main

May 31, 2024

[ ]:
```
# We study the data within one day, so we load all the data of the day at the␣
 ↪beginning of the program.
Date = '2023-01-04'
```

[ ]:
```
# In this study, we focus on the Contious Count Station:
# [601, 626, 632, 635, 643, 645, 647, 648, 656, 658, 662, 671, 672, 675, 676]

Utah_Vehicles_Person = 0.850 # vehicles per person in Utah
Region_1_population = 19902# Highland population
Region_2_population = 0.5 * 84373 # half of population of Lehi 2022
Region_3_population = 0.5 * 84373 # the other half of population of Lehi 2022
Region_4_population = 37268 + 37630 # the population of American Fork and␣
 ↪Pleasant Grove 2022
Region_5_population = 17865 # Heber City which is far from Highland
Region_6_population = 19080+59179+50731 # Bluffdale + Herriman + Draper␣
 ↪population
Region_7_population = 54149  # Eagle Mountain population
region_8_population = 11704 + 95910 + 14535 # Lindon + Orem + Vineyard␣
 ↪population

## suppose the total number of cars in the city
region_1_initial_traffic_amount = int(Region_1_population *␣
 ↪Utah_Vehicles_Person)
region_2_initial_traffic_amount = int(Region_2_population *␣
 ↪Utah_Vehicles_Person)
region_3_initial_traffic_amount = int(Region_3_population *␣
 ↪Utah_Vehicles_Person)
region_4_initial_traffic_amount = int(Region_4_population *␣
 ↪Utah_Vehicles_Person)
region_5_initial_traffic_amount = int(Region_5_population *␣
 ↪Utah_Vehicles_Person)
region_6_initial_traffic_amount = int(Region_6_population *␣
 ↪Utah_Vehicles_Person)
region_7_initial_traffic_amount = int(Region_7_population *␣
 ↪Utah_Vehicles_Person)
region_8_initial_traffic_amount = int(region_8_population *␣
 ↪Utah_Vehicles_Person)
```

[ ]:
```
import model
import numpy as np

_initial_traffic_amouts_list = [region_1_initial_traffic_amount,␣
 ↪region_2_initial_traffic_amount,
                                region_3_initial_traffic_amount,␣
 ↪region_4_initial_traffic_amount,
                                region_5_initial_traffic_amount,␣
 ↪region_6_initial_traffic_amount,
                                region_7_initial_traffic_amount,␣
 ↪region_8_initial_traffic_amount]

# initialization of CCSs
CCS_601, CCS_626, CCS_632, CCS_635, CCS_643,\
CCS_645, CCS_647, CCS_648, CCS_656, CCS_658,\
CCS_662, CCS_671, CCS_672, CCS_675, CCS_676,\
Region_1, Region_2, Region_3, Region_4,\
Region_5, Region_6, Region_7, Region_8 = \
model.initialize_CSSs_and_Regions(date=Date,
                                  ␣
 ↪initials_traffic_amount_list=_initial_traffic_amouts_list)

model.initialize_region_transition_matrix_v2()
```

```
CCS idx:  645
Warning: the data for this date is empty
CCS idx:  656
Warning: the data for this date is empty
The initial traffic amount of Region  1  is  16916
The initial traffic amount of Region  2  is  35858
The initial traffic amount of Region  3  is  35858
The initial traffic amount of Region  4  is  63663
The initial traffic amount of Region  5  is  15185
The initial traffic amount of Region  6  is  109641
The initial traffic amount of Region  7  is  46026
The initial traffic amount of Region  8  is  103826
```

[ ]:
```
print("The traffic amount of each region during the day: ")
print(model.current_time_traffic_amount)
```

```
The traffic amount of each region during the day:
[[ 16916.  35858.  35858.  63663.  15185. 109641.  46026. 103826.]
 [ 16943.  35946.  35707.  63602.  15179. 109742.  46116. 103738.]
 [ 16967.  35905.  35620.  63649.  15183. 109826.  46160. 103663.]
 [ 16969.  35914.  35586.  63688.  15181. 109895.  46145. 103595.]
 [ 16968.  35946.  35642.  63821.  15178. 109753.  46082. 103583.]
 [ 16930.  35971.  35902.  64095.  15178. 109400.  45834. 103663.]
```

```
 [ 16932.  35833.  36694.  65058.  15179. 108329.  45168. 103780.]
 [ 16559.  36233.  37943.  66497.  15183. 106229.  44296. 104033.]
 [ 16150.  36407.  39751.  68752.  15191. 104467.  42768. 103487.]
 [ 15477.  35416.  40055.  71770.  15211. 105269.  41848. 101927.]
 [ 15015.  35400.  40130.  73566.  15220. 105785.  41306. 100551.]
 [ 14797.  36092.  40489.  74286.  15229. 105576.  40967.  99537.]
 [ 14551.  36569.  40764.  74423.  15236. 105759.  40824.  98847.]
 [ 14319.  37533.  40645.  74251.  15243. 105610.  40808.  98564.]
 [ 14232.  38345.  40489.  74243.  15251. 105610.  40897.  97906.]
 [ 14317.  38893.  40266.  73557.  15267. 105994.  40991.  97688.]
 [ 14470.  39389.  39771.  72068.  15271. 106720.  41637.  97647.]
 [ 14572.  41414.  38993.  69768.  15250. 106958.  42518.  97500.]
 [ 14901.  42893.  37977.  67421.  15210. 106314.  43538.  98719.]
 [ 15237.  43090.  36368.  65161.  15191. 108393.  44984.  98549.]
 [ 15506.  43821.  35333.  64029.  15205. 109208.  45811.  98060.]
 [ 15732.  44064.  34637.  62871.  15195. 109772.  46380.  98322.]
 [ 15934.  44384.  34205.  62110.  15197. 110170.  46813.  98160.]
 [ 16035.  44588.  33862.  61584.  15192. 110606.  47168.  97938.]
 [ 16108.  44623.  33656.  61377.  15184. 110833.  47354.  97838.]]
```

