High Performance Dynamic Traffic Assignment Based on Variational Inequality

GU Yangsong, LIANG Geyu

Mentor: Dr. Cheng Liu, Dr. Kwai Wong

08/02/2018

Outline

1. Introduction

- Dynamic Traffic Assignment
- Dynamic User Equilibrium

2. Algorithm

- Differential Variational Inequality
- Dynamic Network Loading Based on ODE
- Dynamic Network Loading Based on LWR

3. Implementation

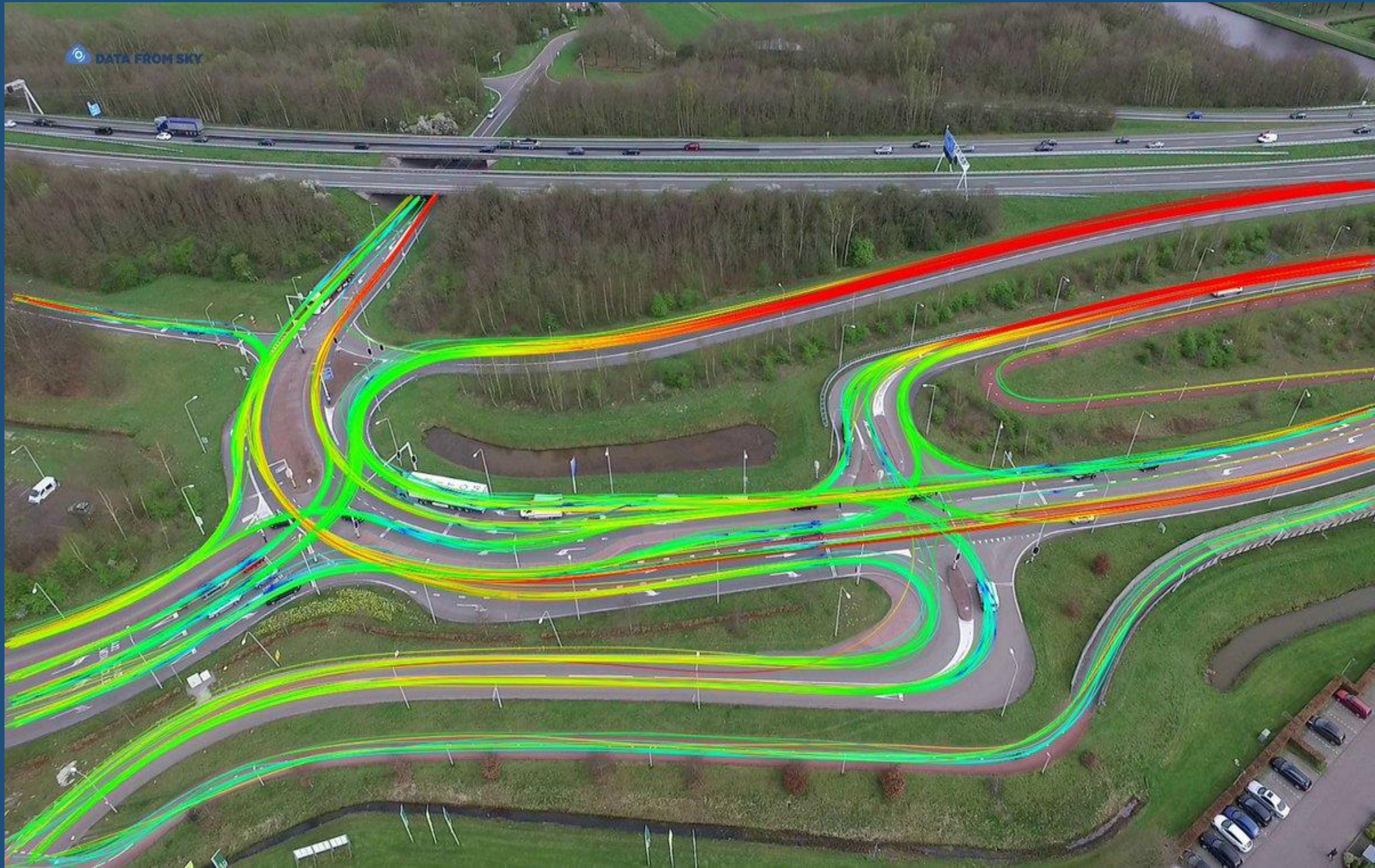
- openMP on DTA of ODE
- Discretization LWR model

What’s Dynamic Traffic Assignment ?

