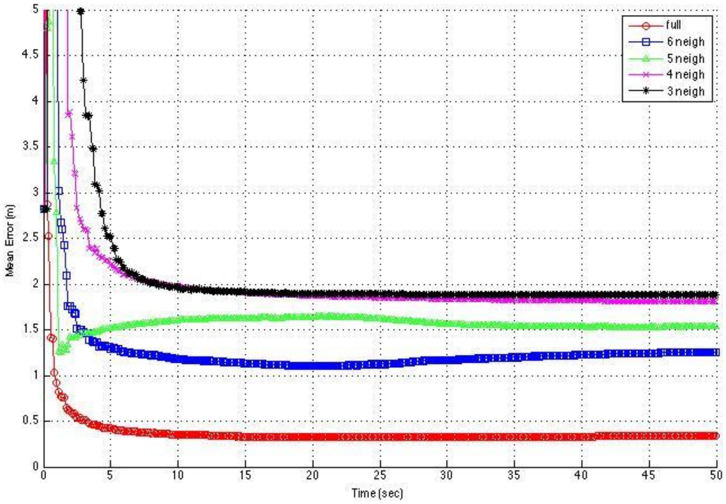

Figure 1: Plot of the mean errors for varying neighborhood sizes (with DICI-OR)

Figure 1 shows how the performance of the filter reduces as the neighborhood size reduces. This is expected as the covariance matrix is being approximated with the use of DICI-OR and the matrix that is being reconstructed using the method may not necessarily replicate the true values of the direct inverses. This is a trade-off between having the centralized implementation and the distributed implementation – while one aims for accuracy and falters when it comes to scalability, the other is highly scalable for large scale IoT application but compromises on the accuracy. The time synchronization error measurements are presented in Figure 2 below.

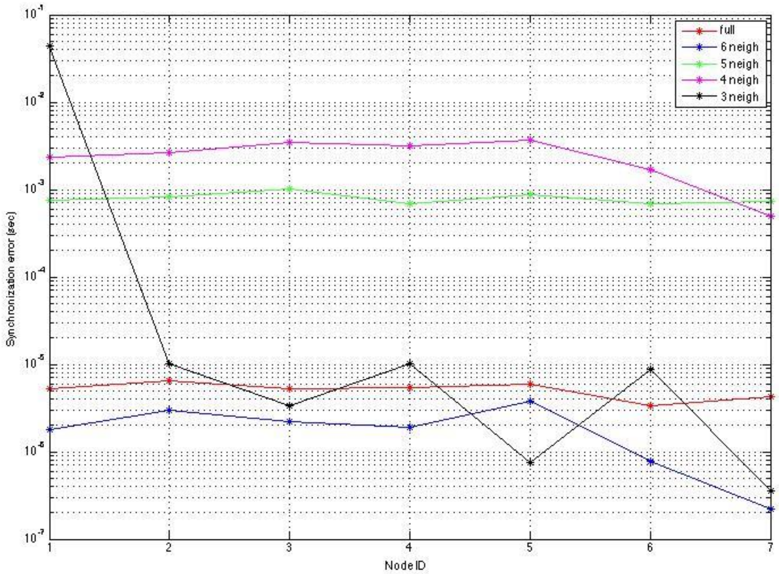

Figure 2: Plot of the synchronization errors for Nodes in the network, with varying neighborhood sizes (with DICI-OR)

We now proceed to compare each of the neighborhood configurations between the different filter types that were employed. As a recap of the notations in the legend appended in the graphs, DOpt is the optimization filter, implemented by Amr. Al Anwar at NESL. The DKal is the diffusion Kalman filter from Prof. Sayed's paper while the DLarge includes the large scale implementation done as a part of the project. DKal still tracks the whole error covariance matrix and state vectors, and is thus to serve as the benchmark for comparing the DLarge algorithm with DICI-OR implemented.

Some of the observations include the how the performance in the distributed algorithm is sensitive to the structure of the graph connectivity between the nodes. If a graph is constructed and the neighborhood defined for non proximal nodes, the performance deteriorates as opposed to the case where the neighborhood is defined as the ones that are physically proximal to each other.