[ ]:
```
np.set_printoptions(precision=5, suppress=True)
np.set_printoptions(threshold=np.inf)
print("The transition amount among different regions: ")
print()
for _ in range(model.hourly_traffic_among_regions.shape[0]):
    print(f'The transition amount during {_}:00 to {_+1}:00')
    print(model.hourly_traffic_among_regions[_])
    print()
```

```
The transition amount among different regions:

The transition amount during 0:00 to 1:00
[[ 16837.     76.      0.      0.      3.      0.      0.      0.]
 [    97.  34210.    111.    644.      0.    796.      0.      0.]
 [     0.    260.  35351.     98.      0.      0.    149.      0.]
 [     0.    705.    186.  61967.      0.      0.      0.    805.]
 [     9.      0.      0.      0.  15176.      0.      0.      0.]
 [     0.    695.      0.      0.      0. 108946.      0.      0.]
 [     0.      0.     59.      0.      0.      0.  45967.      0.]
 [     0.      0.      0.    893.      0.      0.      0. 102933.]]

The transition amount during 1:00 to 2:00
[[ 16910.     28.      0.      0.      5.      0.      0.      0.]
 [    56.  34918.     63.    411.      0.    498.      0.      0.]
 [     0.    125.  35459.     54.      0.      0.     69.      0.]
 [     0.    420.     73.  62624.      0.      0.      0.    485.]
 [     1.      0.      0.      0.  15178.      0.      0.      0.]
```

```
 [     0.    414.      0.      0.      0. 109328.      0.      0.]
 [     0.      0.     25.      0.      0.      0.  46091.      0.]
 [     0.      0.      0.    560.      0.      0.      0. 103178.]]

The transition amount during 2:00 to 3:00
[[ 16943.     21.      0.      0.      3.      0.      0.      0.]
 [    21.  35102.     60.    314.      0.    408.      0.      0.]
 [     0.    108.  35394.     66.      0.      0.     52.      0.]
 [     0.    344.     65.  62854.      0.      0.      0.    386.]
 [     5.      0.      0.      0.  15178.      0.      0.      0.]
 [     0.    339.      0.      0.      0. 109487.      0.      0.]
 [     0.      0.     67.      0.      0.      0.  46093.      0.]
 [     0.      0.      0.    454.      0.      0.      0. 103209.]]

The transition amount during 3:00 to 4:00
[[ 16935.     34.      0.      0.      0.      0.      0.      0.]
 [    30.  35069.     86.    397.      0.    332.      0.      0.]
 [     0.     59.  35397.     88.      0.      0.     42.      0.]
 [     0.    310.     54.  62884.      0.      0.      0.    440.]
 [     3.      0.      0.      0.  15178.      0.      0.      0.]
 [     0.    474.      0.      0.      0. 109421.      0.      0.]
 [     0.      0.    105.      0.      0.      0.  46040.      0.]
 [     0.      0.      0.    452.      0.      0.      0. 103143.]]

The transition amount during 4:00 to 5:00
[[ 16872.     96.      0.      0.      0.      0.      0.      0.]
 [    58.  34240.    263.    773.      0.    612.      0.      0.]
 [     0.    113.  35239.    210.      0.      0.     80.      0.]
 [     0.    557.     72.  62303.      0.      0.      0.    889.]
 [     0.      0.      0.      0.  15178.      0.      0.      0.]
 [     0.    965.      0.      0.      0. 108788.      0.      0.]
 [     0.      0.    328.      0.      0.      0.  45754.      0.]
 [     0.      0.      0.    809.      0.      0.      0. 102774.]]

The transition amount during 5:00 to 6:00
[[ 16625.    304.      0.      0.      1.      0.      0.      0.]
 [   307.  31070.    903.   2086.      0.   1605.      0.      0.]
 [     0.    298.  34642.    714.      0.      0.    248.      0.]
 [     0.   1485.    235.  60037.      0.      0.      0.   2338.]
 [     0.      0.      0.      0.  15178.      0.      0.      0.]
 [     0.   2676.      0.      0.      0. 106724.      0.      0.]
 [     0.      0.    914.      0.      0.      0.  44920.      0.]
 [     0.      0.      0.   2221.      0.      0.      0. 101442.]]

The transition amount during 6:00 to 7:00
[[ 16205.    722.      0.      0.      5.      0.      0.      0.]
 [   353.  25997.   1834.   4182.      0.   3467.      0.      0.]
 [     0.    767.  33880.   1367.      0.      0.    680.      0.]
```

```
 [     0.   3180.    677.  56415.      0.      0.      0.   4786.]
 [     1.      0.      0.      0.  15178.      0.      0.      0.]
 [     0.   5567.      0.      0.      0. 102762.      0.      0.]
 [     0.      0.   1552.      0.      0.      0.  43616.      0.]
 [     0.      0.      0.   4533.      0.      0.      0.  99247.]]

The transition amount during 7:00 to 8:00
[[15051.  1496.     0.     0.    12.     0.     0.     0.]
 [ 1095. 20553.  2872.  5852.     0.  5861.     0.     0.]
 [    0.  1341. 33521.  2356.     0.     0.   725.     0.]
 [    0.  5394.  1105. 52782.     0.     0.     0.  7216.]
 [    4.     0.     0.     0. 15179.     0.     0.     0.]
 [    0.  7623.     0.     0.     0. 98606.     0.     0.]
 [    0.     0.  2253.     0.     0.     0. 42043.     0.]
 [    0.     0.     0.  7762.     0.     0.     0. 96271.]]

The transition amount during 8:00 to 9:00
[[14336.  1783.     0.     0.    31.     0.     0.     0.]
 [ 1130. 19894.  2297.  6217.     0.  6869.     0.     0.]
 [    0.  1351. 34906.  2646.     0.     0.   848.     0.]
 [    0.  6321.  1084. 53677.     0.     0.     0.  7670.]
 [   11.     0.     0.     0. 15180.     0.     0.     0.]
 [    0.  6067.     0.     0.     0. 98400.     0.     0.]
 [    0.     0.  1768.     0.     0.     0. 41000.     0.]
 [    0.     0.     0.  9230.     0.     0.     0. 94257.]]

The transition amount during 9:00 to 10:00
[[14037.  1412.     0.     0.    28.     0.     0.     0.]
 [  959. 21692.  1850.  5081.     0.  5834.     0.     0.]
 [    0.  1346. 36021.  1961.     0.     0.   727.     0.]
 [    0.  5632.   990. 58682.     0.     0.     0.  6466.]
 [   19.     0.     0.     0. 15192.     0.     0.     0.]
 [    0.  5318.     0.     0.     0. 99951.     0.     0.]
 [    0.     0.  1269.     0.     0.     0. 40579.     0.]
 [    0.     0.     0.  7842.     0.     0.     0. 94085.]]

The transition amount during 10:00 to 11:00
[[ 13611.   1375.      0.      0.     29.      0.      0.      0.]
 [  1166.  23197.   1508.   4404.      0.   5125.      0.      0.]
 [     0.   1068.  36894.   1413.      0.      0.    755.      0.]
 [     0.   5118.    993.  61746.      0.      0.      0.   5709.]
 [    20.      0.      0.      0.  15200.      0.      0.      0.]
 [     0.   5334.      0.      0.      0. 100451.      0.      0.]
 [     0.      0.   1094.      0.      0.      0.  40212.      0.]
 [     0.      0.      0.   6723.      0.      0.      0.  93828.]]

The transition amount during 11:00 to 12:00
[[ 13032.   1728.      0.      0.     37.      0.      0.      0.]
```

```
 [  1489.  22340.   1618.   5372.      0.   5273.      0.      0.]
 [     0.   1215.  36929.   1430.      0.      0.    915.      0.]
 [     0.   6196.   1159.  60738.      0.      0.      0.   6193.]
 [    30.      0.      0.      0.  15199.      0.      0.      0.]
 [     0.   5090.      0.      0.      0. 100486.      0.      0.]
 [     0.      0.   1058.      0.      0.      0.  39909.      0.]
 [     0.      0.      0.   6883.      0.      0.      0.  92654.]]

The transition amount during 12:00 to 13:00
[[ 12556.   1959.      0.      0.     36.      0.      0.      0.]
 [  1734.  22141.   1629.   5814.      0.   5251.      0.      0.]
 [     0.   1601.  36547.   1511.      0.      0.   1105.      0.]
 [     0.   6432.   1348.  59772.      0.      0.      0.   6871.]
 [    29.      0.      0.      0.  15207.      0.      0.      0.]
 [     0.   5400.      0.      0.      0. 100359.      0.      0.]
 [     0.      0.   1121.      0.      0.      0.  39703.      0.]
 [     0.      0.      0.   7154.      0.      0.      0.  91693.]]

The transition amount during 13:00 to 14:00
[[ 12553.   1705.      0.      0.     61.      0.      0.      0.]
 [  1626.  22861.   1470.   6031.      0.   5545.      0.      0.]
 [     0.   1549.  36465.   1479.      0.      0.   1152.      0.]
 [     0.   6685.   1491.  59280.      0.      0.      0.   6795.]
 [    53.      0.      0.      0.  15190.      0.      0.      0.]
 [     0.   5545.      0.      0.      0. 100065.      0.      0.]
 [     0.      0.   1063.      0.      0.      0.  39745.      0.]
 [     0.      0.      0.   7453.      0.      0.      0.  91111.]]

The transition amount during 14:00 to 15:00
[[12420.  1749.     0.     0.    63.     0.     0.     0.]
 [ 1850. 22527.  1534.  6204.     0.  6230.     0.     0.]
 [    0.  1867. 35864.  1453.     0.     0.  1305.     0.]
 [    0.  6904.  1657. 58089.     0.     0.     0.  7593.]
 [   47.     0.     0.     0. 15204.     0.     0.     0.]
 [    0.  5846.     0.     0.     0. 99764.     0.     0.]
 [    0.     0.  1211.     0.     0.     0. 39686.     0.]
 [    0.     0.     0.  7811.     0.     0.     0. 90095.]]

The transition amount during 15:00 to 16:00
[[12348.  1905.     0.     0.    64.     0.     0.     0.]
 [ 2062. 19889.  1992.  7167.     0.  7783.     0.     0.]
 [    0.  2238. 34404.  1738.     0.     0.  1886.     0.]
 [    0.  8300.  2135. 54082.     0.     0.     0.  9040.]
 [   60.     0.     0.     0. 15207.     0.     0.     0.]
 [    0.  7057.     0.     0.     0. 98937.     0.     0.]
 [    0.     0.  1240.     0.     0.     0. 39751.     0.]
 [    0.     0.     0.  9081.     0.     0.     0. 88607.]]
```

```
The transition amount during 16:00 to 17:00
[[12295.  2128.     0.     0.    47.     0.     0.     0.]
 [ 2209. 19313.  1945.  7709.     0.  8213.     0.     0.]
 [    0.  2589. 33120.  1805.     0.     0.  2257.     0.]
 [    0.  9409.  2552. 50119.     0.     0.     0.  9988.]
 [   68.     0.     0.     0. 15203.     0.     0.     0.]
 [    0.  7975.     0.     0.     0. 98745.     0.     0.]
 [    0.     0.  1376.     0.     0.     0. 40261.     0.]
 [    0.     0.     0. 10135.     0.     0.     0. 87512.]]

The transition amount during 17:00 to 18:00
[[12230.  2317.     0.     0.    25.     0.     0.     0.]
 [ 2606. 20836.  1734.  8520.     0.  7718.     0.     0.]
 [    0.  2605. 32260.  1759.     0.     0.  2369.     0.]
 [    0.  8773.  2634. 47374.     0.     0.     0. 10987.]
 [   65.     0.     0.     0. 15185.     0.     0.     0.]
 [    0.  8362.     0.     0.     0. 98596.     0.     0.]
 [    0.     0.  1349.     0.     0.     0. 41169.     0.]
 [    0.     0.     0.  9768.     0.     0.     0. 87732.]]

The transition amount during 18:00 to 19:00
[[ 13186.   1709.      0.      0.      6.      0.      0.      0.]
 [  2026.  25112.   1425.   6254.      0.   8076.      0.      0.]
 [     0.   2472.  31918.   1285.      0.      0.   2302.      0.]
 [     0.   7800.   2169.  49480.      0.      0.      0.   7972.]
 [    25.      0.      0.      0.  15185.      0.      0.      0.]
 [     0.   5997.      0.      0.      0. 100317.      0.      0.]
 [     0.      0.    856.      0.      0.      0.  42682.      0.]
 [     0.      0.      0.   8142.      0.      0.      0.  90577.]]

The transition amount during 19:00 to 20:00
[[ 14051.   1151.      0.      0.     35.      0.      0.      0.]
 [  1434.  32064.   1031.   4014.      0.   4547.      0.      0.]
 [     0.   1784.  32210.    935.      0.      0.   1439.      0.]
 [     0.   5090.   1480.  53531.      0.      0.      0.   5060.]
 [    21.      0.      0.      0.  15170.      0.      0.      0.]
 [     0.   3732.      0.      0.      0. 104661.      0.      0.]
 [     0.      0.    612.      0.      0.      0.  44372.      0.]
 [     0.      0.      0.   5549.      0.      0.      0.  93000.]]

The transition amount during 20:00 to 21:00
[[ 14671.    814.      0.      0.     21.      0.      0.      0.]
 [  1030.  35097.    767.   3325.      0.   3602.      0.      0.]
 [     0.   1421.  32246.    645.      0.      0.   1021.      0.]
 [     0.   3694.   1172.  54879.      0.      0.      0.   4284.]
 [    31.      0.      0.      0.  15174.      0.      0.      0.]
 [     0.   3038.      0.      0.      0. 106170.      0.      0.]
 [     0.      0.    452.      0.      0.      0.  45359.      0.]
```

```
 [     0.      0.      0.   4022.      0.      0.      0.  94038.]]

The transition amount during 21:00 to 22:00
[[ 15171.    547.      0.      0.     14.      0.      0.      0.]
 [   751.  37205.    546.   2644.      0.   2918.      0.      0.]
 [     0.   1049.  32445.    424.      0.      0.    719.      0.]
 [     0.   3063.    928.  55547.      0.      0.      0.   3333.]
 [    12.      0.      0.      0.  15183.      0.      0.      0.]
 [     0.   2520.      0.      0.      0. 107252.      0.      0.]
 [     0.      0.    286.      0.      0.      0.  46094.      0.]
 [     0.      0.      0.   3495.      0.      0.      0.  94827.]]

The transition amount during 22:00 to 23:00
[[ 15605.    313.      0.      0.     16.      0.      0.      0.]
 [   409.  39668.    438.   1688.      0.   2181.      0.      0.]
 [     0.    744.  32673.    279.      0.      0.    509.      0.]
 [     0.   2118.    597.  57198.      0.      0.      0.   2197.]
 [    21.      0.      0.      0.  15176.      0.      0.      0.]
 [     0.   1745.      0.      0.      0. 108425.      0.      0.]
 [     0.      0.    154.      0.      0.      0.  46659.      0.]
 [     0.      0.      0.   2419.      0.      0.      0.  95741.]]

The transition amount during 23:00 to 24:00
[[ 15874.    158.      0.      0.      3.      0.      0.      0.]
 [   223.  41769.    221.   1075.      0.   1300.      0.      0.]
 [     0.    385.  33039.    177.      0.      0.    261.      0.]
 [     0.   1238.    321.  58681.      0.      0.      0.   1344.]
 [    11.      0.      0.      0.  15181.      0.      0.      0.]
 [     0.   1073.      0.      0.      0. 109533.      0.      0.]
 [     0.      0.     75.      0.      0.      0.  47093.      0.]
 [     0.      0.      0.   1444.      0.      0.      0.  96494.]]
```

