README for "Multiple Model Particle Filter for Traffic Estimation and Incident Detection.".

Ren Wang and Daniel B. Work

June 10, 2015

Abstract

This document describes the implementation of the multiple model particle filter(MMPF) introduced in the article 'Multiple Model Particle Filter for Traffic Estimation and Incident Detection." by Wang, Work and Sowers, submitted to the IEEE Transaction on Intelligent Transportation Systems A preprint of the article is available for download on the second author's website. The source code is hosted at `https://github.com/Lab-Work/MMPF_Traffic_Estimation_Incident_Detection`.

1 License

2 Running the code

The provided .py files can be used to reproduce the results presented in the publication.

1. Download CORSIMOutPutTextFile.zip and CORSIMOutPutTextFile2.zip from the link: `https://uofi.app.box.com/s/6qk5mxn103oemijmj9cpdeu2s5axytxq`
2. Unzip CORSIMOutPutTextFile.zip and copy it to folder: /CORSIM filter factor/
3. Unzip CORSIMOutPutTextFile2.zip and copy it to folder: /CORSIM_FD/
4. Go to the folder: /CORSIM filter factor/PythonCode and running:

   `extract_keyData.py`

   `extract_trajectory.py`

   `generate_measurements.py`

   `generate_trajectory.py`

   `generate_true_state.py`

5. Go to the folder /CORSIM_FD/PythonCode and running:

   `extract_keyData_FD.py`

   `process_keyData_FD.py`

6. To generate figure 2, go to the folder /CORSIM_FD/PythonCode and running:

```
generate_keyData_FD.py
```

7. To generate figure 3, go to the folder/test/MMPF/ and running

   ```
   Main_MMPF.py
   ```

   ```
   change the PR value and lag value to generate results for different penetration rates a
   ```

8. To generate figure 4, go to the folder/test/MMPF/ and running

   ```
   Main_MMPF_PR.py
   ```

   ```
   Error_PR.py
   ```

9. To generate figure 5, go to the folder/test/MMPF/ and running

   ```
   Main_MMPF_inflow.py
   ```

   ```
   change the inflow value to generate results for different inflows
   ```