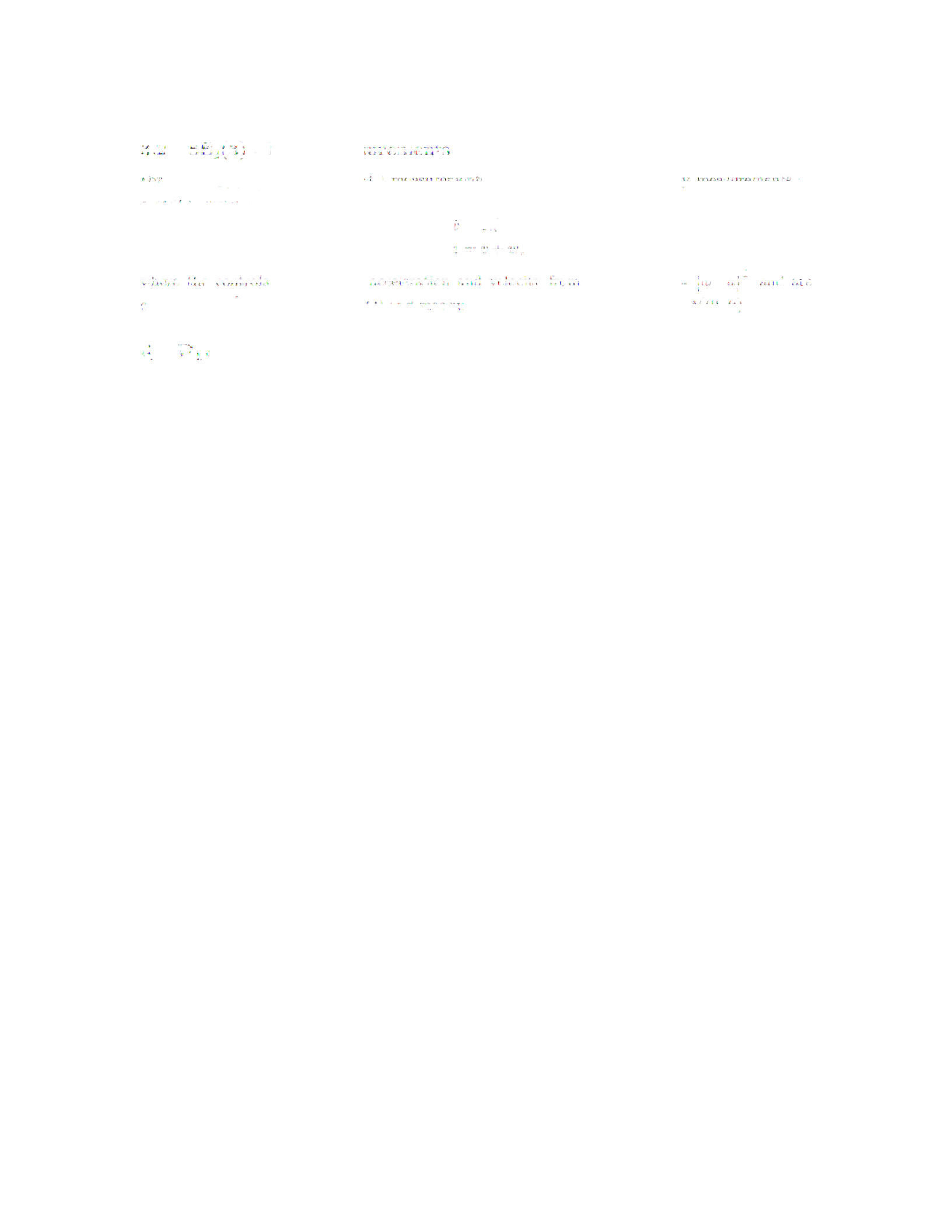

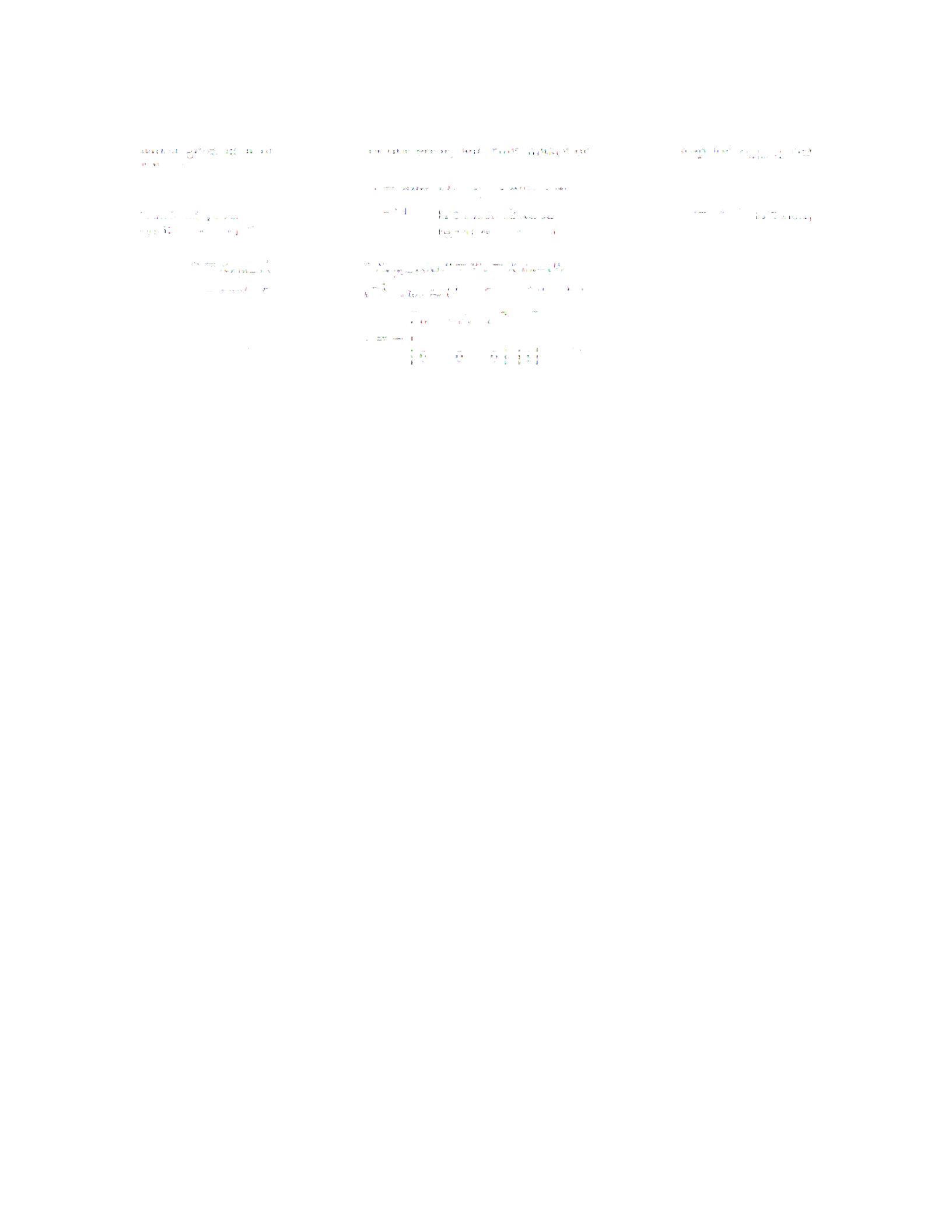

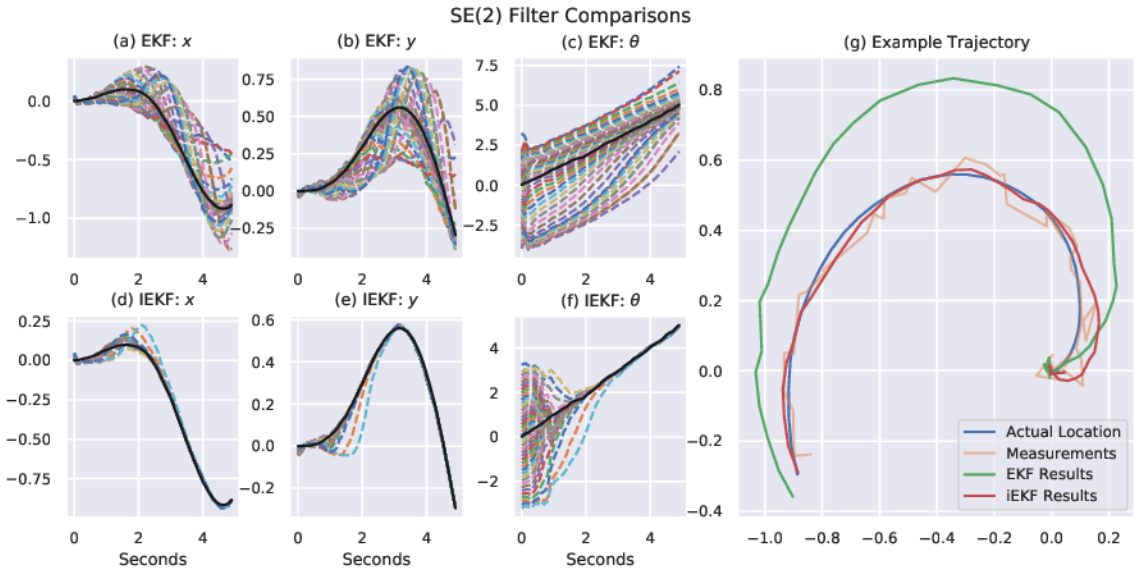

Figure 1: Comparison of the performance of the EKF and IEKF on SE(2) system. The simulation was run a total of 41 times for both filters, with the same system path and measurements, but with varied initial estimates of the system's starting pose. Plots (a), (b), and (c) show the pose estimates, broken down into x, y, and θ respectively, of each of the EKF simulations. The estimates are represented by dashed lines, while the true pose of the system is represented by a solid black line. Plots (d), (e), and (f) are constructed in the same fashion as those above, but represent the results of the IEKF. Lastly, plot (g) shows the path from the system, the measurements, and the EKF and IEKF estimates of the system's path. The same initial estimate was used for both the EKF and IEKF results shown in plot (g).

5.1 Performance of the EKF and IEKF on SE(2)

Using the unicycle model, defined in section 3.1, both the EKF and IEKF were implemented on the system. In order to verify that the IEKF outperforms the EKF, the unicycle path and measurements were generated and then each of the filters were run a total of 41 times. The unicycle's starting pose, represented as (x, y, θ), was $(0, 0, 0)$ and the estimated starting poses of the filters ranged from $(-20, -20, -\pi)$ to $(20, 20, \pi)$ with a step size of 1 for the x and y positions, and a step size of $\frac{2\pi}{40}$ for θ. The simulation results are found in Figure 1 and demonstrate how the IEKF is able to converge very quickly to the model's true position, while the EKF converges much more slowly.

5.2 IEKF Implemented on $SE_2(3)$

The unicycle system is very simple, a more complex system presents a greater challenge for implementing the EKF, and greater non-linearities have a negative impact on its performance. The IEKF is also more difficult to implement on a more complex system, but much less so than the EKF. Section 4 provides a greater evaluation on the specifics of the IEKF implementations for each system.

