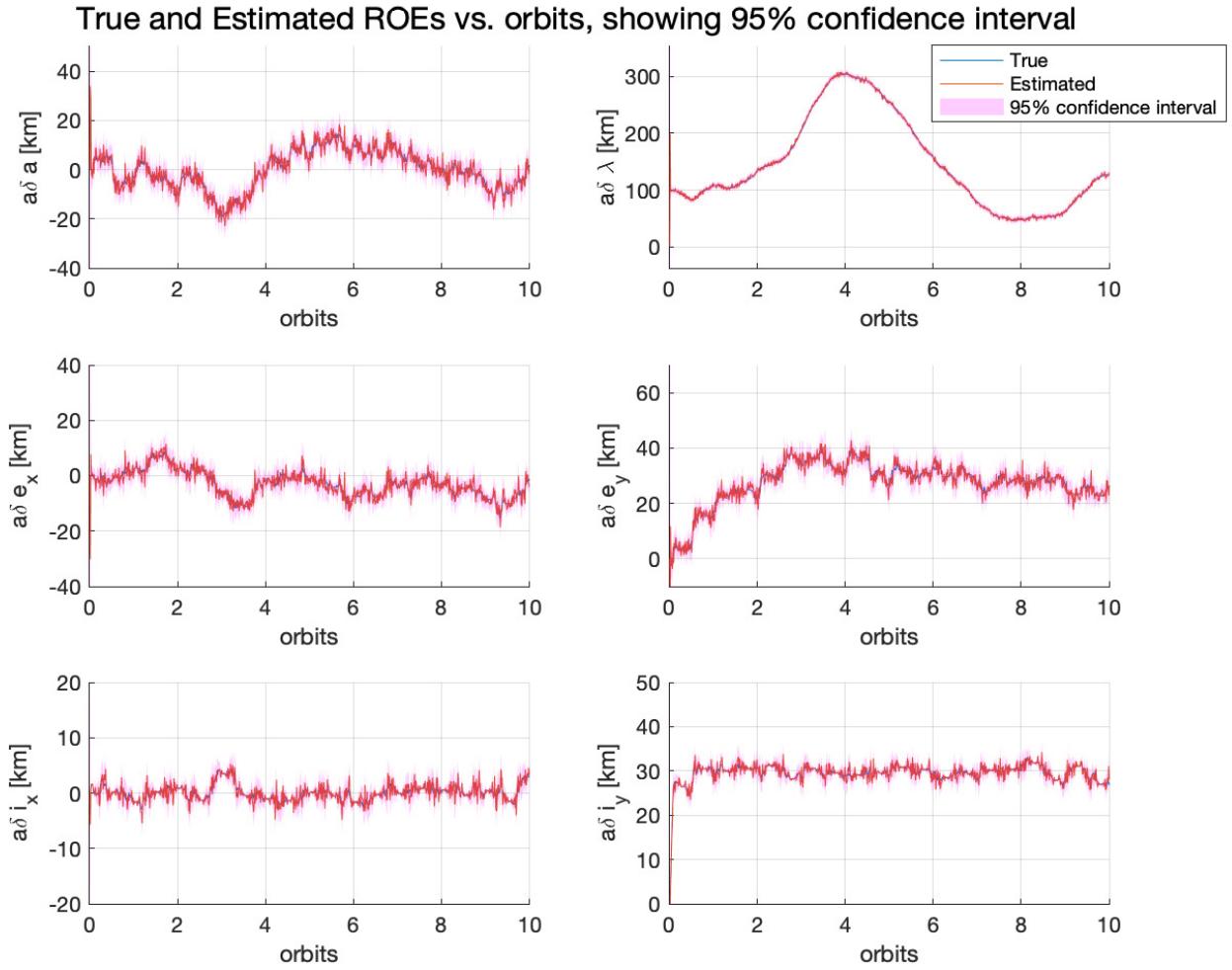


Figure 82: ROEs vs time showing simulation performance, with noisy actuators.