The plots are as below –

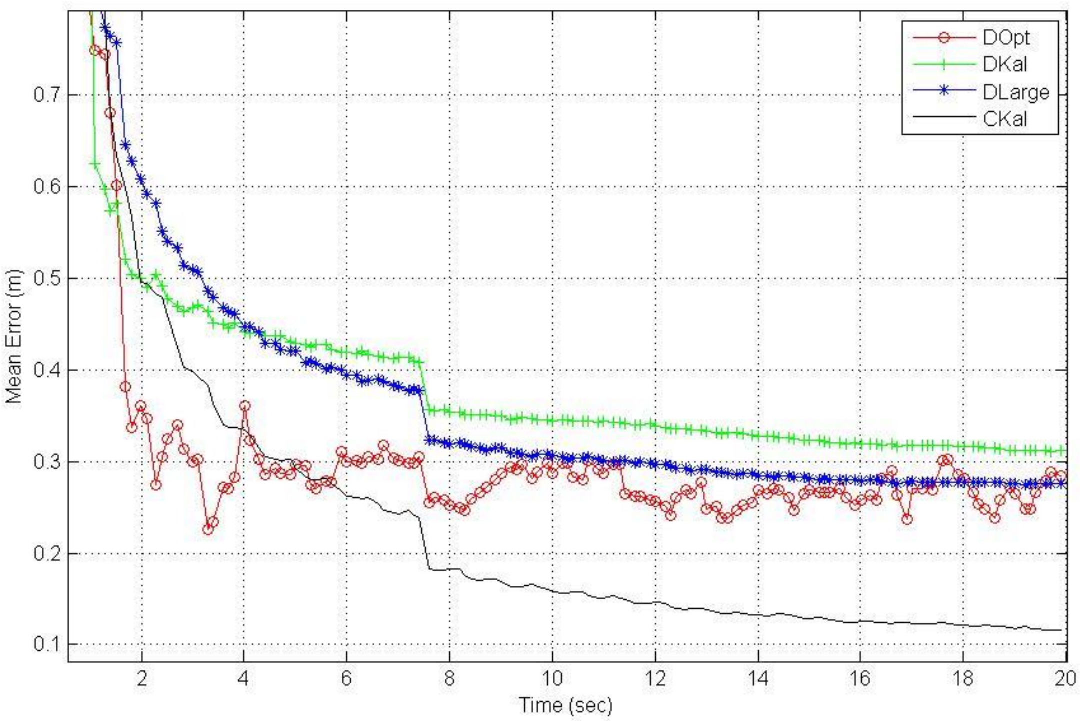

Figure 3: Plot of the localization error – Comparing Optimization filter, Diffusion Kalman, DLarge (DICI-OR) and Centralized Kalman filter in fully connected network.

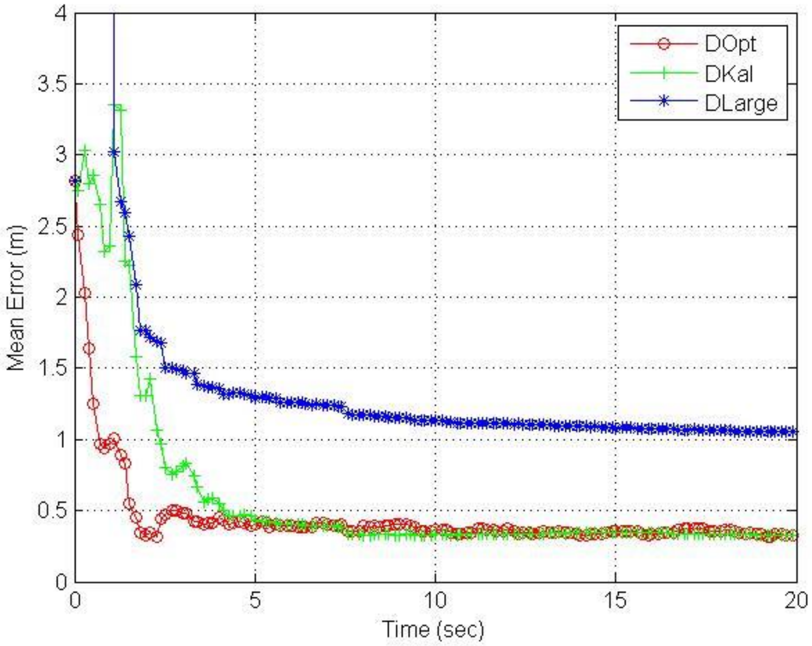

Figure 4: Plot of the localization error – Comparing Optimization filter, Diffusion Kalman, DLarge (DICI-OR) in 6 neighbor network.