Dynamic traffic assignment is aimed at allocating traffic flow to every path and making their travel time minimized over the time.

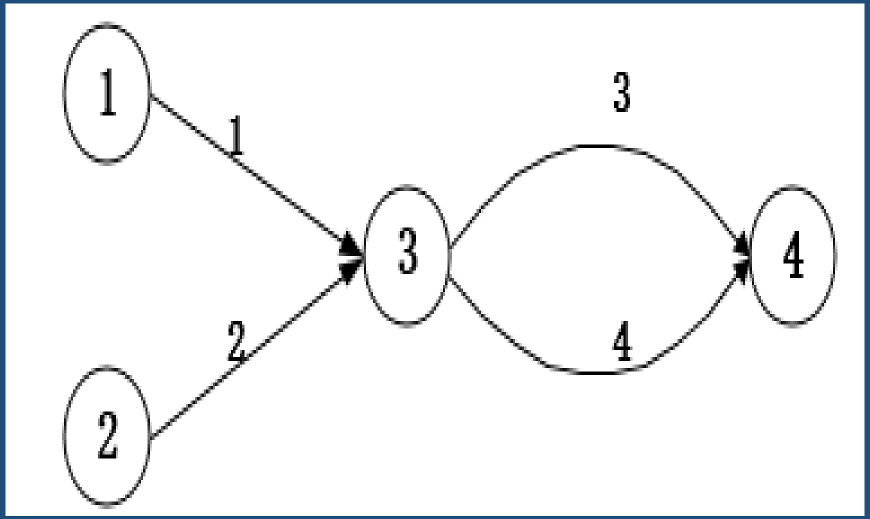
Dynamic traffic assignment belongs to traffic planning, it plays an important role in Intelligent Transportation System

Such as Route Guidance in Google map
Heat Map in Baidu map



Introduction

Dynamic traffic assignment is the positive modeling of time-varying flows of automobiles on road network consistent with established traffic flow theory and travel demand theory.



Abstract Network

- Nodes: origin or destination
- Links : road
- Origin-Destination Pair
- Time cost = Delay = Travel Time

Introduction

Continuous Time Dynamic User Equilibrium (DUE)

- Users choose the path with same and minimum travel time

Desired solution:

- The departure rate function for each path
- The corresponding cost function

Continuous Time Dynamic User Equilibrium (DUE)

For each individual, compared with your current travel cost:

- If there is another path will lessen your travel cost, you switch!
- If there is another departure time will lessen your travel cost, you switch!
- Facing with a new scenario, go back to the first two steps.
- ……

Until the Nash equilibrium is reached!

Nash equilibrium

- In game theory, the Nash equilibrium, named after American mathematician John Forbes Nash Jr., is a solution concept of a non-cooperative game involving two or more players in which each player is assumed to know the equilibrium strategies of the other players, and no player has anything to gain by changing only their own strategy.

Solution?

- How's the transportation system going to look like under that equilibrium?

