ASEN 6080 Final Project

Sai Chikine

Due May 4th, 2018

Contents

Introduction

Orbit determination has a long and storied history, beginning with Carl Friedrich Gauss in 1809. Since then, humanity has come a long way, demonstrating our capacity for both goodness and for malice. One set of tools, that has perhaps been borne out of both of these two extremes, is the field of statistical estimation. This project explores the nuanced consequences of attempting to use these tools to determine the state of an unknown spacecraft which moves seemingly accordingly to the whims of a dark and vengeful god. Truly, this will be a foray into dark depths. However, we hope to come upon results worthy of hundreds of years of human curiosity.

This project consists of 4 parts. The first part aims to construct a system of describing the motion of man-made spaceborne objects (the system dynamics). The second part of the project consists of estimating the state of a spacecraft doing a flyby of Earth using given data, while having some 'truth' reference data to work off of. The third part also involves estimating the state of a spacecraft doing a flyby with supplied data. However, no part of the true trajectory is known. In parts 2 and 3, the final products are B-Plane parameters. The B-Plane is a plane that contains the focus of the (hyperbolic) orbit, and is also perpendicular to the incoming hyperbolic velocity asymptote v_∞. Finally, the fourth part involves an extra bit.

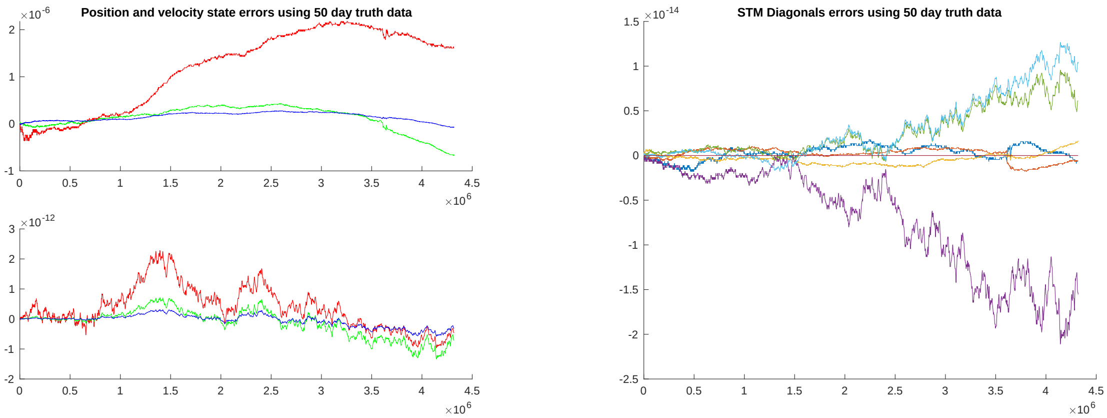
1 Part 1

This part essentially consists of setting up the various modules comprising the system dynamics. This includes the integrator setup and creating station data. The dynamics modeled included gravity contributions from the Earth and the Sun, and solar radiation pressure (SRP). Note that the state being estimated here is at least the usual state (position and velocity components) as well as C_R, which is a parameter used in the SRP model.

2 Part 2

2.1 Dynamics Verification

To verify the dynamics used, the provided truth initial state was propagated forward using the aforementioned dynamics. Since truth data was provided, state errors could be calculated for this time period. They were, indeed, calculated, and are shown in Figure 1.



(a) State errors over 50 days.

(b) Errors in diagonal elements of STM over 50 days.

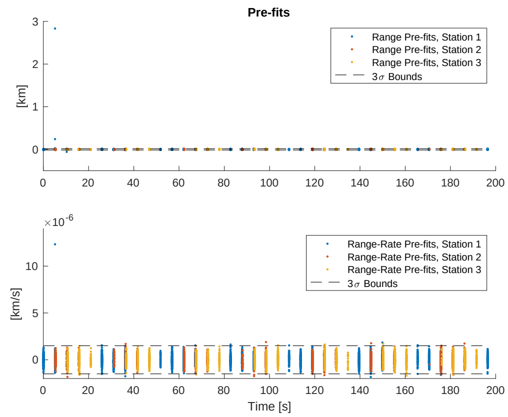
Figure 1: Dynamics comparison with provided truth values.

The errors were deemed small enough to be acceptable, and therefore the model was verified.

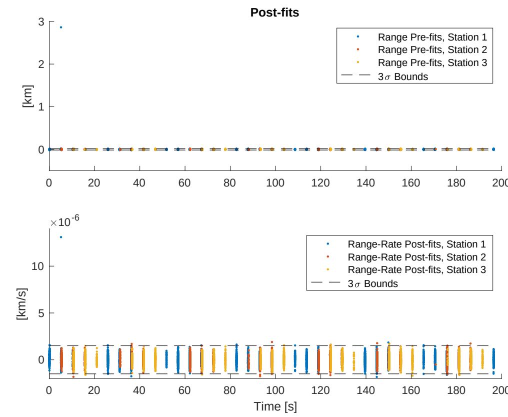
Interestingly, group-mates seemed to be seeing differently results for these. Ultimately, identical code was run on one another's computers, and produced different results, indicating some interplay with the specific processor architecture here.

2.2 Filtering

The goal of this section was to process available DSN measurements in order to obtain a B-plane parameter estimate. The task of processing these measurements was bestowed upon the unscented Kalman filter. The UKF was chosen for its ease of implementation and elegance. The DSN data was processed with the suggested covariances and initial condition. The full range of the filter pre-fits and post-fits are shown in Figure 2. Note that there is one outlier in both the range and range-rate. This one data point is isolated and does not seem to negatively affect the performance of the filter at all. Therefore, this point was discarded. The revised filter residuals are shown in Figure 3.

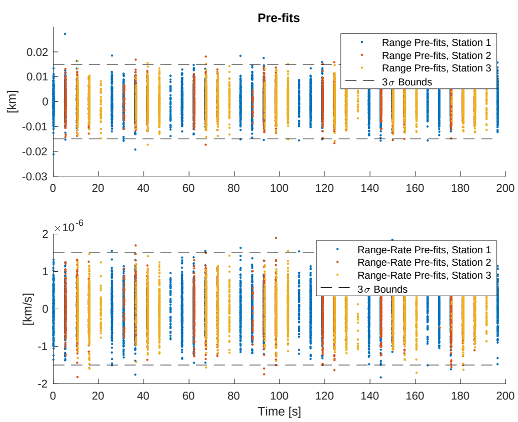


(a) Full range of UKF pre-fit residuals.

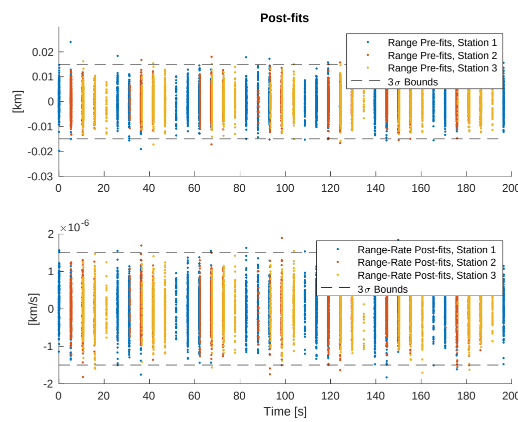


(b) Full range of UKF post-fit residuals.

Figure 2: Full range of UKF pre and post-fit residuals. Note the single outlier.