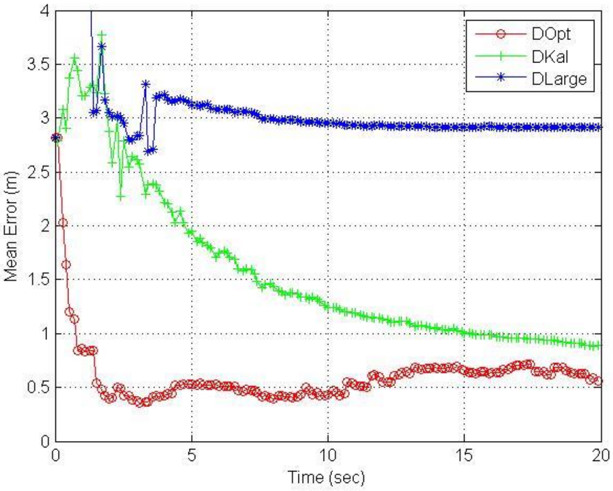

Figure 5: Plot of the localization error – Comparing Optimization filter, Diffusion Kalman, DLarge (DICI-OR) in 5 neighbor network.

The 5 neighbor network here manifests in erroneous estimates due to how the network connectivity is defined. This was done to show the sensitivity of the algorithm to the same. If we compare this to another configuration used for 5 neighbor networks in the overall mean localization error plots (Figure 1), we can clearly see this impact.

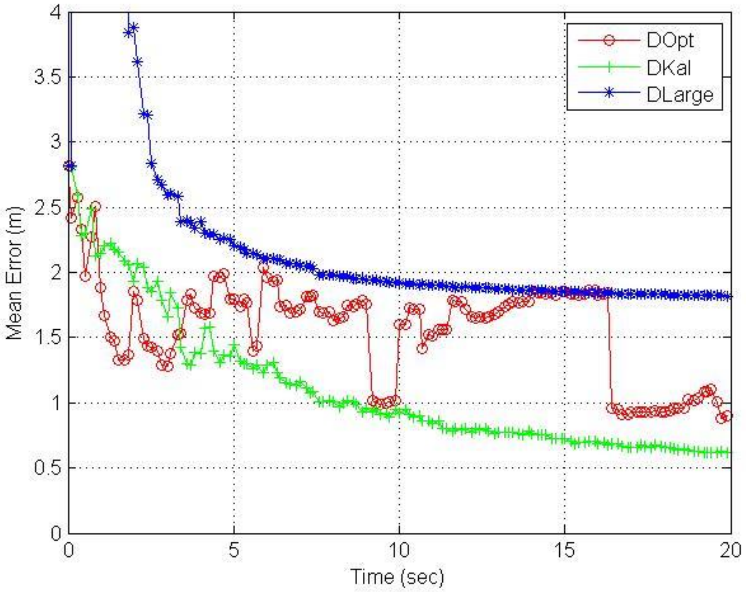

Figure 6: Plot of the localization error – Comparing Optimization filter, Diffusion Kalman, DLarge (DICI-OR) in 4 neighbor network.