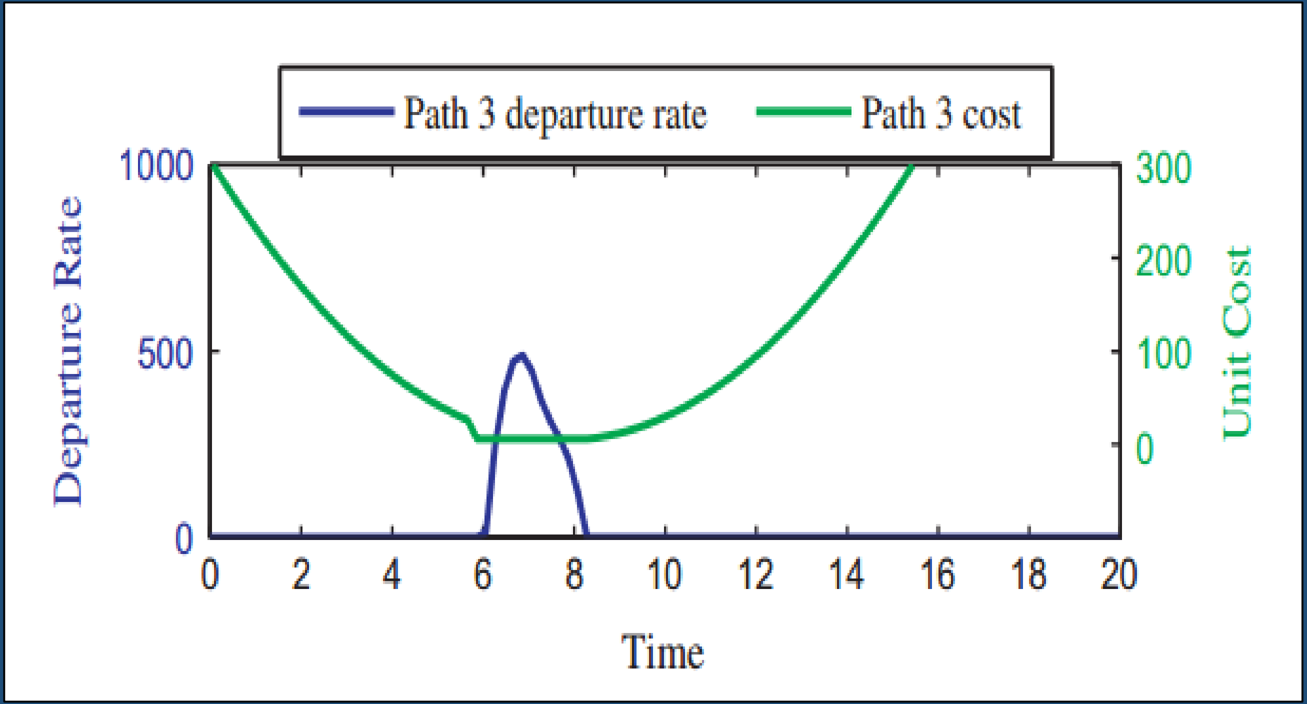


Fig Departure rates and corresponding travel cost in the DUE solution

Solution?

- To know the equilibrium strategies of the other players:

  Dynamic Network Loading: The problem of finding link activity when travel demand and departure rates (path flows) are known is commonly referred to as the dynamic network loading problem.

- To find the equilibrium:

  Differential Variational Inequality (dVI)

Progress

➢ Part 1: Dynamic Network Loading(Friesz,2011)

Link state equation

$$\frac{dx_{a_1}^p(t)}{dt} = h_p(t) - g_{a_1}^p(t) \ \ \forall p \in P$$

$$\frac{dx_{a_i}^p(t)}{dt} = g_{a_{i-1}}^p(t) - g_{a_i}^p(t) \ \ \forall p \in P, i \in [2, num(p)]$$

$$\frac{dg_{a_i}^p(t)}{dt} = r_{a_i}^p(t) \ \forall p \in P, i \in [1, num(p)]$$

Medium equation (coming from Taylor expansion)