[ ]:
```python
np.set_printoptions(precision=10, suppress=True)

print()
# print(get_hour_k_step_transition_matrix(1, 2))
print()
print(model.get_hour_k_step_transition_matrix(0, 9))
print()
print(np.dot(model.get_hour_k_step_transition_matrix(0, 7), model.
  ↪get_hour_k_step_transition_matrix(7, 2)))
print()
print( np.allclose(model.get_hour_k_step_transition_matrix(0, 9),
                   np.dot(model.get_hour_k_step_transition_matrix(0, 7),
                          model.get_hour_k_step_transition_matrix(7, 2))) )
```

```
[[0.7522263469 0.1610128263 0.0137692748 0.0318555146 0.0032778707
  0.0355869684 0.0001892964 0.0020819018]
 [0.0417305737 0.2293585186 0.1214470627 0.2533109777 0.0000883888
  0.2956559087 0.0043790406 0.0540295293]
 [0.0031252635 0.0681929011 0.6760196654 0.1488091176 0.0000032762
  0.0196125887 0.0625333115 0.0217038761]
 [0.0061093668 0.1185596261 0.0498622289 0.5259369699 0.0000067511
  0.0384005551 0.0014574945 0.2596670077]
 [0.0019069032 0.0002292292 0.0000158809 0.0000371373 0.9977671048
  0.0000414445 0.0000001936 0.0000021065]
 [0.00623569   0.1133986646 0.015548362  0.034700998  0.0000065376
  0.8268734472 0.000269031  0.0029672695]
 [0.0001080792 0.0054990464 0.1274007442 0.0113767674 0.000000049
  0.0006535632 0.8542217858 0.0007399648]
 [0.0003596385 0.0171934179 0.0046559837 0.1830017652 0.0000002147
  0.0021909413 0.0000650568 0.7925329819]]

[[0.7522263469 0.1610128263 0.0137692748 0.0318555146 0.0032778707
  0.0355869684 0.0001892964 0.0020819018]
 [0.0417305737 0.2293585186 0.1214470627 0.2533109777 0.0000883888
  0.2956559087 0.0043790406 0.0540295293]
 [0.0031252635 0.0681929011 0.6760196654 0.1488091176 0.0000032762
  0.0196125887 0.0625333115 0.0217038761]
 [0.0061093668 0.1185596261 0.0498622289 0.5259369699 0.0000067511
  0.0384005551 0.0014574945 0.2596670077]
 [0.0019069032 0.0002292292 0.0000158809 0.0000371373 0.9977671048
  0.0000414445 0.0000001936 0.0000021065]
 [0.00623569   0.1133986646 0.015548362  0.034700998  0.0000065376
  0.8268734472 0.000269031  0.0029672695]
 [0.0001080792 0.0054990464 0.1274007442 0.0113767674 0.000000049
  0.0006535632 0.8542217858 0.0007399648]
 [0.0003596385 0.0171934179 0.0046559837 0.1830017652 0.0000002147
  0.0021909413 0.0000650568 0.7925329819]]

True
```

[ ]:
```python
test_atol = 2e-2
false_case_count = 0
count = 0
for begin_hour in range(0, 24):
    for k in range(1, 24):
        for l in range(1, 24):
            if begin_hour + k + l >= 24:
                continue
            count += 1
```

```
                        # print(get_hour_k_step_transition_matrix(begin_hour,␣
 ↪k))
            tmp_bool = np.allclose(
                model.get_hour_k_step_transition_matrix(begin_hour, k+l),
                np.dot(model.get_hour_k_step_transition_matrix(begin_hour, k),
                       model.get_hour_k_step_transition_matrix(begin_hour + k,␣
 ↪l)),
                        )
            if not tmp_bool:
                print((begin_hour, k, l), ' this is the pair of (begin_hour, k,␣
 ↪l)')
                false_case_count += 1

print(count, " this is the count")
print(false_case_count, " this is the false case count")
```

```
2024  this is the count
0  this is the false case count
```

[ ]:
```
# experiment 0: the calculation of k-step transition probability matrix of␣
 ↪discrete time Markov Chain
exp0_begin_hour = 0
exp0_k = 5

print("Experiment 0")
print('The construction of Markov Chain and the calculation of k-step␣
 ↪transition probability matrix')
print('-----------------------------------------------------')
# # print(Region_1.getTransitionMatrix2Regions(begin_hour=exp0_hour1,␣
 ↪end_hour=exp0_hour3))

print(f"The {exp0_k}-step transition matrix from hour {exp0_begin_hour}")
print(model.get_hour_k_step_transition_matrix(begin_hour=exp0_begin_hour,␣
 ↪k=exp0_k))
```

```
Experiment 0
The construction of Markov Chain and the calculation of k-step transition
probability matrix
-----------------------------------------------------
The 5-step transition matrix from hour 0
[[0.9844283927 0.0141073345 0.0001000432 0.0003684498 0.0006464278
  0.0003457101 0.0000001108 0.0000035311]
 [0.0068681506 0.8444942206 0.014918352  0.0639403918 0.0000015379
  0.0685430255 0.0000392506 0.0011950709]
 [0.000051876  0.0170542394 0.9569735787 0.0144051392 0.000000002
  0.0005869097 0.010720576  0.000207679 ]
 [0.000087474  0.0330524964 0.0070290145 0.9138798042 0.000000003
```

```
 0.0009898659 0.0000304349 0.044930907 ]
[0.0011743182 0.0000102107 0.0000000329 0.0000001071 0.9988152403
 0.0000000903 0.           0.0000000004]
[0.000053063  0.0245519889 0.000180023  0.0006495359 0.0000000017
 0.9745595506 0.000000179  0.0000056578]
[0.0000000352 0.0000319309 0.012518772  0.0000458253 0.
 0.0000003978 0.9874028041 0.0000002347]
[0.0000004876 0.0003968684 0.00006541   0.0291429578 0.
 0.0000054856 0.0000001309 0.9703886598]]
```