(a) UKF pre-fit residuals excluding outlier.



(b) UKF post-fit residuals excluding outlier.

Figure 3: UKF pre and post-fit residuals excluding the one outlier.

With this one point removed, we can see that all the filter residuals are within the noise. **However**, this is **not** enough to declare that the filter is working properly. Since C_R was estimated as a parameter, it is possible that the filter absorbed errors into this parameter and allowed it vary over time to fix errors. Normally, we would expect C_R to be constant. Therefore, the proper way to check this filter would be to find the estimate at the epoch, fix any estimated parameters that were meant to be constant, and propagate the estimate forward. We should then recompute post-fits along this propagated trajectory (without any filtering). If *these* are within noise, we can be very confident that our filter is working properly.

When the above process is performed, the propagated post-fits shown in Figure 4.
We can see that these residuals are all within noise as well, indicating that our filter produces an estimated trajectory that is accurate to within our ability to obtain measurements.

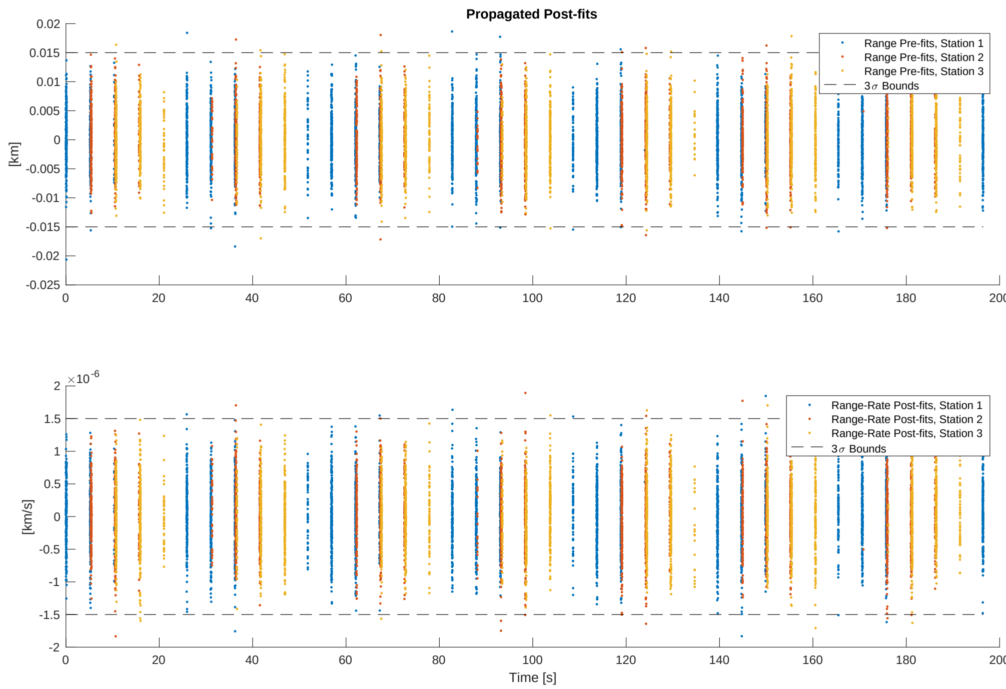


Figure 4: Post-fit residuals obtained by propagating the epoch estimate without filtering.

2.3 B-Plane Parameter Estimation

The ultimate goal of this section is to estimate B-plane parameters. To do this, we find when the distance to Earth (the central body, and the body we are performing a fly-by of) is equal to 3 times the sphere of influence radius, denoted $3RSOI$. In this case, the end of the measurement data (DCO) corresponds to a point in the trajectory before $3RSOI$. Therefore, the orbit is propagated to the $3RSOI$ point using an ODE event to check for when the spacecraft's distance is equal to $3RSOI$. The spacecraft distance over time from DCO to $3RSOI$ is shown in Figure 5.

When the spacecraft is at this point, its velocity $\mathbf{v}$ is selected to be v_∞, which is the hyperbolic excess velocity vector. This vector defines the B-plane.

From here, an alternate method of calculating B-plane parameters was used. We are primarily interested in the vector $\mathbf{B}$, which is the spacecraft position projected onto the B-plane expressed in B-plane coordinates. This vector can be expressed as the vector sum of the spacecraft position vector and some scalar multiple of the vector $\mathbf{v}_\infty$. Let s denote this scalar. Then, s is given by:

$$s = \frac{\mathbf{r} \cdot \mathbf{v}_\infty}{v_\infty^2}$$

The B-plane parameters are then given by $\mathbf{B}$ projected onto the in-plane axes of the B-plane coordinate system.

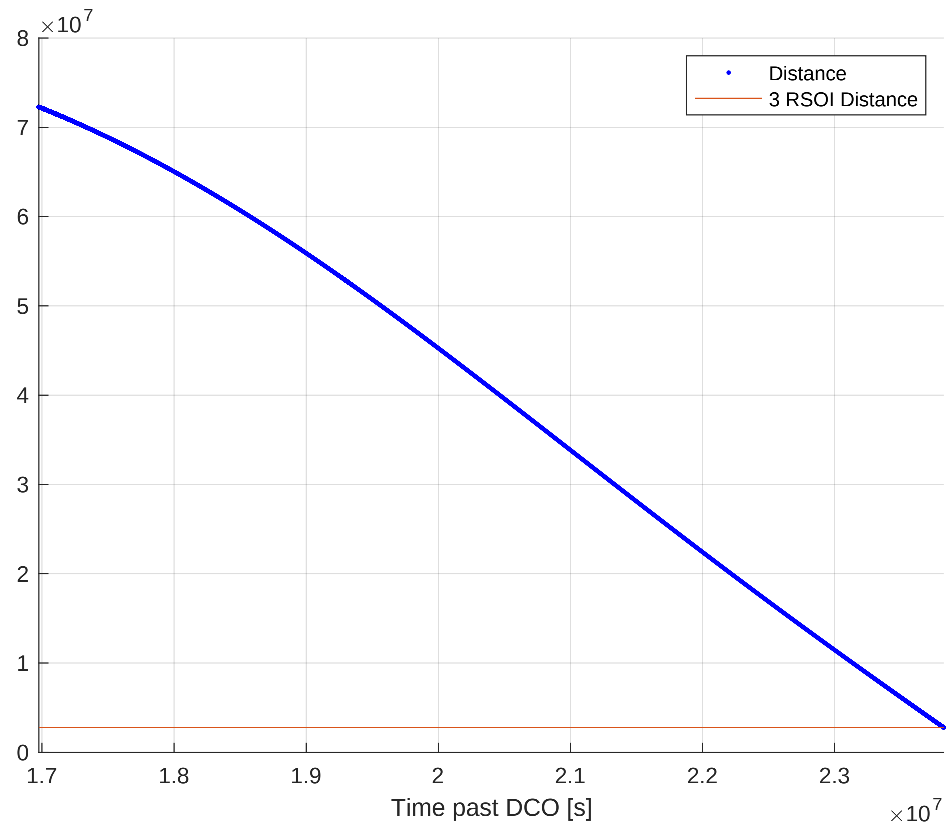


Figure 5: Spacecraft distance after DCO.