$$\frac{dr_{a_1}^p(t)}{dt} = R_{a_1}^p(x, g, r, h) \ \ \forall p \in P$$

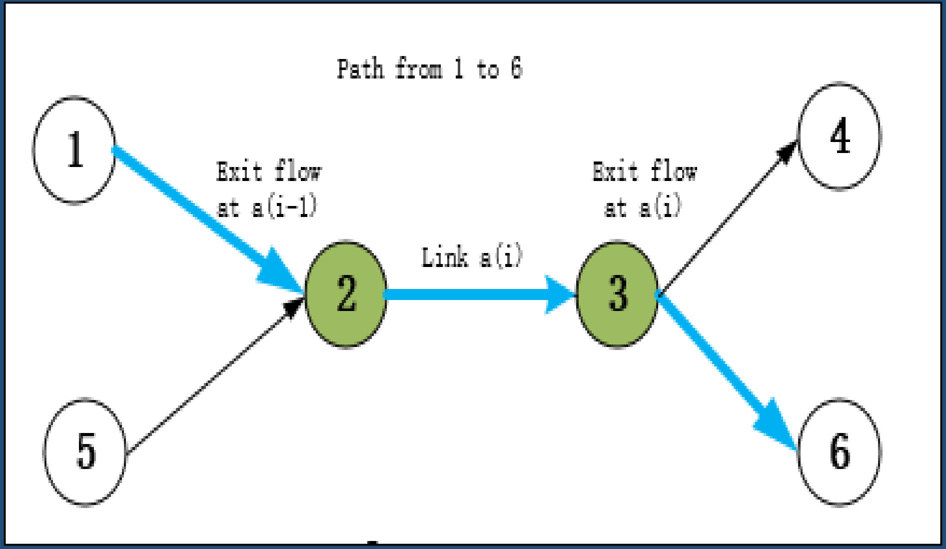
$$\frac{dr_{a_i}^p(t)}{dt} = R_{a_i}^p(x, g, r) \ \ \forall p \in P, i \in [2, num(p)]$$

Initial conditions

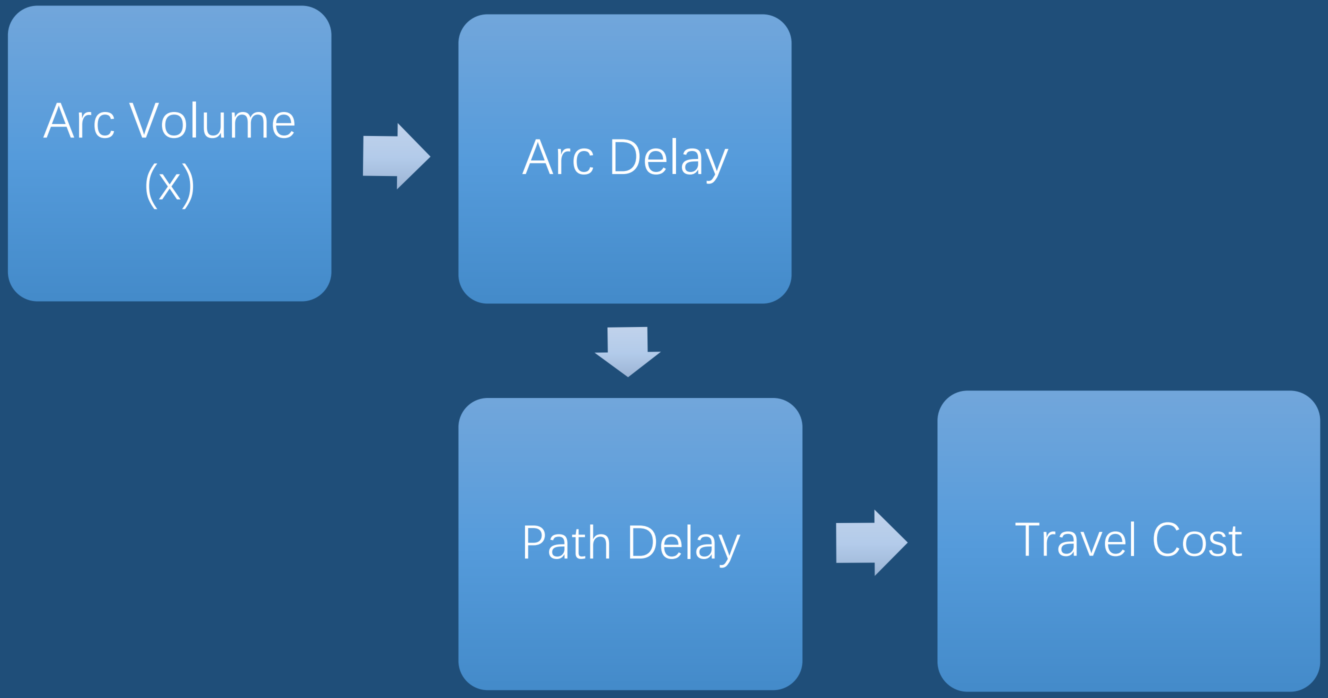
$$x_{a_i}^p((\tau - 1) \cdot \Delta) = x_{a_i}^{p,0} \ \ \forall p \in P, i \in [1, num(p)]$$

$$g_{a_i}^p((\tau - 1) \cdot \Delta) = 0 \ \ \forall p \in P, i \in [1, num(p)]$$

$$r_{a_i}^p((\tau - 1) \cdot \Delta) = 0 \ \ \forall p \in P, i \in [1, num(p)]$$