[ ]:
```
# experiment 1: the verification of the Chapman-Kolmogorov equation
exp1_m = 5
exp1_k = 10
exp1_l = 4

exp1_P_begin_m_step_k_plus_l = model.get_hour_k_step_transition_matrix(
    ␣
 ↪                                                                  begin_hour=exp1_m,
                                    k=exp1_k+exp1_l)

exp1_P_begin_m_step_k = model.get_hour_k_step_transition_matrix(
                                                                        begin_hour=exp1_m,
                                                                        k=exp1_k)

exp1_P_begin_m_plus_k_step_l = model.get_hour_k_step_transition_matrix(
                                                                        begin_hour=exp1_m+exp1
                                                                        k=exp1_l)


print("Experiment 1")
print('The Chapman-Kolmogorov equation verification')
print('-------------------------------------------------')
print("The left side of the Chapman-Kolmogorov equation")
print(exp1_P_begin_m_step_k_plus_l)
print()
print("The right side of the Chapman-Kolmogorov equation")
print(np.dot(exp1_P_begin_m_step_k, exp1_P_begin_m_plus_k_step_l))
print()
print("The equality of the left side and the right side of the␣
 ↪Chapman-Kolmogorov equation")
print(np.allclose(exp1_P_begin_m_step_k_plus_l,
                                  np.dot(exp1_P_begin_m_step_k,␣
 ↪exp1_P_begin_m_plus_k_step_l)))

print("exp1_P_begin_m_step_k_plus_l")
print(exp1_P_begin_m_step_k_plus_l)
print()
print("exp1_P_begin_m_step_k")
```

```
print(exp1_P_begin_m_step_k)
print()
print("exp1_P_begin_m_plus_k_step_l")
print(exp1_P_begin_m_plus_k_step_l)
print()
```

```
Experiment 1
The Chapman-Kolmogorov equation verification
------------------------------------------------
The left side of the Chapman-Kolmogorov equation
[[0.2506058269 0.1693671176 0.0646380422 0.1358869916 0.0158096418
  0.2600794715 0.016994452  0.0866184563]
 [0.0495052355 0.1262049758 0.0994541392 0.1593780286 0.0013739797
  0.3448476217 0.0438547831 0.1753812364]
 [0.0276242593 0.1102931417 0.2518682041 0.1556875836 0.0003426976
  0.1305146811 0.1850447622 0.1386246704]
 [0.0322355785 0.1092912909 0.0846435845 0.2240014307 0.0005065758
  0.163675691  0.0274415028 0.3582043458]
 [0.0174510871 0.0049318644 0.0005777729 0.0017172704 0.972232541
  0.0025630259 0.0000694184 0.0004570199]
 [0.0360223586 0.1325303742 0.0443310845 0.0942868838 0.0005163503
  0.6213757889 0.0119326711 0.0590044885]
 [0.0063437213 0.0370743964 0.1562929921 0.0430774328 0.0000470468
  0.0269624525 0.7060095619 0.0241923962]
 [0.0160048253 0.0814676324 0.0502231965 0.2398649274 0.0001346311
  0.069027515  0.0101358561 0.5331414163]]

The right side of the Chapman-Kolmogorov equation
[[0.2506058269 0.1693671176 0.0646380422 0.1358869916 0.0158096418
  0.2600794715 0.016994452  0.0866184563]
 [0.0495052355 0.1262049758 0.0994541392 0.1593780286 0.0013739797
  0.3448476217 0.0438547831 0.1753812364]
 [0.0276242593 0.1102931417 0.2518682041 0.1556875836 0.0003426976
  0.1305146811 0.1850447622 0.1386246704]
 [0.0322355785 0.1092912909 0.0846435845 0.2240014307 0.0005065758
  0.163675691  0.0274415028 0.3582043458]
 [0.0174510871 0.0049318644 0.0005777729 0.0017172704 0.972232541
  0.0025630259 0.0000694184 0.0004570199]
 [0.0360223586 0.1325303742 0.0443310845 0.0942868838 0.0005163503
  0.6213757889 0.0119326711 0.0590044885]
 [0.0063437213 0.0370743964 0.1562929921 0.0430774328 0.0000470468
  0.0269624525 0.7060095619 0.0241923962]
 [0.0160048253 0.0814676324 0.0502231965 0.2398649274 0.0001346311
  0.069027515  0.0101358561 0.5331414163]]

The equality of the left side and the right side of the Chapman-Kolmogorov
equation
```

```
True
exp1_P_begin_m_step_k_plus_l
[[0.2506058269 0.1693671176 0.0646380422 0.1358869916 0.0158096418
  0.2600794715 0.016994452  0.0866184563]
 [0.0495052355 0.1262049758 0.0994541392 0.1593780286 0.0013739797
  0.3448476217 0.0438547831 0.1753812364]
 [0.0276242593 0.1102931417 0.2518682041 0.1556875836 0.0003426976
  0.1305146811 0.1850447622 0.1386246704]
 [0.0322355785 0.1092912909 0.0846435845 0.2240014307 0.0005065758
  0.163675691  0.0274415028 0.3582043458]
 [0.0174510871 0.0049318644 0.0005777729 0.0017172704 0.972232541
  0.0025630259 0.0000694184 0.0004570199]
 [0.0360223586 0.1325303742 0.0443310845 0.0942868838 0.0005163503
  0.6213757889 0.0119326711 0.0590044885]
 [0.0063437213 0.0370743964 0.1562929921 0.0430774328 0.0000470468
  0.0269624525 0.7060095619 0.0241923962]
 [0.0160048253 0.0814676324 0.0502231965 0.2398649274 0.0001346311
  0.069027515  0.0101358561 0.5331414163]]

exp1_P_begin_m_step_k
[[0.4005255823 0.1957855316 0.0514273197 0.1258562575 0.0124194497
  0.173988575  0.0045836281 0.0354136561]
 [0.0493455937 0.128926448  0.1229724874 0.2037515056 0.0009053714
  0.3394136555 0.0214460581 0.1332388804]
 [0.0166751698 0.0986345964 0.4190169135 0.1779382236 0.0001593687
  0.0761537386 0.1239353765 0.087486613 ]
 [0.0236217974 0.1099074607 0.0818064328 0.3151652489 0.0002604237
  0.1155309517 0.0100301026 0.3436775822]
 [0.0108409078 0.0019306334 0.0001318286 0.0004363272 0.9861250936
  0.0004853273 0.0000044721 0.00004541  ]
 [0.0241182857 0.1224395523 0.0353019444 0.0813054306 0.0002601054
  0.7065911312 0.0035854793 0.0263980711]
 [0.002430745  0.0218099086 0.1723466473 0.0322113687 0.000016689
  0.0097605954 0.7517241107 0.0096999354]
 [0.0070795362 0.0572163085 0.0281835732 0.2617176897 0.0000494966
  0.0284044816 0.001955505  0.6153934092]]

exp1_P_begin_m_plus_k_step_l
[[0.5709600048 0.212641907  0.0213281927 0.0751337537 0.0087319926
  0.099375168  0.0012679087 0.0105610723]
 [0.0818966876 0.1876900158 0.0844013349 0.2044213077 0.0003300085
  0.3481103087 0.0122214089 0.0809289279]
 [0.0121497469 0.1147258433 0.5164217268 0.114288705  0.0000086348
  0.0452020938 0.1683054484 0.028897801 ]
 [0.0204337429 0.1480006181 0.0804933266 0.3609520312 0.0000168592
  0.0761182666 0.0093278371 0.3046573183]
 [0.0112416982 0.0021526962 0.0000748936 0.0003107232 0.985815323
  0.0003891397 0.0000014669 0.0000140593]
```

```
[0.0142647108 0.1235355002 0.0113461493 0.040357885  0.0000102517
 0.8043976932 0.0006503853 0.0054374245]
[0.0003452269 0.0084395956 0.083207715  0.0062369044 0.0000000499
 0.0012853271 0.8998233601 0.0006618209]
[0.0019177303 0.0385606346 0.0148473065 0.2092801588 0.0000003075
 0.0070847566 0.0007553283 0.7275537773]]
```

[ ]:
```
# experiment 2: the construction of time-homogeneous Markov Chain and the␣
 ↪calculation of limit distribution and stationary distribution

# # Suppose the traffic flow is a time-homogeneous Markov chain
# # The transition matrix is the same for all hours
exp2_begin_hour = 0
exp2_k = 1
exp2_initial_distribution_time = 10

exp2_initial_distribution = model.
 ↪current_time_traffic_amount[exp2_initial_distribution_time]
exp2_initial_distribution = exp2_initial_distribution / np.
 ↪sum(exp2_initial_distribution)
exp2_transition_matrix_P = model.get_hour_k_step_transition_matrix(
                                                                begin_hour=exp2_begin_hour,
                                                                k=exp2_k)

# compute the limit distribution of the traffic flow according to the␣
 ↪transition matrix and the initial distribution
def compute_limit_distribution(transition_matrix, initial_distribution, alltol␣
 ↪= 1e-6):
       next_distribution = np.dot(initial_distribution, transition_matrix)
       while not np.allclose(initial_distribution, next_distribution,␣
 ↪atol=alltol):
              initial_distribution = next_distribution
              next_distribution = np.dot(initial_distribution,␣
 ↪transition_matrix)
       return next_distribution

print("Experiment 2")
print('The construction of time-homogeneous Markov Chain and the calculation of␣
 ↪limit distribution')
print('--------------------------------------------------')
print('The transition matrix of the traffic flow:')
print(exp2_transition_matrix_P)
print()
print('The initial distribution of the traffic flow:')
print(exp2_initial_distribution)
print(exp2_initial_distribution * np.sum(_initial_traffic_amouts_list))
```

```
print()
exp2_limit_distribution = compute_limit_distribution(exp2_transition_matrix_P,␣
 ↪exp2_initial_distribution)
print('The limit distribution of the traffic flow:')
print(exp2_limit_distribution)
print(exp2_limit_distribution * np.sum(_initial_traffic_amouts_list))



# compute the stationary distribution of the traffic flow according to the␣
 ↪transition matrix
def compute_stationary_distribution(transition_matrix):
        eigenvalues, eigenvectors = np.linalg.eig(transition_matrix.T)
        stationary_index = np.where(np.isclose(eigenvalues, 1))[0][0]
        stationary_distribution = np.real(eigenvectors[:, stationary_index])
        stationary_distribution /= np.sum(stationary_distribution)
        return stationary_distribution


print("Experiment 2")
print('The construction of time-homogeneous Markov Chain and the calculation of␣
 ↪stationary distribution')
print('-----------------------------------------------------')
exp2_stationary_distribution =␣
 ↪compute_stationary_distribution(exp2_transition_matrix_P)
print('The stationary distribution of the traffic flow:')
print(exp2_stationary_distribution)
print(exp2_stationary_distribution * np.sum(_initial_traffic_amouts_list))
```

```
Experiment 2
The construction of time-homogeneous Markov Chain and the calculation of limit
distribution
-----------------------------------------------------
The transition matrix of the traffic flow:
[[0.9953298652 0.0044927879 0.           0.           0.0001773469
  0.           0.           0.          ]
 [0.0027051146 0.9540409393 0.0030955435 0.01795973   0.
  0.0221986725 0.           0.          ]
 [0.           0.0072508227 0.9858608958 0.0027330024 0.
  0.           0.0041552792 0.          ]
 [0.           0.0110739362 0.0029216342 0.9733597223 0.
  0.           0.           0.0126447073]
 [0.0005926902 0.           0.           0.           0.9994073098
  0.           0.           0.          ]
 [0.           0.0063388696 0.           0.           0.
  0.9936611304 0.           0.          ]
 [0.           0.           0.0012818842 0.           0.
```

```
  0.           0.9987181158 0.          ]
 [0.           0.           0.           0.0086009285 0.
  0.           0.           0.9913990715]]

The initial distribution of the traffic flow:
[0.035166158  0.0829092238 0.0939872076 0.1722966089 0.0356462821
 0.2477557129 0.0967414801 0.2354973265]
[ 15015.  35400.  40130.  73566.  15220. 105785.  41306. 100551.]

The limit distribution of the traffic flow:
[0.0509213378 0.0840687562 0.060964514  0.11979023   0.0172132835
 0.2944262904 0.1964999378 0.1761156503]
[ 21742.0363622036  35895.089035811   26030.2014381436  51147.1938933037
   7349.6073096246 125712.0764899111  83900.1679231133  75196.6275478868]
Experiment 2
The construction of time-homogeneous Markov Chain and the calculation of
stationary distribution
--------------------------------------------------
The stationary distribution of the traffic flow:
[0.0506437051 0.0841115653 0.061160225  0.1198787901 0.0151537925
 0.2945580551 0.1982533349 0.176240532 ]
[ 21623.4946878873  35913.3673916747  26113.764748294   51185.0066352778
   6470.2602434536 125768.3364677012  84648.8211549889  75249.9486707226]
```