In order to get a sense of the uncertainty in the B-plane parameter estimates, the covariance must be propagated to the B-plane as well. To do this, another event was set up for integration, and the state including the STM was propagated to the B-plane. With the STM at this time, the covariance matrix P at DCO can be propagated to the B-plane. Note that we must also rotate this from EME2000 coordinates into B-plane coordinates. From here, the $\hat{T}$ and $\hat{R}$ diagonal components of the P matrix define the uncertainty ellipses on the plane.

The estimates using 50, 100, 150, and 200 days of data are shown in Figure 6, along with their associated uncertainty ellipses. A close up view of the later estimates is shown in Figure 7.

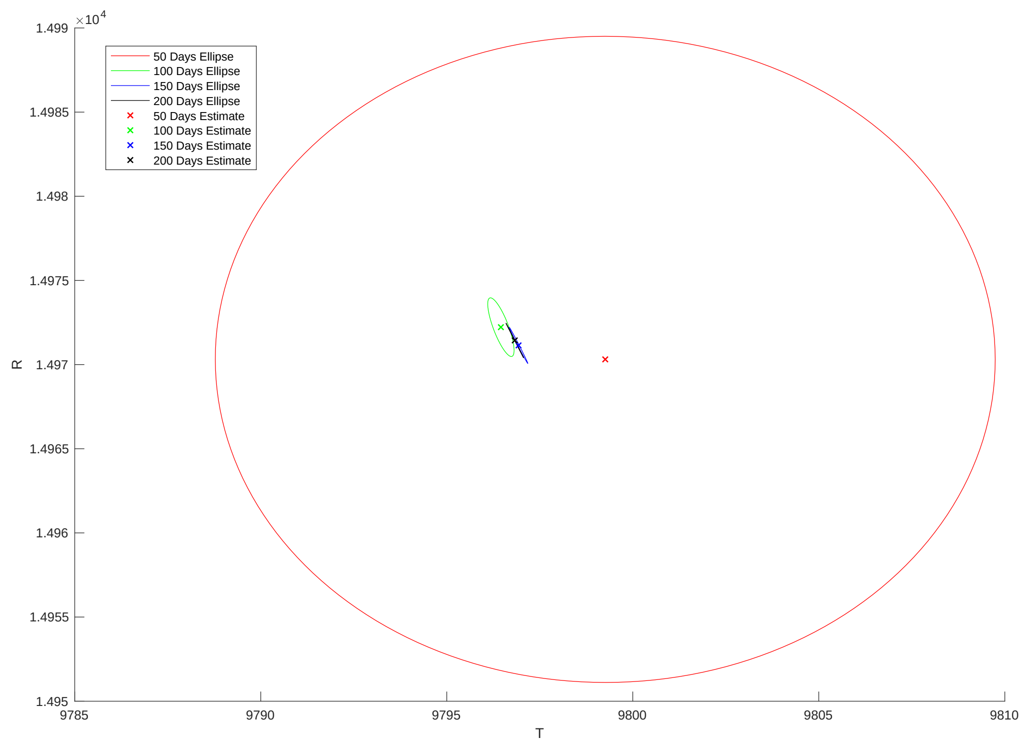


Figure 6: B-plane estimates based on different amounts of data along with corresponding uncertainty ellipses.

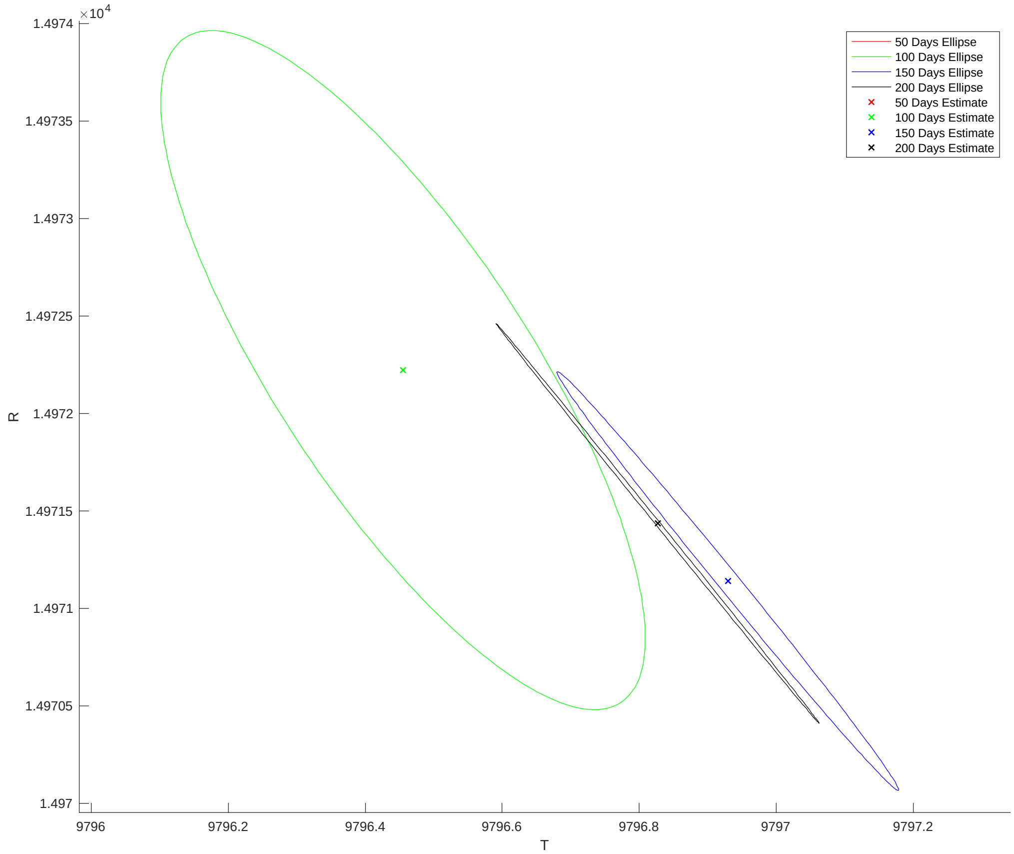


Figure 7: Scaled-up view of the later estimates.

The final B-plane estimate obtained was:

$$\mathbf{B} \cdot \mathbf{R} = 14971.436282823\ km$$
$$\mathbf{B} \cdot \mathbf{T} = 9796.82679372762\ km$$

These values differ from the given (truth) values by:

$$\mathbf{B} \cdot \mathbf{R} \text{ error } = 0.612282823045462\ km$$
$$\mathbf{B} \cdot \mathbf{T} \text{ error } = 0.0897937276222365\ km$$

These seem to indicate that both the filtering and B-plane computations were done correctly.

3 Part 3

At last, we arrive at the cunningly disguised wolf in sheep's clothing. This section involves estimating the state of a spacecraft with unknown truth data and weird dynamics.

3.1 The Naive Attempt

To get an idea of the issues surrounding this problem, the UKF was used to process the new data. As before, the spacecraft state and C_R were estimated using the recommended initial conditions and covariances. The entirety of the data was processed. The filter post-fits are shown in Figure 8.