Flow Propagation

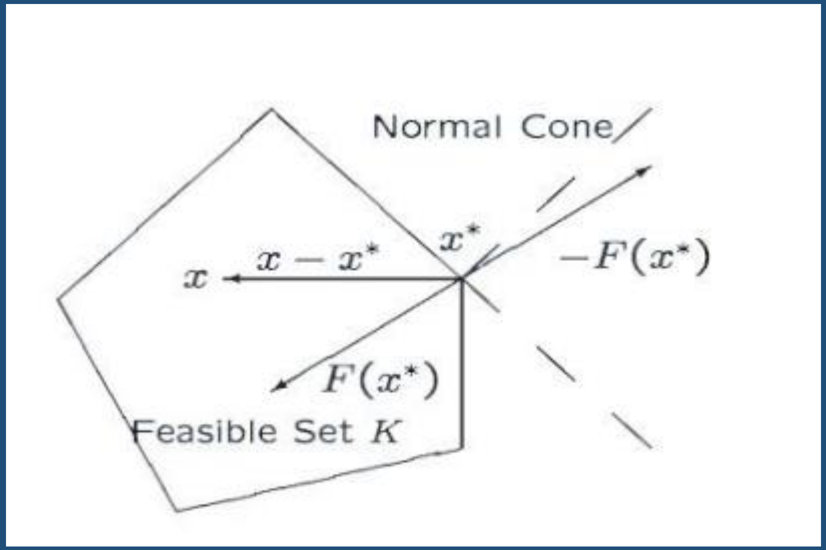

Arc Volume (x)
Arc Delay
Path Delay
Travel Cost

Progress

- Part 2: Convert DUE to DVI

$$\text{DVI}(\Psi, \Lambda, [t_0, t_f]): find\ h^* \in \Lambda_0\ such\ that$$

$$\begin{cases} \sum_{p \in P} \int_{t_0}^{t_f} \Psi_p(t, h^*)(h - h^*)dt \geq 0\ \forall h \in \Lambda \\ where\ \Lambda = h \geq 0 : \frac{dy_{ij}}{dt} = \sum_{p \in P_{ij}} h_p(t), y_{ij}(0) = 0, y_{ij}(t_f) = Q_{ij} \end{cases}$$



Progress

- Part 3: Solve DVI by Fixed-Point Iteration

Equal solution $$h^* = P_{\Lambda}[h^* - \alpha \Psi_p(t, h^*)]$$

For each iteration step $$\sum_{p \in P_{ij}} \int_{t_0}^{t_f} \left[h_p^k(t) - \alpha \Psi(t, h_p^k) + v_{ij}\right]_+ = Q_{ij}$$

Update new departure rate $$h_p^{k+1} = \left[h_p^k(t) - \alpha \Psi(t, h_p^k) + v_{ij}\right]_+$$

Algorithm 1 Computing ODE and fixed point iteration

Initialization: path, timespan, arc, h0, Q(demand), epsilon (tolerance) et.