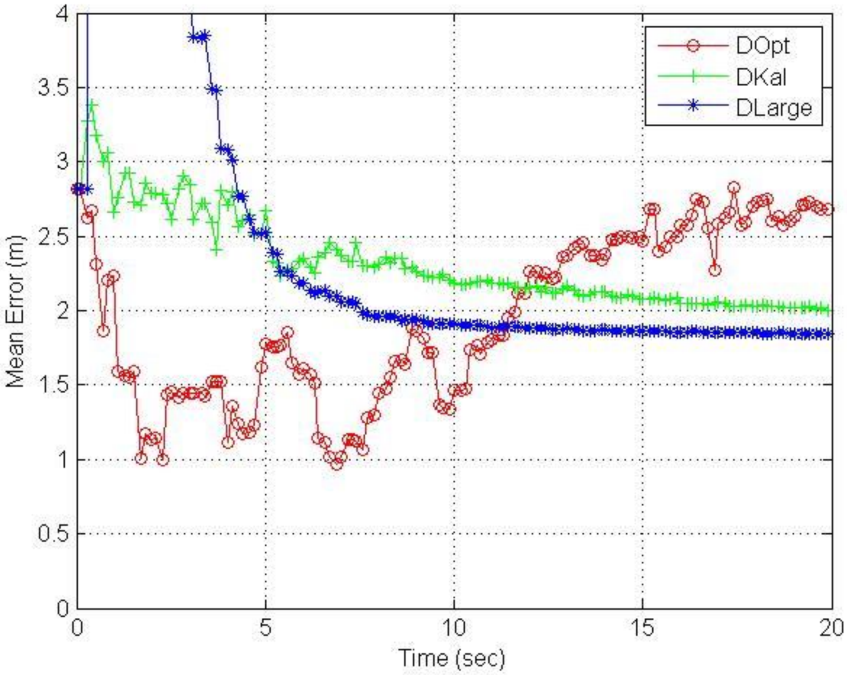

Figure 7: Plot of the localization error – Comparing Optimization filter, Diffusion Kalman, DLarge (DICI-OR) in 3 neighbor network.

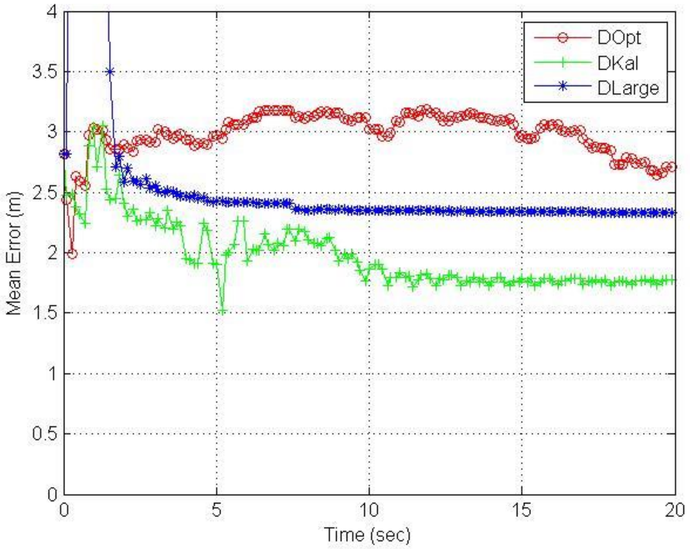

Figure 8: Plot of the localization error – Comparing Optimization filter, Diffusion Kalman, DLarge (DICI-OR) in 2 neighbor network.

We observe varying performance of the 3 filter types in the static nodes case. While some filters have strong performance in higher neighborhood sizes, such as the optimization filter, the performance deteriorates when the neighborhood size has reduced. We now proceed to plot the time synchronization error for different neighboring nodes sizes.

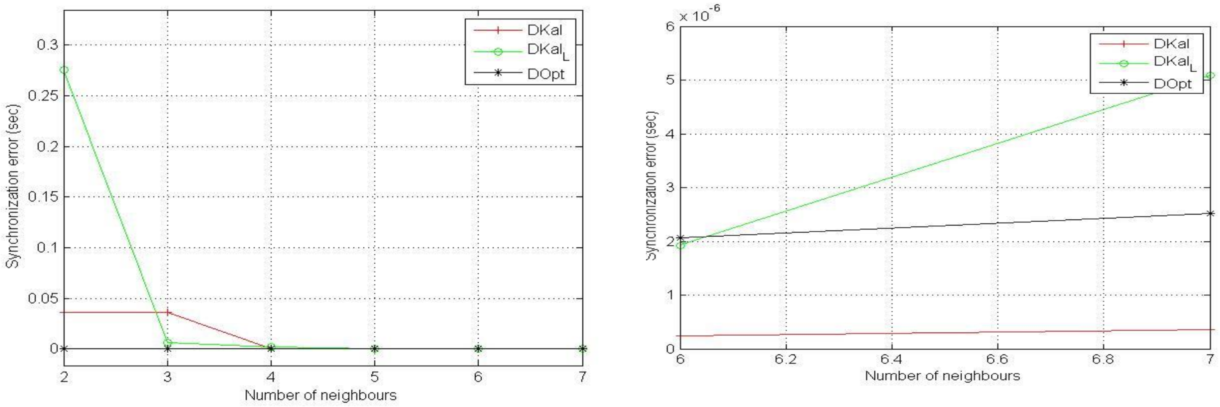

Figure 9: Plot of the time synchronization error – Comparing Optimization filter, Diffusion Kalman, DLarge (DICI-OR) in 2 neighbor network.