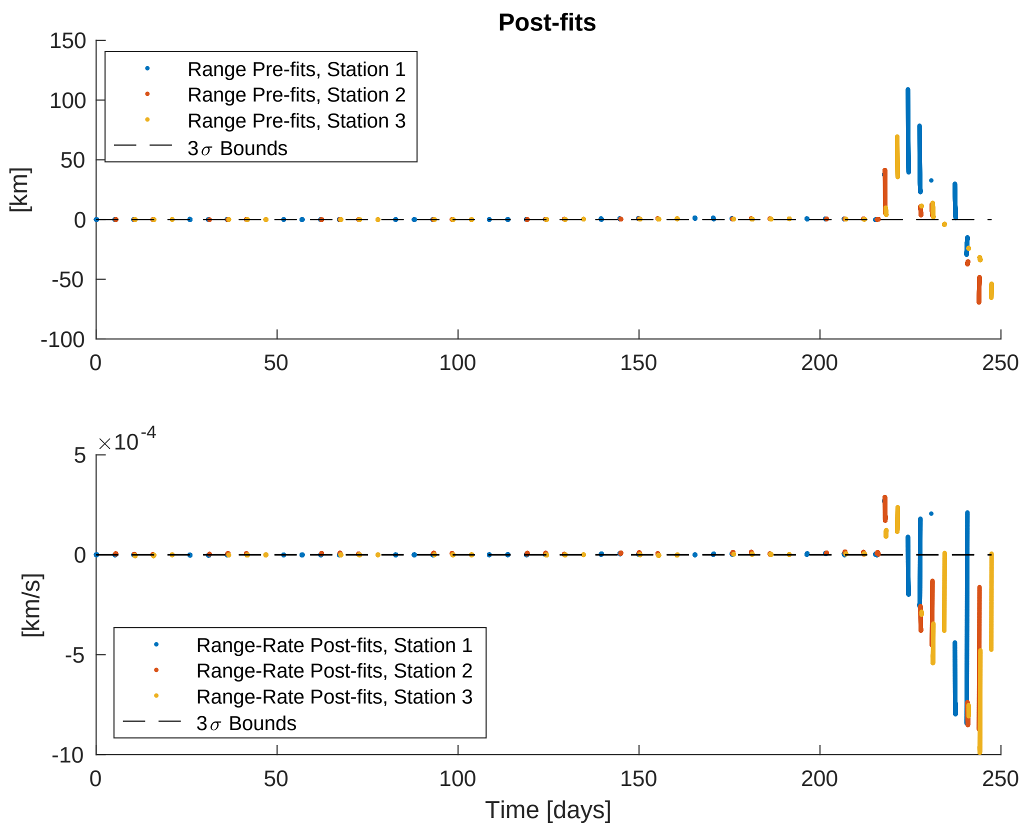


Figure 8: Initial attempt at processing the data.

Immediately, we see very large errors later in the data. If we look closer, we can see some more intricacies, as shown in Figure 9.

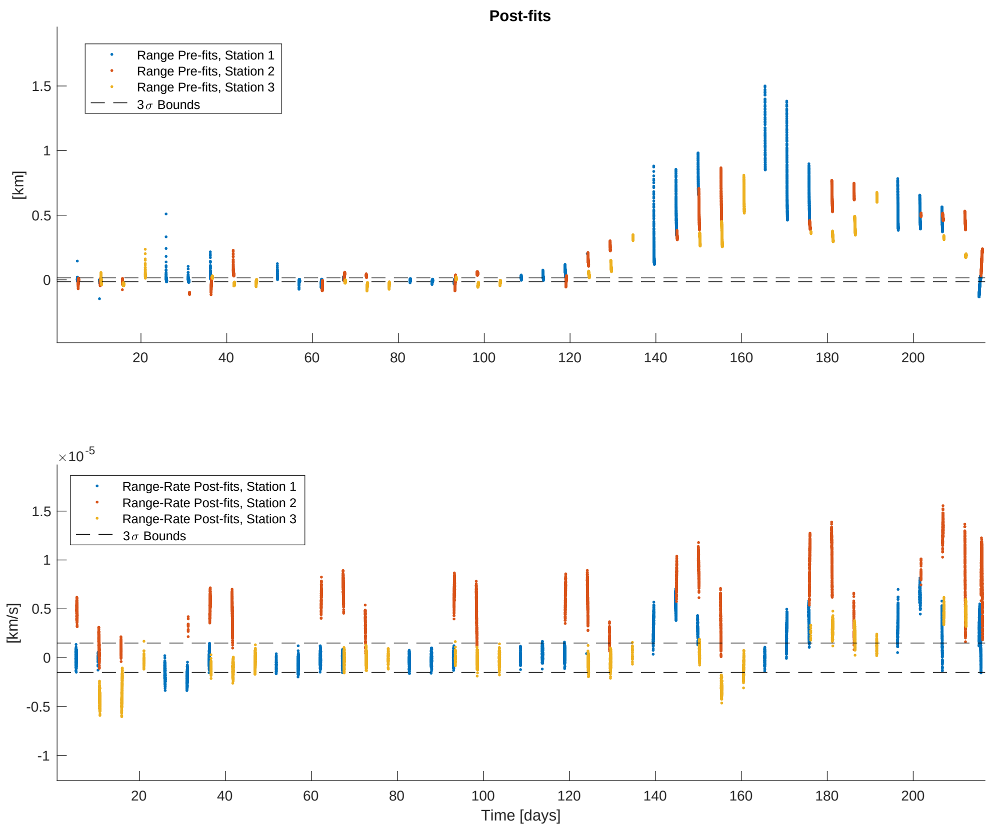


Figure 9: Closer view of naive attempt at processing the data.

Here, we can see smaller, but still significant errors.

The large discrepancy seems to indicate a maneuver taking place. In the time before this event, the smaller errors are harder to diagnose. In any case, the presence of patterns (certain stations always being off in the same way) suggest there are dynamics not being taken into account here. Therefore, the first step taken was attempting to estimate more parameters.

3.2 Pre-Maneuver Data: Station Position Estimation

We first assume that there are no maneuvers before the obvious one. Therefore, we split the trajectory up into two parts: before the maneuver and after the maneuver.

At first, only station positions were estimated. Since the most weirdness was seen in Station 2, it was given the most freedom to move while the other two were held fixed via a small initial covariance.

Although these results managed to improve on the naive attempt, they were not great. Filter post-fits for this attempt are shown in Figure 10. Note that only 100 days of data are shown.

