# 1 Tuning

## 1.1 Thoughts on tuning VP

For the simulated data, the tuning to achieve acceptable consistency seemed to mainly involve $Q$ and $R$,

From the data we sat that the odometry measurements had sampling time of roughly 0.025 seconds. Guessing that the vehicle might accelerate at $1m/s^2$ The tuning was mainly done on the first 5000 iterations, due to constraints in runtime. We first tuned by only looking at the odometry results, trying to get as good fit as possible on the first few thousand samples. The weakness in the odometry appeared to be long turns, where it slowly drifted resulting in an increasing off-set. This then appears to be what the laser measurements must handle. Considering that the odometry appears fairly capable of following most maneuvers, it would seem that the noise required for the laser updates should be fairly small in order to compensate for the odometry drift. Tuning R, with some tadjustment of JCCB$\alpha$s and Q, we made a few observations. R had to be kept below certain values for NIS to stay above the lower bound. Also a too high bearing noise $r_\theta$ tended to introduce sharp changes in bearing. We assume this to be caused by mismatching of landmarks due to the high noise. The JCBB$\alpha$s also affected NIS greatly, lower values reduced the variance in NIS making it easier to fit inside the interval. However low R values along with low JCBB$\alpha$s also made the laser measurements seemingly more unreliable and introduced unwanted offsets leading to large deviations from the GNSS measurements.

- Higher $R_1$ Less landmarks
- Trust odometry to much causes predicted position to change making it harder to make associations
- Avoid detecting same object as different
- Make sure Q is tuned so that P contains reasonable values
- High Q causes initial pose offset that stays uncorrected
- To many landmarks indicate that our position has drifted such that the same landmark is detected again
- To few landmarks indicate that we treat different landmarks as the same, either indicates drift or to high noise
- To high $\sigma_3$ introduces pose error, to low makes the system react to slowly to turns
- Hard to get NIS above lower limit while matching GNSS trajectory

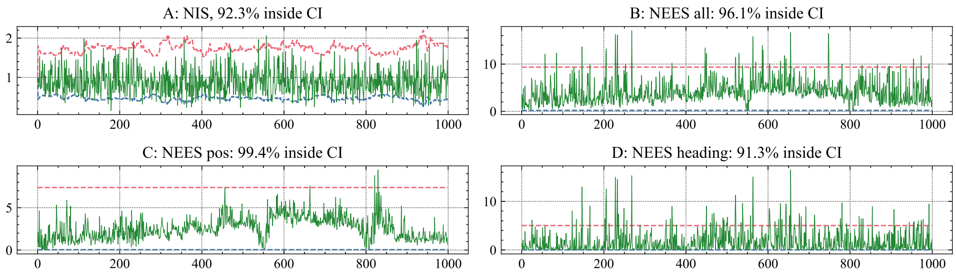


Figure 1: Consistency for simulated dataset

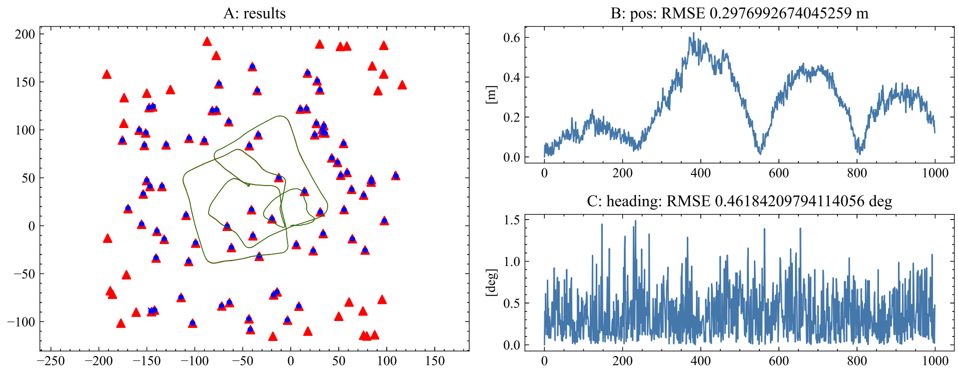


Figure 2: Result and RMSE for simulated dataset

- Smaller JCBB alphas increase variance in NIS and decrease number of landmarks
- Large R, overfit and NIS becomes small, few landmarks

# 2 Consistency

## 2.1 Fit to GNSS

To try to identify any significant offset between our estimates and the GNSS measurements we transformed the GNSS measurements onto our estimates. The optimal rotation was found using SVD,

$$H = \begin{bmatrix} \mathbf{Lo}_{\mathbf{gnss}} & \mathbf{La}_{\mathbf{gnss}} \end{bmatrix} \begin{bmatrix} \mathbf{Lo}_{\mathbf{est}} \\ \mathbf{La}_{\mathbf{est}} \end{bmatrix}, \quad U, S, V = SVD(H), \quad R = VU^T$$

whereas the translation was solved by OLS.

$$T = \frac{1}{N}\sum_{1}^{N}\begin{bmatrix} Lo_{gnss_i} \\ La_{gnss_i} \end{bmatrix} - \mathbf{R}\frac{1}{N}\sum_{1}^{N}\begin{bmatrix} Lo_{est_i} \\ La_{est_i} \end{bmatrix}$$

We observed some transformation when iterating over less data (5-10k iterations). When iterating over more data (20k+ iterations), the optimal transformation became close to negligible, as seen in figure (blabla). Our understanding from this is that any drift is incurred mainly from turning. With proper tuning, the incurred drift is seemingly negated at a later time when turning in the opposite direction or when we are able to correct our trajectory by making correct joins from laser measurements. In other words, when driving around at random, we get a good track. If however this was a NASCAR dataset, we would probably have to tune differently to avoid an increasing offset.

To compare with GNSS we computed NIS by using the odometry predictions as if we would update using GNSS. We paired a prediction with a GNSS measurement by locating the closest GNSS sample to each odometry measurement that was also within 0.01 seconds (chosen by a little trial and error) of eachother.