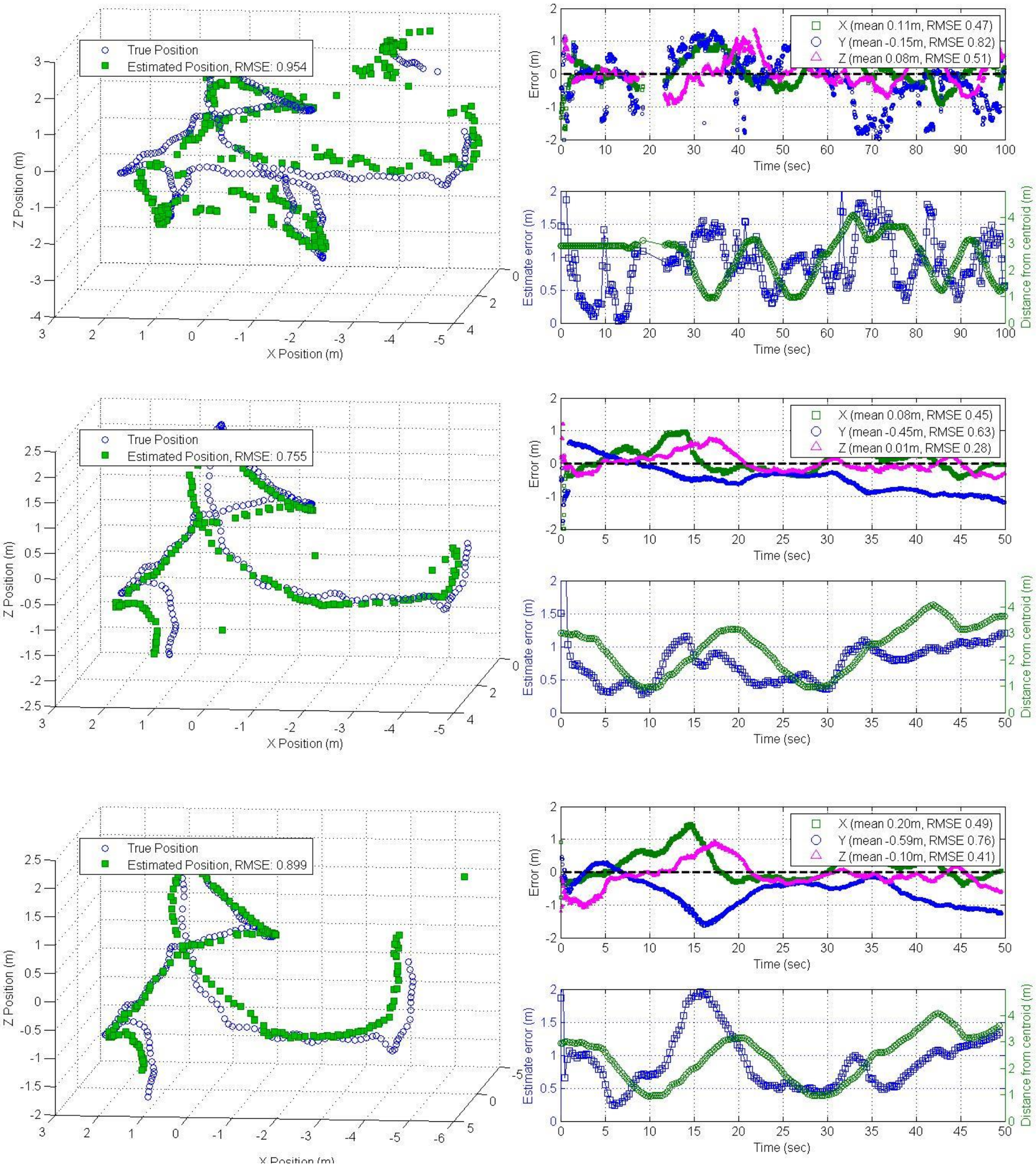

Figure 11: Plot of the mobile node localization error – Comparing Optimization filter (Top), Diffusion Kalman (Middle), DLarge (DICI-OR)(Bottom).