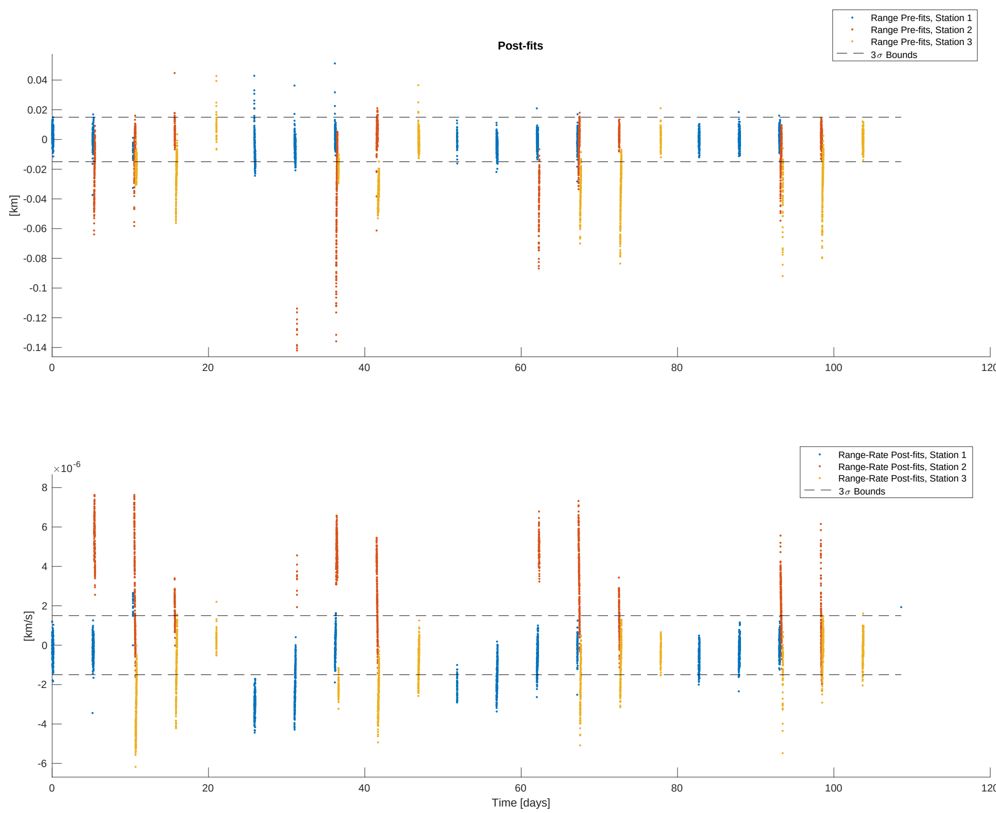


Figure 10: Filter post-fit residuals for the first 100 days of the day

We can see that there are still some errors occurring in a pattern. This might indicate that there might be some other parameters that need to be estimated. Additionally, adding station position estimation to the UKF resulted in numerical issues with the `sqrtm` function in MATLAB. Initially, these numerical issues resulted in filter divergence over time. Upon closer inspection, it was found that small complex components were introduced. Wrapping the necessary calculations in `real()` was able to remove these small components before they could grow. However, this also resulted in small numbers of isolated outliers. These were assumed to be numerical artifacts and were disregarded.

The next step was to attempt to estimate more parameters.

3.3 Pre-Maneuver Data: Estimating More Parameters

In this attempt, all station states were estimated (albeit some with very small covariances), as well as μ_S, μ, E, ω_E, and C_R. Additionally, since some station data points showed strange issues that would not be able to be explained by an inaccurate position, it was theorized that stations could have an additive bias in both range and range-rate. These were two extra constant parameters per station that were estimated as well. In total, the state vector in this attempt consists of 25 elements.

Initially, all station biases were given room to move an appreciable amount. However, this experiment quickly gave poor results. This could be due to observability issues, since a bias in addition to station movement could result in the filter attempting to do two things that have very similar effects. After realizing this, station biases were held at close to zero except for one station.

Since Station 1 seemed to consistently give good results, it was chosen to be fixed. Station 2 seemed to show issues indicating an error in position, so it was allowed to move a bit. With these two measures in place, Station 3 showed small errors that exhibited a clear pattern, so it was given a non-negligible bias.

Now, with Station 1 fixed, Station 2 given σ_0 of $10\ km$, and Station 3 given range and range-rate biases with σs of $1e-3$ and $1e-6$, respectively (to match the order of magnitude of the residual errors that were seen beforehand for Station 3), results looked much better.

However, the small hump at around 100 days leading into the maneuver was still present. Attempting to increase the uncertainty in the station positions and biases resulted in awful results for the early section and did not manage to correct the hump. This presented a conundrum for quite a while.

3.4 What Now?

At this point, after trying for many, many hours to vary different starting values or tweak where exactly to put `real` in the filter (placement affected results subtly), it became necessary to examine more closely what exactly was going on. A look at the time history of the parameters resulted in a moment of revelation.

Beforehand, it was assumed that most of the parameters we were estimating were meant to be constant, and should stabilize to their true values after a while. In other words, it was assumed that time-varying parameters were not allowed. However, the time history revealed that the filter had stopped changing most of the parameters by the time the hump appeared, and had no real degrees of freedom to change the estimate.

Since most of the parameters should, physically, be constant (such as μ_E, μ_E, station positions, etc), it was deemed nonsensical to artifically inflate the covariance for all of them to allow them to vary. Doing this would give the filter its required authority and produce nice filter post-fit residuals, but these could be a result of the filter artificially moving and altering parameters to account for errors.

One parameter that could in fact be time-varying is C_R. For example, if the spacecraft began tumbling, or entered the Earth's penumbra, its C_R would change. Spurred by this revelation, a form of state noise compensation was used for the C_R component of the covariance. This component was not allowed to drop below some threshold value, thus ensuring C_R could vary over time as needed.

Since results looked great up until the hump, SNC was only turned on around 100 days, and covariances in other parameters were set to be small.

So, the scheme at this point is to estimate parameters until the point of the hump, and then use SNC to model a time varying C_R. Filter post-fits using the combination of the two methods discussed are shown in Figure 11.

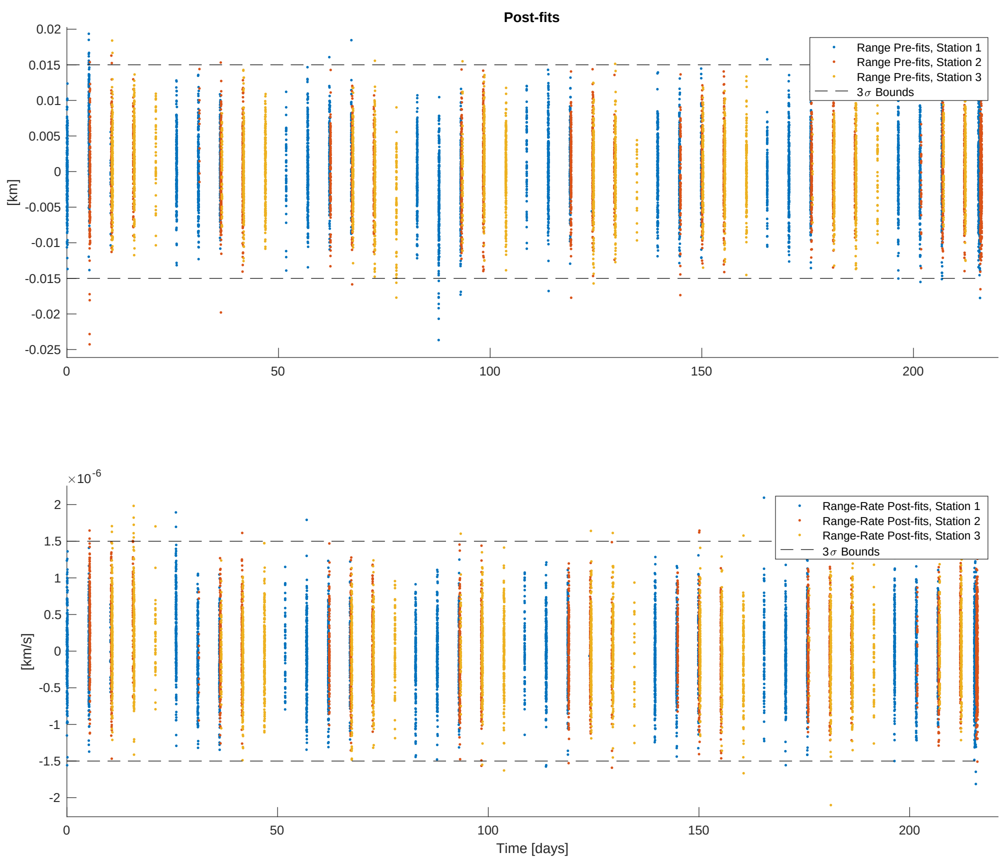


Figure 11: Results up to the maneuver using the hybrid method described above.