while condition is true ▲ |hk - hk+1| is larger than tolerance

solve ODE to get link volume

get link delay

get effective path delay

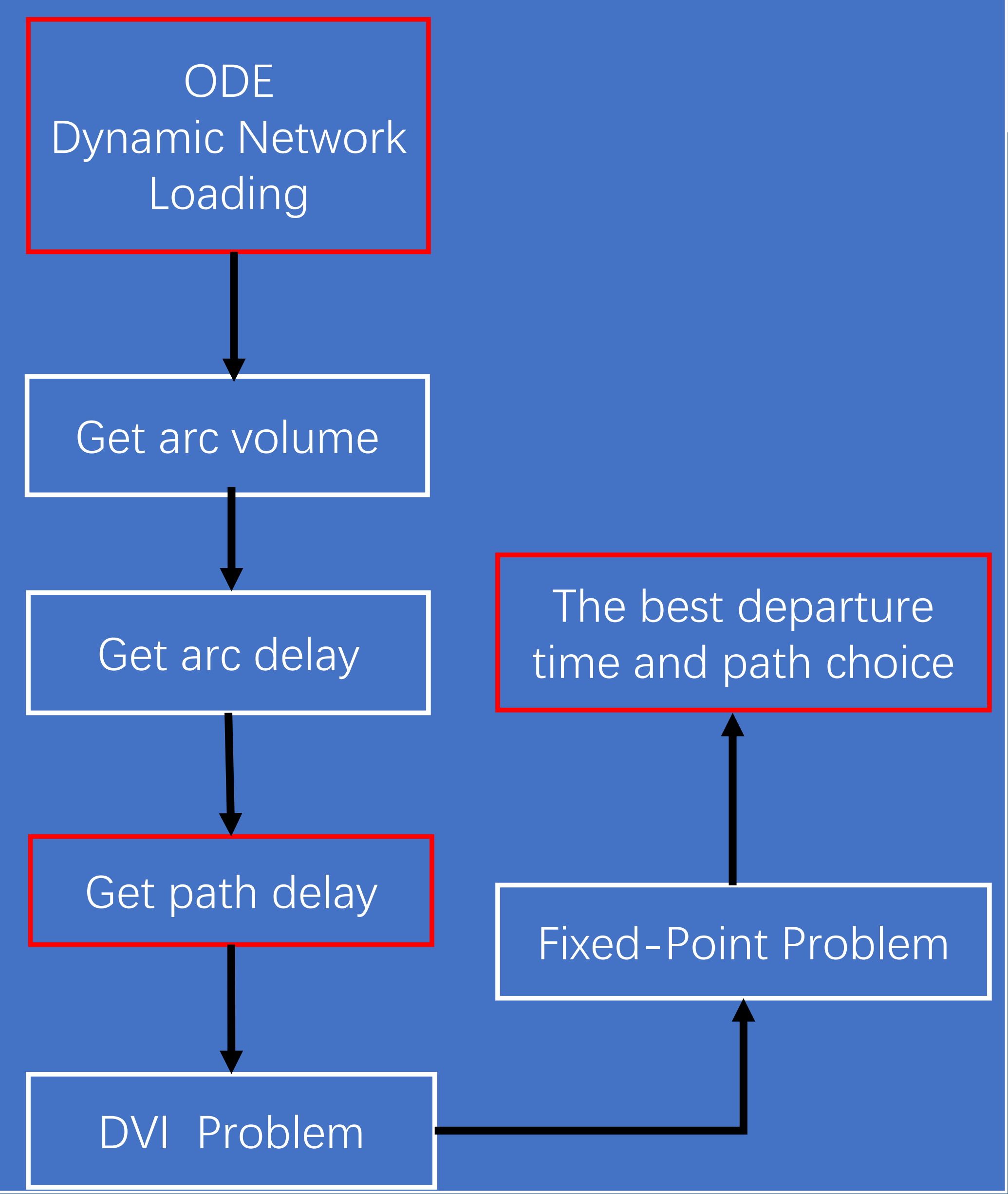
solve v

update hk+1

end while

Output: Phi, hk

Flow chart



Example: Small graph

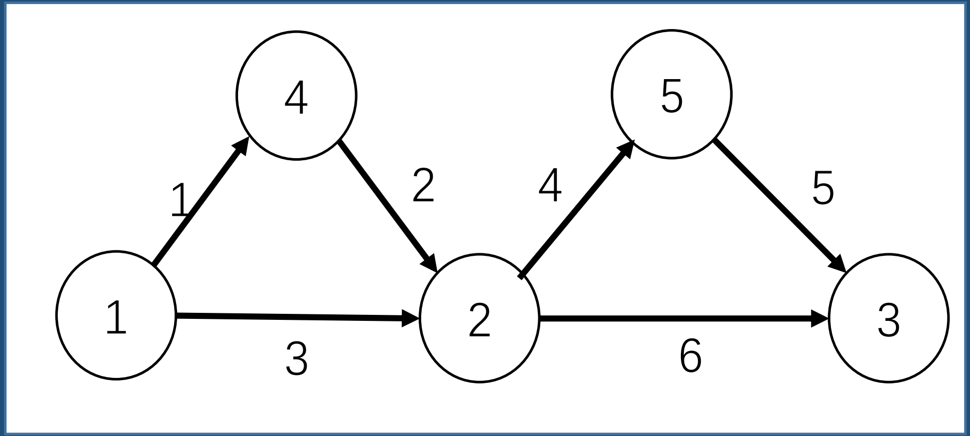