[ ]:
```
# experiment 3: the construction of non-time-homogeneous Markov Chain

# exp3_begin_hour - exp3_begin_hour + 3.
# We Use the linear model to construct the transition matrix of␣
 ↪non-time-homogeneous Markov Chain
exp3_begin_hour = 18
exp3_end_hour = exp3_begin_hour + 3


# We combine the transition matrix of exp3_begin_hour - exp3_begin_hour + 1 and␣
 ↪exp3_begin_hour + 1 - exp3_begin_hour + 2
# to construct the transition matrix of exp3_begin_hour + 2 - exp3_begin_hour +␣
 ↪3
exp3_transition_matrix_6_7 = model.
 ↪get_hour_k_step_transition_matrix(begin_hour=exp3_begin_hour, k=1)
exp3_transition_matrix_7_8 = model.
 ↪get_hour_k_step_transition_matrix(begin_hour=exp3_begin_hour+1, k=1)

# Use the linear interpolation to construct the transition matrix of␣
 ↪exp3_begin_hour + 2 - exp3_begin_hour + 3
exp3_model_transition_matrix_8_9 = 2 * exp3_transition_matrix_7_8 -␣
 ↪exp3_transition_matrix_6_7
```

```
exp3_real_transition_matrix_8_9 = model.
  ↪get_hour_k_step_transition_matrix(begin_hour=exp3_begin_hour+2, k=1)



print("Experiment 3")
print('The construction of non-time-homogeneous Markov Chain')
print('-------------------------------------------------')
print(f'The transition matrix of the traffic flow from {exp3_begin_hour}:00 to␣
  ↪{exp3_begin_hour+1}:00:')
print(exp3_transition_matrix_6_7)
print()
print(f"The transition matrix of the traffic flow from {exp3_begin_hour+1}:00␣
  ↪to {exp3_begin_hour+2}:00:")
print(exp3_transition_matrix_7_8)
print()
print(f'The model transition matrix of the traffic flow from␣
  ↪{exp3_begin_hour+2}:00 to {exp3_begin_hour+3}:00:')
print(exp3_model_transition_matrix_8_9)
print()
print(f'The real transition matrix of the traffic flow from {exp3_begin_hour+2}:
  ↪00 to {exp3_begin_hour+3}:00:')
print(exp3_real_transition_matrix_8_9)
print()


print('The real transition number is ')
print(model.hourly_traffic_among_regions[exp3_begin_hour+2][0])

print('The Non Time Alignment model transition number is ')
print(model.current_time_traffic_amount[exp3_begin_hour+2] *␣
  ↪exp3_model_transition_matrix_8_9[0])

print('The Time Alignment model transition number is ')
print(model.current_time_traffic_amount[exp3_begin_hour+2] *␣
  ↪(exp3_transition_matrix_6_7+exp3_transition_matrix_7_8)[0]/2)
```

```
Experiment 3
The construction of non-time-homogeneous Markov Chain
-------------------------------------------------
The transition matrix of the traffic flow from 18:00 to 19:00:
[[0.8849070532 0.1146902892 0.           0.           0.0004026575
  0.           0.           0.          ]
 [0.0472338144 0.5854568344 0.0332222041 0.1458046768 0.
  0.1882824703 0.           0.          ]
 [0.           0.0650920294 0.8404560655 0.0338362693 0.
  0.           0.0606156358 0.          ]
```

```
 [0.           0.1156909568 0.0321709853 0.7338959671 0.
  0.           0.           0.1182420907]
 [0.0016436555 0.           0.           0.           0.9983563445
  0.           0.           0.          ]
 [0.           0.0564083752 0.           0.           0.
  0.9435916248 0.           0.          ]
 [0.           0.           0.0196609858 0.           0.
  0.           0.9803390142 0.          ]
 [0.           0.           0.           0.0824765243 0.
  0.           0.           0.9175234757]]

The transition matrix of the traffic flow from 19:00 to 20:00:
[[0.9221631555 0.0755398044 0.           0.           0.0022970401
  0.           0.           0.          ]
 [0.0332791831 0.7441169645 0.0239266651 0.093153864  0.
  0.1055233233 0.           0.          ]
 [0.           0.0490541135 0.8856687198 0.0257094149 0.
  0.           0.0395677519 0.          ]
 [0.           0.0781142094 0.0227129725 0.8215190068 0.
  0.           0.           0.0776538113]
 [0.0013823975 0.           0.           0.           0.9986176025
  0.           0.           0.          ]
 [0.           0.0344302676 0.           0.           0.
  0.9655697324 0.           0.          ]
 [0.           0.           0.0136048373 0.           0.
  0.           0.9863951627 0.          ]
 [0.           0.           0.           0.0563070148 0.
  0.           0.           0.9436929852]]

The model transition matrix of the traffic flow from 20:00 to 21:00:
[[0.9594192577 0.0363893196 0.           0.           0.0041914227
  0.           0.           0.          ]
 [0.0193245518 0.9027770945 0.0146311261 0.0405030513 0.
  0.0227641762 0.           0.          ]
 [0.           0.0330161976 0.930881374  0.0175825604 0.
  0.           0.018519868  0.          ]
 [0.           0.040537462  0.0132549597 0.9091420464 0.
  0.           0.           0.0370655319]
 [0.0011211395 0.           0.           0.           0.9988788605
  0.           0.           0.          ]
 [0.           0.0124521601 0.           0.           0.
  0.9875478399 0.           0.          ]
 [0.           0.           0.0075486887 0.           0.
  0.           0.9924513113 0.          ]
 [0.           0.           0.           0.0301375053 0.
  0.           0.           0.9698624947]]

The real transition matrix of the traffic flow from 20:00 to 21:00:
```

```
[[0.9461498775 0.0524958081 0.           0.           0.0013543145
  0.           0.           0.          ]
 [0.0235047124 0.8009173684 0.0175030237 0.0758768627 0.
  0.0821980329 0.           0.          ]
 [0.           0.0402173605 0.9126312512 0.0182548892 0.
  0.           0.028896499  0.          ]
 [0.           0.057692608  0.0183042059 0.8570960034 0.
  0.           0.           0.0669071827]
 [0.002038803  0.           0.           0.           0.997961197
  0.           0.           0.          ]
 [0.           0.0278184748 0.           0.           0.
  0.9721815252 0.           0.          ]
 [0.           0.           0.0098666259 0.           0.
  0.           0.9901333741 0.          ]
 [0.           0.           0.           0.0410157047 0.
  0.           0.           0.9589842953]]

The real transition number is
[14671.   814.     0.     0.    21.     0.     0.     0.]
The Non Time Alignment model transition number is
[14876.7550104497  1594.6163743911     0.               0.
    63.7305815415     0.               0.               0.          ]
The Time Alignment model transition number is
[14010.21532801    4168.036467264      0.               0.
    20.5244513044     0.               0.               0.          ]
```