3.5 Maneuver

At this point, we are at the point where it is clear a maneuver occurs. There is a large jump in range-rate (and also range), indicating a delta-v.

One way to deal with this is to use DMC. DMC is very easy to implement with the UKF, since there are no STMs to propagate. We can simply model an additional acceleration vector, which obeys a simple first order ODE decay for its dynamics. Its time constant can be set to model a maneuver and the state propagated and updated as usual. The initial state is constrained to be equal to the end estimate of the previous filter, but the covariance is restarted to account for the large change in the short time period. One important thing to note is that DMC should start to be used *before* the arc where one first sees the maneuver. This is because the maneuver seems to be somewhere during the measurement gap. If DMC is used afterwards, results vary wildly and often result in very high accelerations which cause filter divergence. On the other hand, if 100 or so observations are processed with DMC on before the gap in which the measurement takes place, behavior is much cleaner.

Although this may seem simple, DMC often behaved unexpectedly in practice. For example, using DMC while not estimating any parameters would often result in visually good fits, but, probably due to large outliers, the final covariance was very high. On the other hand, attempting DMC while estimating parameters was *extremely* finicky, and was by far the most time consuming part of this whole task, Eventually, a time constant of 3600s was chosen as it seemd to give good results. Covariances had to be carefully tweaked as well.

One example of using DMC without parameter estimation over a very short period is shown in Figure 12. Although this looks good, the results would often start to look bad afterwards. DMC without parameter estimation was ultimately not used.

Figure 12: DMC used on just the very beginning of the maneuver section.

3.6 Final Filter Post-Fits

The final choice was using DMC while estimating parameters, though covariances and other things had to be tweaked carefully. Though DMC did a good job of estimating the post-maneuver trajectory, the fit was not nearly as good as the pre-maneuver residuals. The patched residuals, including all 3 sections, are shown in Figure 13.

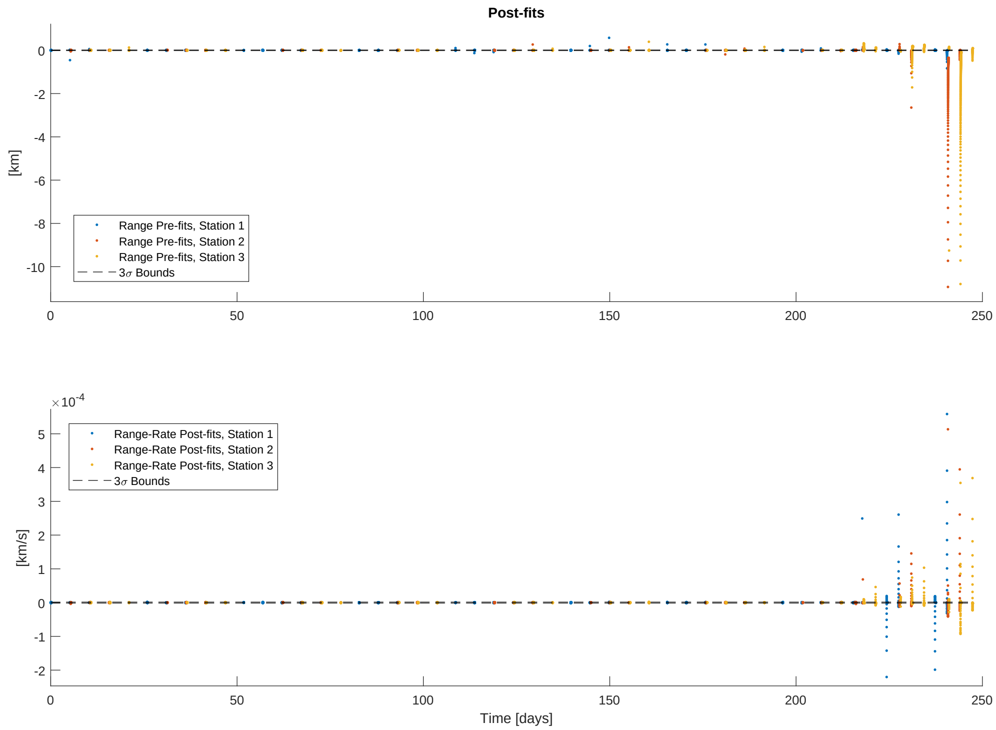


Figure 13: The residuals over the whole trajectory.

Notice that the results look significantly worse towards the end. Although using DMC was able to correct for the maneuver, there were additional problems present near the end of the trajectory that could not be investigated in time. Zooming in a bit, we see the results shown in Figure 14.

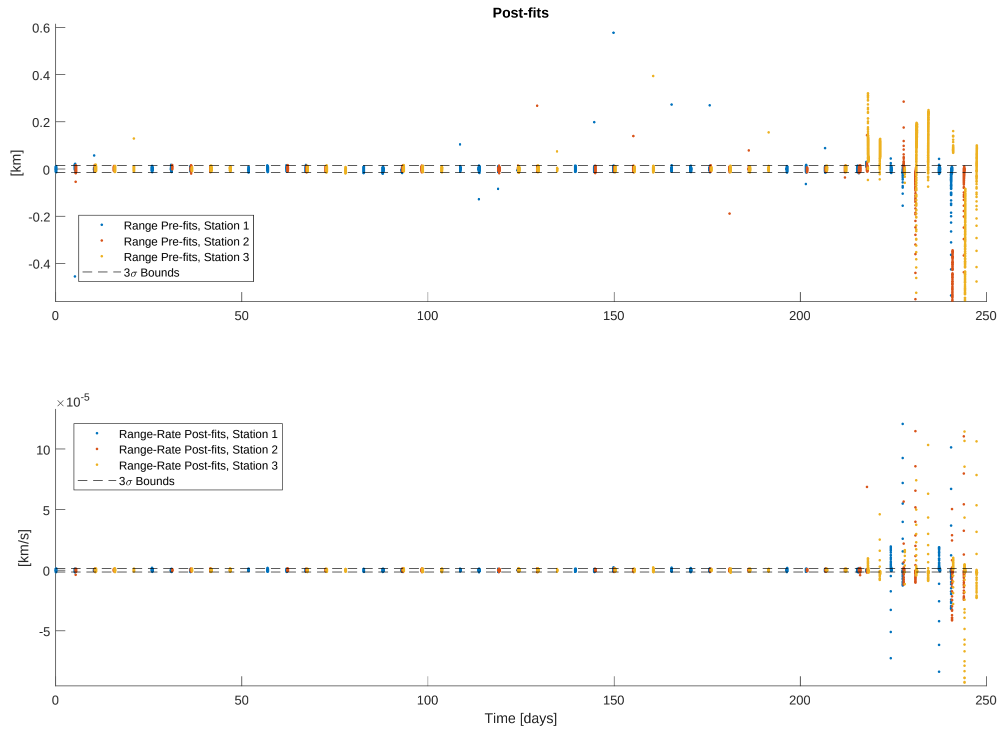


Figure 14: Increased y-scale of post-fits over the whole trajectory.

Again, the pre-maneuver trajectory is fit much better.

3.7 Possible Next Steps in Fitting the Data

Though not all the data was not successfully fit here, there are several approaches one could try in order to conquer this dataset. One way would be to iterate filters. Iterating the UKF in the first segment may have resulted in better pure-propagation residuals.

