**Replication package explanatory file**

**Manuscript title:** Efficacy of decentralized traffic signal controllers on stabilizing heterogeneous urban grid network

**Manuscript ID:** COMMTR-D-24-00021

**Authors:** Namrata Gupta, Gopal R. Patil, Hai L. Vu

**Replication package access**:

The necessary code and documentation to replicate the presented study is available on GitHub https://github.com/NamrataGupta5/COMMTR_Replication_package

**Data description:**

The data is generated by simulating the three-bin network's dynamics in Python and microsimulations of three-ring network and grid networks in PTV Vissim 6.0.

1. Data used in Figure 4 is used by running a microsimulation of a three-ring network. The model is in the folder *'Flow within bin and outflow from a bin/MFD_for fixed time_diff signal plan'* (files: 3rings.inpx, 3rings.layx, and 3rings1.sig). Please enter input turn ratios (by opening .inpx file) as 0 to ensure zero interaction between the rings and adapt signal timings as mentioned in the figure. After each simulation, the link segment result file is exported from Vissim and saved as '.att' files.
2. Data used in Figure 5 is used by running a microsimulation of a three-ring network. The model is in the folder *'Flow within bin and outflow from a bin/Proportionality constant for bin outflow'* (files: 3rings.inpx, 3rings.layx, and 3rings1.sig). Please change turn ratios and signal timings as shown in the figures. After each simulation, the link segment result file is exported from Vissim and saved as '.att' files.
3. Data for Figures 6-8 is obtained by coding the bin dynamics (Eq. 1 & 10), equilibria cond (Eq. 11), and stability condition (Eq. 21 & 26) in python. The code is in the folder '*Stability analyses using three-bin*' (file: *3bin_stability_Analyses_Phase_Diagram_MFD.py*). For a given average network density and policy, the code calculates the equilibria points and checks for the nature of equilibria. It stores average density, average volume, type of equilibria, and adaptivity values in a file '.att' extension.
4. Data for Figures 10-12 is obtained by microsimulating a three-ring network. The model is in the folder *'Three-ring network simulation'* (files: 3rings.inpx, 3rings.layx, and 3rings1.sig). Please change the vehicle loading rates to match the demand pattern shown in the figure. The simulation is run with a script file named '*3Ring_MFD_DiffControlPolicies.py*' for each demand pattern, policy, and adaptivity value. The code saves a file with average parameters at the end of the simulation.
5. Data for Fig. 14 is obtained by microsimulating a 2 × 2 grid network in vissim. The model is given in the folder '*Grid network simulation/Net_2cross2'* (files: Net_2cross2.inpx, Net_2cross2.layx, and 3Ring_3cross3_Net1.sig). The simulation is run with a script file named '*Net_2cross2_MFD_DiffControlPolicies.py*' for each demand pattern, policy, and adaptivity value. After each simulation, the link segment file is manually saved and named 'LS.att'.
6. Data for Fig. 16 is obtained by microsimulating a 4 × 4 grid network in vissim. The model is given in the folder '*Grid network simulation/Net_4cross4'* (files:

Net_4cross4.inpx, Net_4cross4.layx, and 3Ring_3cross3_Net1.sig). The simulation is run with a script file named '*Net_4cross4_MFD_Hysteresis.py*' for each demand pattern, policy, and adaptivity value. After each simulation, the link segment file is manually saved and named '*LS.att'*.

**Code description**:

Versions of software and Python libraries are as follows

1. Python 3.8
2. pandas 1.3.3
3. numpy 1.21.5
4. matplotlib 3.4.3
5. PySimpleGUI 4.45.0
6. Pywin32 302
7. Os
8. Re
9. Warnings
10. Datetime
11. PTV Vissim 6.0

**Simulation software description:**

1. Clone the git repository: git clone https://github.com/NamrataGupta5/COMMTR_Replication_package.git
2. In each script, please adapt the directory location
3. Scripts
    a. Inside folder '*Flow within bin and outflow from a bin/MFD_for fixed time_diff signal plan"*':
        i. Script named '*MFD_Variation_with_signal_plan.py'*
            - reads the result file discussed in the first point of data description to plot the data (Figure 4.)
    b. Inside folder '*Flow within bin and outflow from a bin/Proportionality constant for bin outflow*':
        i. Script named '*Analyses_proportionality_constant.py'*
            - reads the result file discussed in the second point of data description to plot the data (Figure 5.)
    c. Inside folder '*Stability analyses using three-bin*':
        i. Script named '*3bin_stability_Analyses_Phase_Diagram_MFD.py'*
            - Change input parameters (bin characteristics and TSC parameters) in lines 195-200.
            - The code saves the phase digrams and MFDs for each policy
    d. Inside folder '*Three-ring network simulation*':
        i. Script named *'3Ring_MFD_DiffControlPolicies.py'*
            - The script asked for vissim files and takes network, simulation and TSC parameters from lines 61-82
            - Outputs aggregate network parameters for each repetition in a file
        ii. Script named '*Plot_3Ring_MFDShape_LoadingRate.py*' plots link segment result data
    e. Inside folder '*Grid network simulation/Net_2cross2*':
        i. Script named *'Net_2cross2_MFD_DiffControlPolicies.py'*

        - The script asked for vissim files and takes network, simulation and TSC parameters from lines 369-411. Please change the TSC parameters in lines 410-411 to simulate other TSCs
        - Outputs aggregate network parameters for each repetition in a file
    ii. Script named *'Plot_Net_MFDShape_LoadingRate'* for plotting

f. Inside folder '*Grid network simulation/Net_4cross4*':
    i. Script named *'Net_4cross4_MFD_Hysteresis.py'*
        - Same as 2×2 grid network script except network parameters are changed
    ii. Script named *'Plot_Net_MFDShape_Hysteresis'* for plotting

**Experiment design description:**

This work deals with only the simulation experiments discussed in the manuscript.