[ ]:
```
# experiment 4: the construction of time-homogeneous continuous Markov Chain␣
 ↪and the calculation of Q matrix and the limit distribution and stationary␣
 ↪distribution

# The calculation of Q matrix
# input a list of x and a list of matrix, for each element of matrix, build the␣
 ↪linear model of x to the element
# return a function, the input of this function is an element of x, the output␣
 ↪is a matrix
def linearModelMatrix(X, matrix):
    def linearModel(x, y):
        p = np.polyfit(x, y, 1)
        f = np.poly1d(p)
        return f
    _matrix_shape = matrix[0].shape
    _matrix_len = len(matrix)
    _f_list = []
    for i in range(_matrix_shape[0]):
        _f_list.append([])
        for j in range(_matrix_shape[1]):
```

```
            _f_list[i].append(linearModel(X, [matrix[k][i][j] for k in␣
 ↪range(_matrix_len)]))

    def _f(x):
        _result = np.zeros(_matrix_shape)
        for i in range(_matrix_shape[0]):
            for j in range(_matrix_shape[1]):
                _result[i][j] = _f_list[i][j](x)
        return _result
    return _f


# return a Q-matrix
# begin_hour:
# n_hour_steps, the number of hours used to fit the Q matrix
# epsilon: use the epsilon that tends to 0 to calculate the Q matrix
def get_hour_Q_matrix(begin_hour, n_hour_steps, epsilon):
    trans_matrixs_list = [model.get_hour_k_step_transition_matrix(begin_hour,␣
 ↪k) for k in range(1, n_hour_steps+1)]
    phi_t_list = [i - np.eye(8) for i in trans_matrixs_list]
    phi_t_list = [phi_t_list[i] / (i+1) for i in range(n_hour_steps)]
    Phi_t = linearModelMatrix(list(range(1, 1+n_hour_steps)),
                              phi_t_list)
    return Phi_t(epsilon)




exp4_begin_hour1 = 12
exp4_fitting_hour_steps = 4
exp4_epsilon = 0.1
exp4_Q = get_hour_Q_matrix(begin_hour=exp4_begin_hour1,␣
 ↪n_hour_steps=exp4_fitting_hour_steps, epsilon=exp4_epsilon)

exp4_transition_matrix_P1 = model.
 ↪get_hour_k_step_transition_matrix(begin_hour=exp4_begin_hour1, k=1)
exp4_transition_matrix_P2 = model.
 ↪get_hour_k_step_transition_matrix(begin_hour=exp4_begin_hour1, k=2)
exp4_transition_matrix_P3 = model.
 ↪get_hour_k_step_transition_matrix(begin_hour=exp4_begin_hour1, k=3)
exp4_transition_matrix_P4 = model.
 ↪get_hour_k_step_transition_matrix(begin_hour=exp4_begin_hour1, k=4)


print("Experiment 4")
print('The construction of time-homogeneous continuous Markov Chain')
```

```
print('-------------------------------------------------')
print('The Q matrix of the traffic flow:')
print(exp4_Q)
print()
print('The transition matrix of the traffic flow from 12:00 to 13:00:')
print(exp4_transition_matrix_P1)
print()
print('The transition matrix of the traffic flow from 12:00 to 14:00:')
print(exp4_transition_matrix_P2)
print()
print('The transition matrix of the traffic flow from 12:00 to 15:00:')
print(exp4_transition_matrix_P3)
print()
print('The transition matrix of the traffic flow from 12:00 to 16:00:')
print(exp4_transition_matrix_P4)
print()
```

```
Experiment 4
The construction of time-homogeneous continuous Markov Chain
-------------------------------------------------
The Q matrix of the traffic flow:
[[-0.144417628   0.1525599482 -0.0013237875 -0.0031431007  0.0024705162
  -0.0051508894 -0.0000878363 -0.0009072224]
 [ 0.0531883339 -0.4390173745  0.0492818905  0.1823882414 -0.0000486955
   0.1584489658 -0.0005061463 -0.0037352153]
 [-0.0005188432  0.0424664971 -0.1043975108  0.0395343983 -0.0000031929
  -0.002108823   0.0265823295 -0.001554855 ]
 [-0.0008582823  0.098701552   0.0186082181 -0.210926484  -0.0000067846
  -0.0035760849 -0.0003403118  0.0983981775]
 [ 0.0019016641 -0.0001069153 -0.0000056071 -0.000019649  -0.0017470169
  -0.0000218963 -0.0000000499 -0.0000005295]
 [-0.0006457253  0.0565556642 -0.0006759171 -0.0017924488 -0.000004178
  -0.0530322571 -0.0000357028 -0.0003694351]
 [-0.0000237106 -0.0004258436  0.0285481394 -0.0004166275 -0.0000000265
  -0.0000888994 -0.0275428522 -0.0000501796]
 [-0.0001291818 -0.0015974542 -0.000746743   0.0782213154 -0.0000001651
  -0.0004838838 -0.0000294138 -0.0752344737]]

The transition matrix of the traffic flow from 12:00 to 13:00:
[[0.8628960209 0.1346299223 0.           0.           0.0024740568
  0.           0.           0.          ]
 [0.0474172113 0.605458175  0.0445459269 0.1589871202 0.
  0.1435915666 0.           0.          ]
 [0.           0.0392748504 0.8965508782 0.0370670199 0.
  0.           0.0271072515 0.          ]
 [0.           0.0864248955 0.0181126802 0.8031388146 0.
  0.           0.           0.0923236096]
```

```
 [0.0019033867 0.           0.           0.           0.9980966133
  0.           0.           0.          ]
 [0.           0.0510594843 0.           0.           0.
  0.9489405157 0.           0.          ]
 [0.           0.           0.0274593376 0.           0.
  0.           0.9725406624 0.          ]
 [0.           0.           0.           0.0723744777 0.
  0.           0.           0.9276255223]]

The transition matrix of the traffic flow from 12:00 to 14:00:
[[0.7623138042 0.1847490682 0.0052728528 0.0216330446 0.0061414554
  0.0198897748 0.           0.          ]
 [0.0677987061 0.3979758827 0.0668703888 0.2258401921 0.0002020008
  0.2255007311 0.0012625639 0.0145495344]
 [0.0017014602 0.0614271539 0.8073367395 0.068528109  0.
  0.0058023352 0.051812054  0.0033921483]
 [0.0037440887 0.1256394412 0.0357622782 0.6627318371 0.
  0.0127681253 0.0005133671 0.1588408624]
 [0.0051390248 0.0002266411 0.           0.           0.9946343341
  0.           0.           0.          ]
 [0.0022119927 0.0809235004 0.001999772  0.0082045067 0.
  0.9066602282 0.           0.          ]
 [0.           0.0010464882 0.0499689041 0.000999197  0.
  0.           0.9479854107 0.          ]
 [0.           0.0065160521 0.0014533184 0.1279250215 0.
  0.           0.           0.864105608 ]]

The transition matrix of the traffic flow from 12:00 to 15:00:
[[0.674189352  0.205574917  0.0125442881 0.0470066237 0.0094970208
  0.0488053886 0.0001699492 0.0022124605]
 [0.0783681027 0.2787029191 0.0802308151 0.2446519552 0.0005014991
  0.2776782803 0.003380476  0.0364859524]
 [0.0044484584 0.0802175329 0.7206368932 0.0927990785 0.0000075318
  0.0154613596 0.0762991    0.0101300457]
 [0.0093290199 0.1382555711 0.0515098875 0.5528163199 0.0000165737
  0.0324742786 0.001650819  0.2139475303]
 [0.0075608947 0.000764693  0.0000090668 0.0000366692 0.9915918533
  0.0000368229 0.           0.          ]
 [0.0058346149 0.0988559272 0.0051918161 0.0195840675 0.0000097917
  0.8696202345 0.0000644546 0.0008390935]
 [0.0000504891 0.0030118409 0.0723959652 0.0027443032 0.
  0.0001700253 0.921525186  0.0001021901]
 [0.0003143747 0.015791073  0.0044030917 0.1701360196 0.
  0.0010586779 0.0000468419 0.8082499212]]

The transition matrix of the traffic flow from 12:00 to 16:00:
[[0.5924052324 0.2040838123 0.0226166021 0.073190514  0.0124734652
  0.0866942032 0.0007523641 0.0077838067]
```

```
 [0.0823682646 0.2035027395 0.0900284241 0.2380903733 0.00084985
  0.3149627851 0.0070361077 0.0631614556]
 [0.00808961   0.0931672531 0.6248353486 0.1150581104 0.0000273877
  0.0304845357 0.1077445791 0.0205931755]
 [0.0153760066 0.1393456307 0.0671875705 0.4540410381 0.0000582113
  0.057978921  0.0040135279 0.261999094 ]
 [0.0104585966 0.001404182  0.0000479768 0.0001682657 0.9877286513
  0.0001873964 0.0000004247 0.0000045066]
 [0.0102732953 0.1117260678 0.0100695091 0.0329177292 0.0000358352
  0.831503949  0.0003056818 0.0031679326]
 [0.0002032249 0.0058916822 0.0899669741 0.0057070502 0.0000002257
  0.0007614141 0.8970394691 0.0004299597]
 [0.0011083384 0.0276299901 0.0095104934 0.2033249218 0.0000014053
  0.004148193  0.0002516592 0.7540249989]]


The construction of time-homogeneous continuous Markov Chain
------------------------------------------------
The Q matrix of the traffic flow:
[[-0.144417628   0.1525599482 -0.0013237875 -0.0031431007  0.0024705162
  -0.0051508894 -0.0000878363 -0.0009072224]
 [ 0.0531883339 -0.4390173745  0.0492818905  0.1823882414 -0.0000486955
   0.1584489658 -0.0005061463 -0.0037352153]
 [-0.0005188432  0.0424664971 -0.1043975108  0.0395343983 -0.0000031929
  -0.002108823   0.0265823295 -0.001554855 ]
 [-0.0008582823  0.098701552   0.0186082181 -0.210926484  -0.0000067846
  -0.0035760849 -0.0003403118  0.0983981775]
 [ 0.0019016641 -0.0001069153 -0.0000056071 -0.000019649  -0.0017470169
  -0.0000218963 -0.0000000499 -0.0000005295]
 [-0.0006457253  0.0565556642 -0.0006759171 -0.0017924488 -0.000004178
  -0.0530322571 -0.0000357028 -0.0003694351]
 [-0.0000237106 -0.0004258436  0.0285481394 -0.0004166275 -0.0000000265
  -0.0000888994 -0.0275428522 -0.0000501796]
 [-0.0001291818 -0.0015974542 -0.000746743   0.0782213154 -0.0000001651
  -0.0004838838 -0.0000294138 -0.0752344737]]

The transition matrix of the traffic flow from 12:00 to 13:00:
[[0.8628960209 0.1346299223 0.           0.           0.0024740568
  0.           0.           0.           0.          ]
 [0.0474172113 0.605458175  0.0445459269 0.1589871202 0.
  0.1435915666 0.           0.           0.          ]
 [0.           0.0392748504 0.8965508782 0.0370670199 0.
  0.           0.0271072515 0.           0.          ]
 [0.           0.0864248955 0.0181126802 0.8031388146 0.
  0.           0.           0.0923236096]
 [0.0019033867 0.           0.           0.           0.9980966133
  0.           0.           0.           0.          ]
 [0.           0.0510594843 0.           0.           0.
```

```
  0.9489405157 0.           0.          ]
 [0.           0.           0.0274593376 0.           0.
  0.           0.9725406624 0.          ]
 [0.           0.           0.           0.0723744777 0.
  0.           0.           0.9276255223]]

The transition matrix of the traffic flow from 12:00 to 14:00:
[[0.7623138042 0.1847490682 0.0052728528 0.0216330446 0.0061414554
  0.0198897748 0.           0.          ]
 [0.0677987061 0.3979758827 0.0668703888 0.2258401921 0.0002020008
  0.2255007311 0.0012625639 0.0145495344]
 [0.0017014602 0.0614271539 0.8073367395 0.068528109  0.
  0.0058023352 0.051812054  0.0033921483]
 [0.0037440887 0.1256394412 0.0357622782 0.6627318371 0.
  0.0127681253 0.0005133671 0.1588408624]
 [0.0051390248 0.0002266411 0.           0.           0.9946343341
  0.           0.           0.          ]
 [0.0022119927 0.0809235004 0.001999772  0.0082045067 0.
  0.9066602282 0.           0.          ]
 [0.           0.0010464882 0.0499689041 0.000999197  0.
  0.           0.9479854107 0.          ]
 [0.           0.0065160521 0.0014533184 0.1279250215 0.
  0.           0.           0.864105608 ]]

The transition matrix of the traffic flow from 12:00 to 15:00:
[[0.674189352  0.205574917  0.0125442881 0.0470066237 0.0094970208
  0.0488053886 0.0001699492 0.0022124605]
 [0.0783681027 0.2787029191 0.0802308151 0.2446519552 0.0005014991
  0.2776782803 0.003380476  0.0364859524]
 [0.0044484584 0.0802175329 0.7206368932 0.0927990785 0.0000075318
  0.0154613596 0.0762991    0.0101300457]
 [0.0093290199 0.1382555711 0.0515098875 0.5528163199 0.0000165737
  0.0324742786 0.001650819  0.2139475303]
 [0.0075608947 0.000764693  0.0000090668 0.0000366692 0.9915918533
  0.0000368229 0.           0.          ]
 [0.0058346149 0.0988559272 0.0051918161 0.0195840675 0.0000097917
  0.8696202345 0.0000644546 0.0008390935]
 [0.0000504891 0.0030118409 0.0723959652 0.0027443032 0.
  0.0001700253 0.921525186  0.0001021901]
 [0.0003143747 0.015791073  0.0044030917 0.1701360196 0.
  0.0010586779 0.0000468419 0.8082499212]]

The transition matrix of the traffic flow from 12:00 to 16:00:
[[0.5924052324 0.2040838123 0.0226166021 0.073190514  0.0124734652
  0.0866942032 0.0007523641 0.0077838067]
 [0.0823682646 0.2035027395 0.0900284241 0.2380903733 0.00084985
  0.3149627851 0.0070361077 0.0631614556]
 [0.00808961   0.0931672531 0.6248353486 0.1150581104 0.0000273877
```

```
 0.0304845357 0.1077445791 0.0205931755]
[0.0153760066 0.1393456307 0.0671875705 0.4540410381 0.0000582113
 0.057978921  0.0040135279 0.261999094 ]
[0.0104585966 0.001404182  0.0000479768 0.0001682657 0.9877286513
 0.0001873964 0.0000004247 0.0000045066]
[0.0102732953 0.1117260678 0.0100695091 0.0329177292 0.0000358352
 0.831503949  0.0003056818 0.0031679326]
[0.0002032249 0.0058916822 0.0899669741 0.0057070502 0.0000002257
 0.0007614141 0.8970394691 0.0004299597]
[0.0011083384 0.0276299901 0.0095104934 0.2033249218 0.0000014053
 0.004148193  0.0002516592 0.7540249989]]
```