Another thing would be to more closely examine which parameters need to be time varying. For example, at one point, it seemed like C_R was switching from one stable value to another. Though this may lead to yet another dead end, this kind of behavior might indicate that parameters like C_R need to be manually updated at a certain point in the trajectory. This might be due to something like a sudden change in attitude or entering the penumbra of a planet or other body, suddenly changing the spacecraft's response to SRP.

Another possible thing to look into is the biases. Multiplicative biases for the station measurements were briefly investigated here. However, they did not seem to be much different than additive biases. A time varying bias may be worth a shot, although it may be hard to implement when propagating without filtering.

As mentioned earlier, the second segment of the trajectory (where the hump starts) was modeled using a time varying C_R. Although this was good at producing nice filter residuals, when this final estimate was propagated back to compute residuals without filtering, results were not great. This was interesting, and would seem to indicate that varying C_R with time helped the filter account for whatever dynamics may have been present, but may not have been accurately capturing the true dynamics. This might also be the result of some small amount of outgassing or other propulsive maneuver. It may be beneficial to try DMC here as well. DMC was used for that section as a test, but was (perhaps prematurely) dismissed as unphysical as the more likely scenario was deemed to be some dynamics change.

The last section was interesting in that DMC was able to account for the short period of time where the maneuver took place, but then failed to help with the strange measurements afterwards. It may be helpful to only run DMC for a very short period of time, and then switch back to the regular UKF with parameter estimation. It's possible that adding DMC resulted in observability issues near the end since the filter now had another degree of freedom in addition to all the parameters.

Finally, it may be of interest to look into implementing the square-root version of the UKF. Although the UKF made it easy to implement different things, it did almost always result in numerical issues when estimating many parameters. A square-root implementation that could reduce the bad conditioning number of the covariance matrix may help.

3.8 Notes and Comments

Although filter post-fit residuals looked good, this does not necessarily mean the filter is performing well. Ideally, the final filter state estimate would be propagated back to the epoch, which would then be propagated forwards again to compute residuals. Although the goal was to get to this point, numerous setbacks made this infeasible for the time frame. One example of this type of analysis is shown in Figure 15. This represents the first section propagated back. The filter post-fits in that case looked perfect, but the pure propagation post-fits show that there is still some error here. Ideally, the same would be done for the second and third sections. There, the time varying C_R and DMC accelerations must also be taken into account. This was attempted by using a time varying C_R vector linearly interpolated during the integration for the second section (a zero order hold model was also tried, but worked about the same), but the propagated post-fits were off by amounts on the order of tens of meters for range. This indicates that the trajectory past this section leading into the maneuver will not be incredibly accurate.

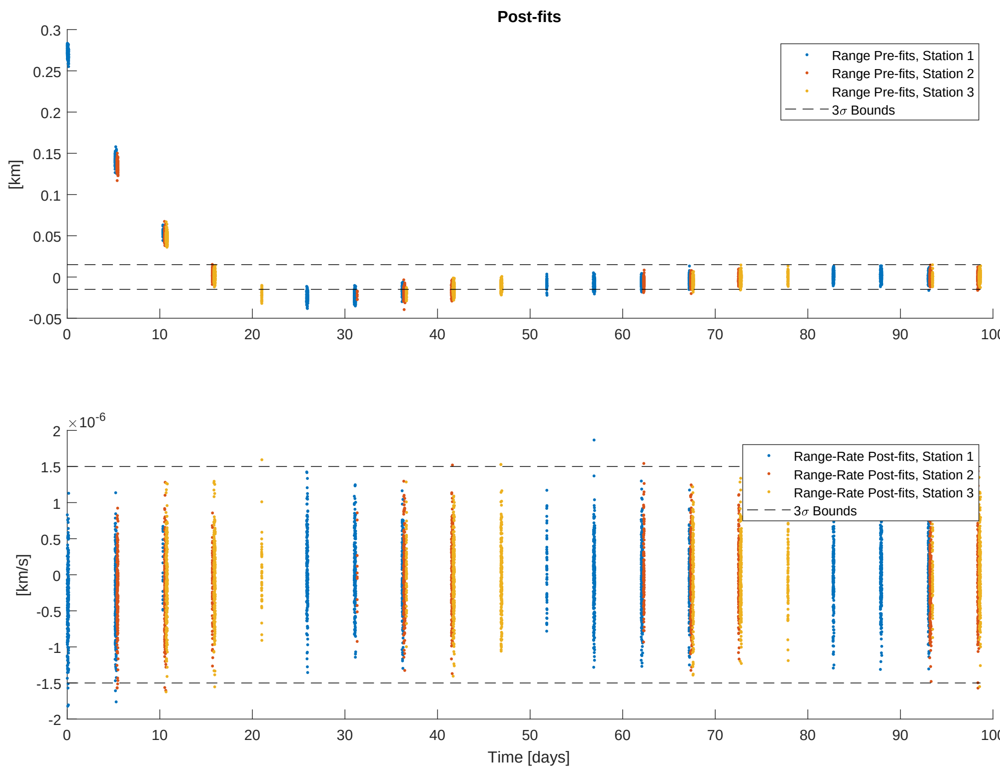


Figure 15: Post-fits computed from purely propagated the filter's predicted state at epoch.

3.9 Estimated State at Epoch

The final estimated state at the epoch was:

$$\mathbf{X}_{\text{epoch}} = \begin{bmatrix} -85775269.6947199 \\ -37761449.4763571 \\ -16358535.3984054 \\ 6.48818541778338 \\ 11.8292812983591 \\ 5.12223180922938 \\ 1.35662400551216 \end{bmatrix}$$

3.10 B-Plane Target Estimate

The B-plane parameters (obtained in the same way as described in Part 2) were:

$$\mathbf{B} \cdot \hat{\mathbf{R}} = 607087.071407091 \; km$$

$$\mathbf{B} \cdot \hat{\mathbf{T}} = 923266.928024003 \; km$$

$$\text{B-plane uncertainty matrix: } \begin{bmatrix} 7250.45719449 & -7215.30315149 \\ -7215.30315149 & 17163.24262546 \end{bmatrix}$$

Note that the method of the s multiplier was used again and not the method detailed in the B-plane handout. Also, while the covariance matrix may have negative entries, a quick check of its eigenvalues will show it is still positive definite, and is definitely symmetric, so it is a valid covariance matrix.

The B-plane estimate along with its covariance ellipse are shown in Figure 16.