A quad-rotor simulation was used to generate data for $SE_2(3)$ with a simulated IMU for the measurements. The data included additive noise in the controls, as shown in Section 4. A plot of the quad-rotor's true location, the pathway traced by the measurements, and the pathway that resulted from the IEKF is found in Figure 2. It is evident that the filter continues to perform exceptionally well, even in situations when the predicted path begins turns in the opposite direction

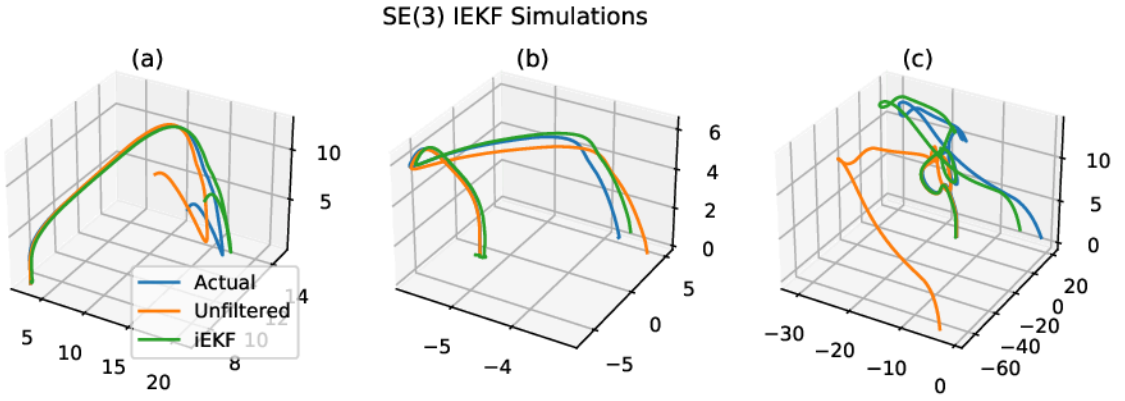

Figure 2: Three separate data sets of a quad-rotor flying, with the true path, predicted path (only using the measurements from the IMU), and the path from the IEKF for each data set. The quad-rotor was free to move about in three dimensional space, so all three plots show the position in coordinates of the form (x, y, z). Plot (a) corresponds to the first set of data, plot (b) to the second, and plot (c) to the third.

of the truth, as is the case in plot (c) of Fig. 2. Note there is still some drift, but this is to be expected since no correctional measures were taken on the position, only on the velocity.

6 Conclusion

Ultimately, the EKF, though widely used, is an approximation that is unable to fully characterize nonlinearities in a system. With the use of an IEKF, those nonlinearities are more accurately characterized by the filter, resulting in an estimate that converges to the truth much more quickly than the EKF, and that overall remains more accurate in tracking the true pose. Additionally, the IEKF is much easier to implement for complex systems, so long as the system is found within the Lie Group. As with the EKF, there are circumstances where an IEKF is not able to converge due to the sensors or measurements used, but it will continue to outperform the EKF so long as there are nonlinearities present in the system.

References

[1] A. Barrau and S. Bonnabel. "The Invariant Extended Kalman Filter as a Stable Observer". In: *IEEE Transactions on Automatic Control* 62.4 (2017), pp. 1797–1812. DOI: 10.1109/TAC.2016.2594085.

[2] Axel Barrau and Silvère Bonnabel. "Invariant Kalman Filtering". In: *Annual Review of Control, Robotics, and Autonomous Systems* 1.1 (2018), pp. 237–257. DOI: 10.1146/annurev-control-060117-105010. URL: https://doi.org/10.1146/annurev-control-060117-105010.

[3] Kalin Norman Easton Potokar. *Invariant Kalman Filter*. 2020. URL: https://github.com/contagon/iekf.

[4] Ross Hartley et al. "Contact-aided invariant extended Kalman filtering for robot state estimation". In: *The International Journal of Robotics Research* 39.4 (2020), pp. 402–430. DOI: 10.1177/0278364919894385. URL: https://doi.org/10.1177/0278364919894385.