The results are obviously not very good. We definitely need a better controller, especially for the  $\delta\lambda$  component.

## 9.4 Problem 4: Conclusion and final remarks

### 9.4.1 Brief Summary

We have built an entire simulation for a single chief-deputy pair of the Starling satellites. We started with simulating the ground truth dynamics, then moved towards implementing impulsive and continuous controllers, and finally added an Unscented Kalman Filter on top for navigation. We closed the loop between navigation and controls in order to mimic the flight software that would be on one of these satellites.

### 9.4.2 Results Assessment

We are pleasantly impressed with the vast majority of our results. Our dynamics integration and propagation works well, although we see more numerical noise than we'd like. We are impressed by the combined simplicity and functionality of state transition matrices. Our control algorithms worked surprisingly well, both impulsive and continuous, and were also much closer to optimal than we expected. Our navigation algorithms also worked decently. We expected the UKF to work a bit better, but we suspect it's poor performance is due to relatively poor/noisy measurements rather than poor algorithmic choice. After closing the loop between navigation and control, our controller did seem to be more unstable than we would have hoped. We suspect that it is more an issue with insufficient measurements (since decreasing  $R$ , the measurement noise covariance, lead to significant improvements in controller performance), we also suspect that our controller isn't as robust to disturbances as it should be. All of these are things that could be improved in future work (see future work section).

### 9.4.3 Lessons Learned

We really learned how to choose the simplest approach that fit our needs. There is no need to use a complicated non-linear simulator that takes into account high order perturbations when simple STM with J2 is sufficient for our level of resolution. Furthermore, there may be little need to perform absolute navigation/controls, especially if the ability to perform relative navigation/controls is faster and more accurate.

We also learned just how important it is to model noise, and how much noise can affect the performance of both navigation filters and controllers. Precise and accurate measurements are far more important than the complexity of the navigation or controls algorithm you use. On the topic of noise, we also learned that it is always best to include model uncertainty in figures so it is possible to discern between signal and noise.

We also learned how to handle a large code-base in a group. Specifically, over the course of the multiple assignments we increasingly abstracted our code into functions and continued to improve those functions.

Lastly, we learned the importance of data presentation. Proper plotting and visualization enhance one's understanding and intuition of data or a problem. Features such as grid-lines should not be overlooked.

We also learned enough about this topic to feel competent enough to go out into industry, or research, and implement what we have learned.

### 9.4.4 Future Work

The most obvious continuation to this project is extending all of our work to multiple deputy satellites, as Starling actually does. Another potential continuation would be to use relative knowledge to estimate absolute orbital elements of the chief (we know this is what ARTMS on Starling will be doing, so it would be interesting to also try to do it ourselves with our current software). Another obvious future work would be to edit the measurement model to use optical measurements (bearing angles), rather than absolute GPS measurements, since this is the StarFOX technology demo.

Beyond that, more potential future work could be: collision avoidance considerations and analysis during reconfiguration maneuvers; attempt to perform further reconfiguration maneuvers to switch the lead and following spacecraft, which may be necessary to balance propulsion usage; designing a new, more

complicated controller that better can handle disturbances and ringing in steady state error, perhaps by adding some sort of derivative control on top of the proportional-based Lyapunov controller; and explore the robustness and performance of centralized vs decentralized navigation algorithms for a fully formed formation.

A final fun continuation would be to try to get the Starling formation to fly in new and/or more complication orbits, such as lunar orbits, halo orbits, or interplanetary orbits.

## 9.5 Acknowledgements

Thank you to Professor D'Amico and TAs Shane and Mason. We cannot express how much cutting edge technology we have learned in this course, and how valuable that knowledge will be in our future endeavors. Thank you for providing us with all your expertise, and guiding us through this incredible course.

## References

- [1] NASA, edited by Loura Hall, *What is Starling?*, Aug 3, 2022, [nasa.gov/directories/spacetech/  
small\\_spacecraft/starling/](https://nasa.gov/directories/spacetech/small_spacecraft/starling/).