[ ]:
```
# experiment 5: the calculation of stationary distribution by Q matrix

# compute the stationary distribution of the traffic flow according to the Q␣
 ↪matrix
from copy import deepcopy
def compute_stationary_distribution_by_Q(Q):
        _Q = deepcopy(Q)
        _Q = _Q.T
        _Q[-1] = np.ones(8)
        b = np.zeros(8)
        b[-1] = 1
        return np.linalg.solve(_Q, b)


exp5_begin_hour = 8
exp5_fitting_hour_steps = 4
exp5_epsilon = 0.5
exp5_Q = get_hour_Q_matrix(begin_hour=exp5_begin_hour,␣
 ↪n_hour_steps=exp5_fitting_hour_steps, epsilon=exp5_epsilon)

exp5_stationary_distribution = compute_stationary_distribution_by_Q(exp5_Q)

print("Experiment 5")
print('The calculation of stationary distribution by Q matrix')
print('-------------------------------------------------')
print('The Q matrix of the traffic flow:')
print(exp5_Q)
print()
print('The stationary distribution of the traffic flow:')
print(exp5_stationary_distribution)
print(exp5_stationary_distribution * np.sum(_initial_traffic_amouts_list))
print("The begin traiffic amount of the traffic flow:")
print(model.current_time_traffic_amount[exp5_begin_hour])
```

```
Experiment 5
```

```
The calculation of stationary distribution by Q matrix
--------------------------------------------------
The Q matrix of the traffic flow:
[[-0.1129871796  0.1128079849 -0.0001750714 -0.0004957066  0.001904023
  -0.0007102407 -0.0000308647 -0.0003129449]
 [ 0.0319879105 -0.4721416894  0.0656971621  0.1779638607 -0.0000083202
   0.196921598  -0.0001276774 -0.0002928441]
 [-0.0001884196  0.0360562339 -0.1262041902  0.0697545931 -0.0000007084
  -0.0004609613  0.0214542247 -0.0004107722]
 [-0.0002629089  0.0954781154  0.0161999205 -0.2268261359 -0.0000016847
  -0.0002894043 -0.000074788   0.1157768858]
 [ 0.0006878843 -0.0000244755 -0.000001043  -0.0000032954 -0.0006555843
  -0.0000034205 -0.0000000056 -0.00000006  ]
 [-0.0002618247  0.0604338016 -0.0001439154 -0.0004201793 -0.0000011258
  -0.0594209578 -0.0000166558 -0.0001691429]
 [-0.0000129222 -0.0000497927  0.0425708733 -0.0000775598 -0.00000001
  -0.0000493722 -0.0423403743 -0.000040842 ]
 [-0.0000665106 -0.0002836668 -0.0002350701  0.0931763786 -0.0000000504
  -0.0002537265 -0.0000082302 -0.0923291239]]

The stationary distribution of the traffic flow:
[0.0232720826 0.085623622  0.0816109942 0.1880372292 0.06543035
 0.2808913891 0.0405895801 0.2345447529]
[  9936.5509338028  36558.9747455205  34845.6910067174  80286.8198650094
  27936.9928173364 119933.0390845458  17330.6547701383 100144.2767769294]
The begin traiffic amount of the traffic flow:
[ 16150.  36407.  39751.  68752.  15191. 104467.  42768. 103487.]
```