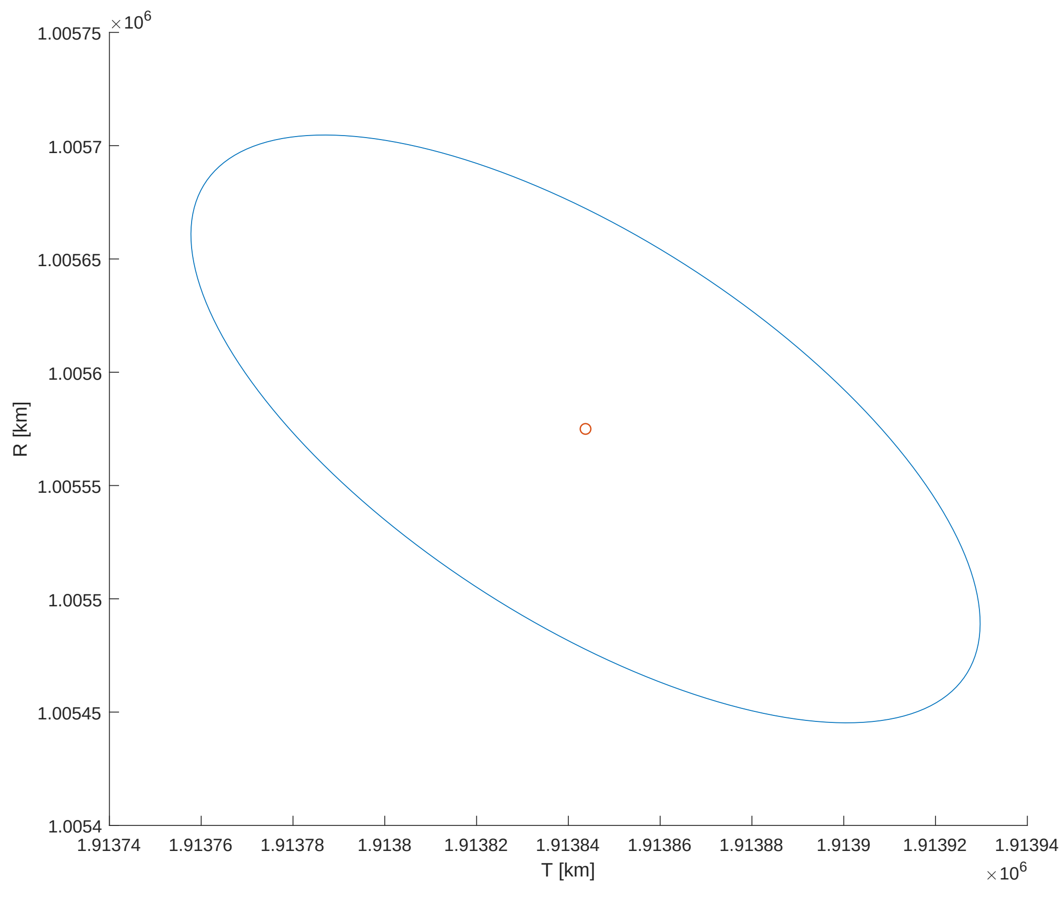


Figure 16: B-Plane estimate for part 3 on the T-R plane with covariance ellipse.

Since the results showed poor residuals near the end of the trajectory, this B-plane estimate is probably not very accurate.

4 Part 4: $RSOI$ Effects on B-Plane Parameters

Changing where we define our velocity vector to be $\mathbf{v}_\infty$ will have a significant effect on the B-plane target. The spacecraft is on a hyperbolic trajectory. Depending on the eccentricity of the hyperbola, a small change in position could result in a significantly different direction of its velocity tangent.

In this case, an integer multiplier of RSOI was considered within a range of 2 to 20. The results on the T-R plane, with covariance ellipses, are shown in Figure 17.

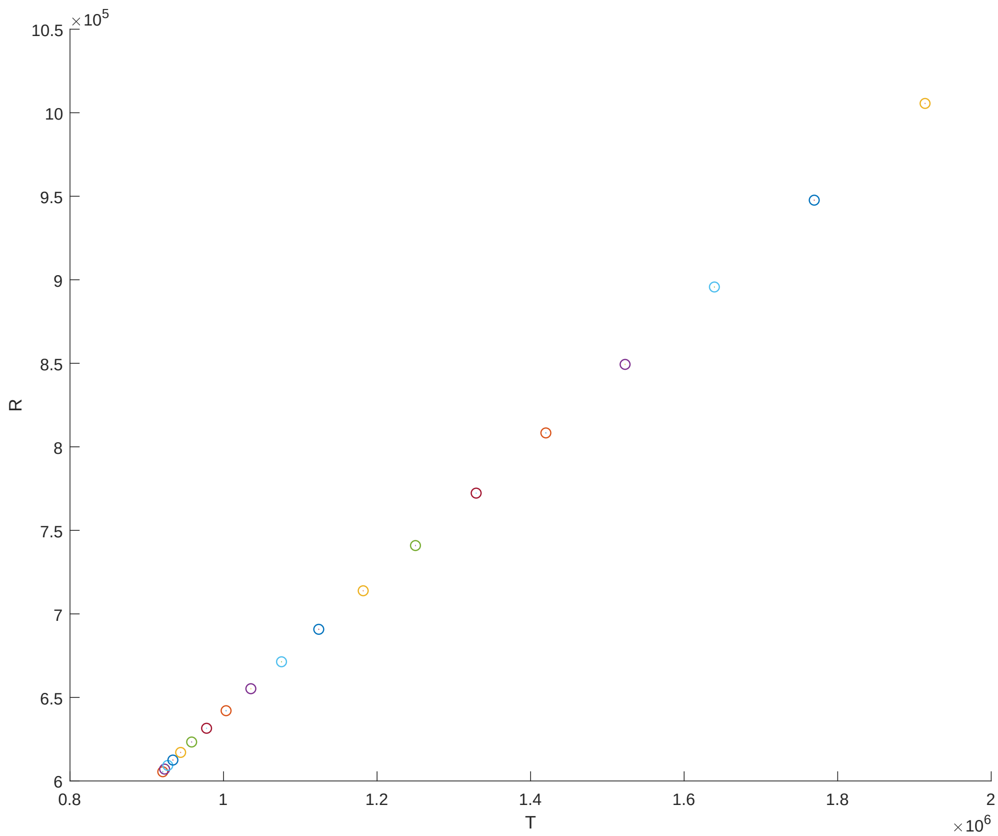


Figure 17: Results on the T-R plane as different multipliers of RSOI were used.

In this figure, the bottom left corresponds to lower RSOIs, and they uniformly increase (at least in this range) towards the upper right. In this plot, the covariance ellipses are too small to be seen. This indicates that, at least with this fit and covariance propagation, changing RSOI by an integer results in no overlap possibility with a different integer RSOI multiplier. Therefore, the choice of RSOI matters greatly. A choice between 2RSOI and 3RSOI would result in B-plane parameters that vary by many kilometers.
As it was increased further, to 50 and beyond, the results split off from the line trend seen here

and began to look like a curve separated from the line off to the left. At very large distances like this, it becomes more difficult to justify calling the corresponding velocity v_∞, so these were not considered here.

5 Conclusions and General Comments

Although the post-fit residuals were able to be fit reasonably well (at least pre-maneuver), it is important to keep in mind that, ideally, estimates would be propagated back and residuals computed without filtering. This is the only way to truly tell how good of a job the OD process has done.

This data-set may indeed have been an example of the *malice* side of the estimation coin. Though victorious against the forces of human reason in this case, battling against it was a fascinating delve into OD beyond the assignments. There was undoubtedly a sense of discovery and investigation, not to mention many late, delirious nights. Although the data was not fit perfectly in the end, it was an amazing learning experience nonetheless, and would definitely make anyone more confident in their estimation abilities if one were to dive in.

6 Acknowledgements

If there is one thing to improve, perhaps, in the future, it ought to be a group project (although there is something to be said for being to verify and match numbers to the 8th decimal place across different codebases). Ansel Rothstein-Dowden and Justin Tyse are great group-mates, and banging heads against the wall collectively is much more satisfying than alone. Of course, shouts out to Professor McMahon and Andrew French for being incredibly helpful and insightful as always.