[ ]:
```python
# expereiment 6: the construction of non-time-homogeneous continuous Markov␣
 ↪Chain
# use linear interpolation to construct the non-time-homogeneous Q matrix
exp6_begin_hour1 = 18
exp6_begin_hour2 = exp6_begin_hour1 + 1

exp6_fitting_hour_steps = 3
exp6_epsilon = 0.1

exp6_Q1 = get_hour_Q_matrix(begin_hour=exp6_begin_hour1,␣
 ↪n_hour_steps=exp6_fitting_hour_steps, epsilon=exp6_epsilon)
exp6_Q2 = get_hour_Q_matrix(begin_hour=exp6_begin_hour2,␣
 ↪n_hour_steps=exp6_fitting_hour_steps, epsilon=exp6_epsilon)
exp6_model_Q3 = 2 * exp6_Q2 - exp6_Q1
exp6_real_Q3 =␣
 ↪get_hour_Q_matrix(begin_hour=exp6_begin_hour2+exp6_fitting_hour_steps,␣
 ↪n_hour_steps=exp6_fitting_hour_steps, epsilon=exp6_epsilon)

print("Experiment 6")
```

```
print('The construction of non-time-homogeneous continuous Markov Chain')
print('-------------------------------------------------')
print('The Q matrix of the traffic flow from 6:00 to 7:00:')
print(exp6_Q1)
print()
print('The Q matrix of the traffic flow from 7:00 to 8:00:')
print(exp6_Q2)
print()
print('The model Q matrix of the traffic flow from 8:00 to 9:00:')
print(exp6_model_Q3)
print()
print('The real Q matrix of the traffic flow from 8:00 to 9:00:')
print(exp6_real_Q3)
print()
```

```
Experiment 6
The construction of non-time-homogeneous continuous Markov Chain
-------------------------------------------------
The Q matrix of the traffic flow from 6:00 to 7:00:
[[-0.1323579182  0.1384299806 -0.000645184  -0.0024856186  0.0001948444
  -0.0029728679 -0.0000162998 -0.0001469364]
 [ 0.0566465991 -0.4967122138  0.0393446746  0.1780239307 -0.0000217236
   0.2257988689 -0.0003261035 -0.0027540323]
 [-0.0005062468  0.0768223944 -0.1822340121  0.0386286444 -0.000000603
  -0.0017874835  0.0699201994 -0.0008428927]
 [-0.0008076479  0.1408908459  0.0374215142 -0.3123837733 -0.0000010718
  -0.0028547973 -0.000331617   0.1380665472]
 [ 0.0016103748 -0.000030996  -0.0000004467 -0.0000019365 -0.0015748978
  -0.0000020979  0.             0.          ]
 [-0.0004119914  0.0672045019 -0.0003159235 -0.0012164438 -0.0000005226
  -0.0651793357 -0.0000080168 -0.0000722681]
 [-0.0000046598 -0.0002582442  0.0226198344 -0.0001352395  0.
  -0.0000162956 -0.0221984435 -0.0000069518]
 [-0.0000311275 -0.001418579  -0.0005117047  0.0972471128  0.
  -0.0001088557 -0.000011127  -0.0951657189]]

The Q matrix of the traffic flow from 7:00 to 8:00:
[[-0.0889752522  0.0898485505 -0.0003290105 -0.0014322923  0.0026507989
  -0.0016946932 -0.0000056417 -0.0000624596]
 [ 0.0390360851 -0.2963976064  0.027797752   0.1098798496 -0.0000107046
   0.1215635789 -0.0001820351 -0.0016869195]
 [-0.0003040121  0.0569365646 -0.12920341    0.0293742831 -0.0000002109
  -0.0011776309  0.0448856096 -0.0005111933]
 [-0.0004463688  0.0920610716  0.0257388082 -0.2032325886 -0.0000003359
  -0.0017286205 -0.0001831368  0.0877911709]
 [ 0.0015065122 -0.0000243998 -0.0000001848 -0.0000008951 -0.0014800447
  -0.0000009878  0.             0.          ]
```

```
 [-0.0002162505  0.0393316268 -0.0001621753 -0.0007119788 -0.000000148
  -0.0382100343 -0.0000025714 -0.0000284685]
 [-0.0000019169 -0.0001441411  0.0156457158 -0.0000661913  0.
  -0.000007448  -0.0154233122 -0.0000027064]
 [-0.0000113807 -0.0008997269 -0.0002988117  0.064278385   0.
  -0.0000442194 -0.0000043978 -0.0630198485]]

The model Q matrix of the traffic flow from 8:00 to 9:00:
[[-0.0455925861  0.0412671205 -0.000012837  -0.0003789659  0.0051067533
  -0.0004165184  0.0000050165  0.0000220172]
 [ 0.0214255711 -0.0960829989  0.0162508293  0.0417357686  0.0000003144
   0.0173282889 -0.0000379666 -0.0006198067]
 [-0.0001017775  0.0370507348 -0.0761728079  0.0201199218  0.0000001812
  -0.0005677783  0.0198510198 -0.000179494 ]
 [-0.0000850897  0.0432312972  0.0140561021 -0.0940814039  0.0000004001
  -0.0006024437 -0.0000346567  0.0375157946]
 [ 0.0014026496 -0.0000178036  0.000000077   0.0000001464 -0.0013851916
   0.0000001222  0.            0.          ]
 [-0.0000205096  0.0114587517 -0.0000084271 -0.0002075137  0.0000002265
  -0.0112407329  0.000002874   0.0000153312]
 [ 0.000000826  -0.000030038   0.0086715973  0.0000028568  0.
   0.0000013997 -0.0086481809  0.0000015391]
 [ 0.0000083662 -0.0003808748 -0.0000859187  0.0313096571  0.
   0.0000204168  0.0000023315 -0.030873978 ]]

The real Q matrix of the traffic flow from 8:00 to 9:00:
[[0.1739940564 0.0298844623 0.0000162272 0.000078933  0.0014864228
  0.0000954539 0.           0.          ]
 [0.0140913648 0.0423096407 0.0150405822 0.058447112  0.0000002873
  0.0755155583 0.0000126772 0.0001383331]
 [0.0000181309 0.0331474539 0.136771505  0.0126827013 0.
  0.0001056957 0.0228004002 0.0000296684]
 [0.000028425  0.0521904971 0.0147266315 0.0838787683 0.
  0.0001657061 0.0000123478 0.0545531798]
 [0.0021218833 0.0000022693 0.           0.           0.2034314029
  0.           0.           0.          ]
 [0.0000132029 0.0243911187 0.0000130844 0.000063646  0.
  0.1810745036 0.           0.          ]
 [0.           0.0000062338 0.0050208552 0.0000028659 0.
  0.           0.2005256007 0.          ]
 [0.           0.0000825663 0.0000214085 0.037936156  0.
  0.           0.           0.1675154247]]
```

[ ]:
```
# experiment 6 extension: the prediction of traffic flow by Q matrix

from scipy.linalg import expm
```

```
# the number of hours to predict the traffic flow
exp6_predict_hour = 1
exp6_model_hour_P = np.clip(expm(exp6_model_Q3 * exp6_predict_hour), a_min=0,␣
 ↪a_max=None)
exp6_real_hour_P = np.clip(expm(exp6_real_Q3 * exp6_predict_hour), a_min=0,␣
 ↪a_max=None)

print("Experiment 6 extension")
print('The prediction of traffic flow by Q matrix')
print('-------------------------------------------------')
print(f'The real transition matrix of the traffic flow from␣
 ↪{exp6_begin_hour1+2}:00 to {exp6_begin_hour1+3}:00:')
print(exp6_real_hour_P)
print()
print("The real transition number is")
print(model.hourly_traffic_among_regions[exp6_begin_hour1+2][0])
print(f'The model transition matrix of the traffic flow from␣
 ↪{exp6_begin_hour1+2}:00 to {exp6_begin_hour1+3}:00:')
print(exp6_model_hour_P)
print()
print('The model transition number is')
print(model.current_time_traffic_amount[exp6_begin_hour1+2] *␣
 ↪exp6_model_hour_P[0])
```

```
Experiment 6 extension
The prediction of traffic flow by Q matrix
-------------------------------------------------
The real transition matrix of the traffic flow from 20:00 to 21:00:
[[1.1902904662 0.033357726  0.0002774673 0.0010574323 0.0017953357
  0.0014036288 0.0000024379 0.0000227186]
 [0.015728903  1.0463446861 0.0169433158 0.0624537109 0.0000123851
  0.0845948133 0.000213222  0.0019161767]
 [0.0002860987 0.0366663442 1.1470227279 0.0152366812 0.0000001553
  0.0015470815 0.0269973619 0.0004507621]
 [0.0004406893 0.0559415085 0.0168919393 1.090402555  0.0000002404
  0.0023828594 0.0002113229 0.0619360981]
 [0.0025628708 0.0000390806 0.0000002261 0.0000008738 1.2256029961
  0.0000011491 0.0000000016 0.0000000147]
 [0.0002121783 0.0273237087 0.000226823  0.0008643115 0.0000001149
  1.1995595319 0.0000019913 0.0000185532]
 [0.0000005567 0.0001022982 0.0059450695 0.0000420014 0.0000000002
  0.0000030148 1.222113388  0.000000884 ]
 [0.0000065306 0.0011886343 0.0003493048 0.0430700935 0.0000000028
  0.0000353583 0.0000030651 1.1835542235]]

The real transition number is
```

```
[14671.   814.     0.     0.    21.     0.     0.     0.]
The model transition matrix of the traffic flow from 20:00 to 21:00:
[[0.9558500061 0.0384612367 0.0003010828 0.00044492   0.0049893355
  0.           0.0000061118 0.0000124792]
 [0.0199714481 0.9099909127 0.0151879491 0.0381170506 0.0000524911
  0.0164168569 0.0001170793 0.0001462122]
 [0.0002755004 0.034408943  0.9271506113 0.0191957949 0.0000005764
  0.           0.0190326616 0.0001810417]
 [0.0003500441 0.0395697152 0.0132378133 0.9117104288 0.0000009265
  0.           0.0000999842 0.0352548047]
 [0.0013702154 0.0000106344 0.0000000769 0.           0.9986192917
  0.           0.0000000039 0.0000000132]
 [0.000096735  0.0108647902 0.0000792148 0.0000273711 0.0000003765
  0.9889176477 0.0000031397 0.0000107249]
 [0.000001164  0.0001240291 0.0083143363 0.0000864572 0.000000003
  0.           0.9914725507 0.0000018366]
 [0.0000074661 0.0002724971 0.0001248719 0.0294258967 0.0000000239
  0.000011504  0.0000023251 0.9701554153]]

The model transition number is
[14821.4101940969  1685.4098535761    10.6381583465    28.4877823067
    75.8628455908     0.               0.2799892895     1.2237080344]
```