Fig. Small Network

Arc	Jam density (vehicles/km)	Free flow speed (km/5min)	Length (km)
1	800	6	4
2	800	6	8
3	800	8	4
4	800	8	10
5	1000	8	8
6	600	6	10

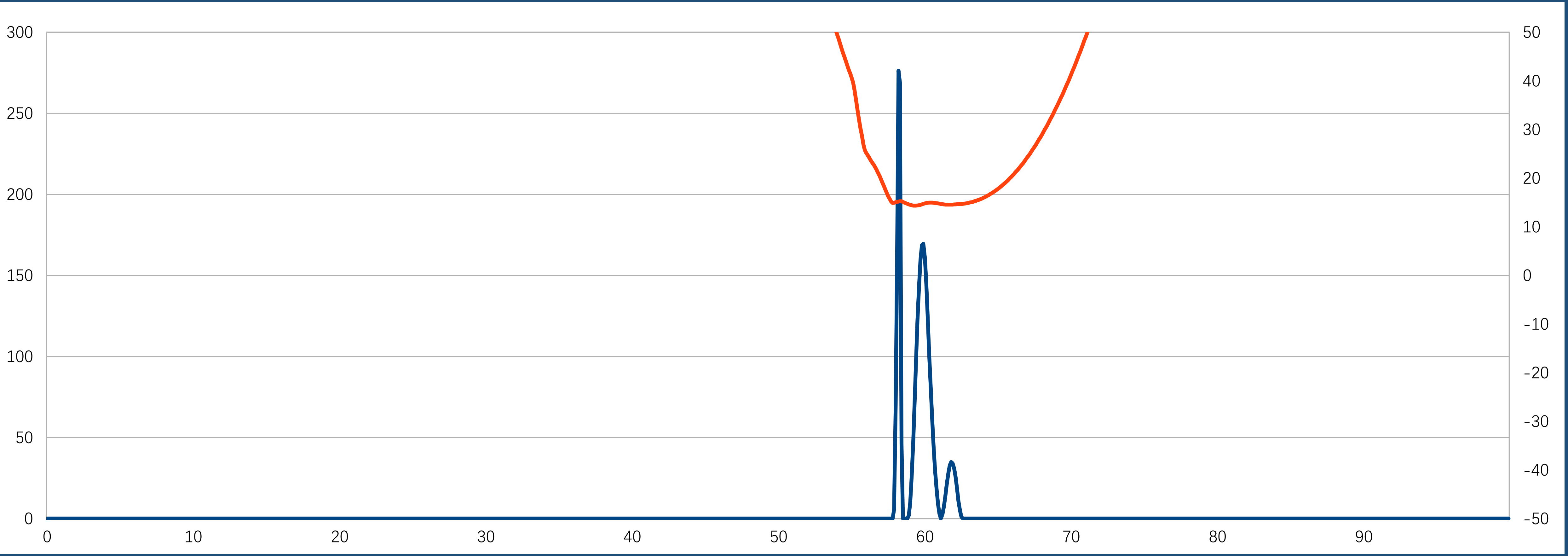
```
double TA[2] = {75.0,50.0};
double Q[2] = {400,200};


//initializing arc info
std::vector< arc_type > arc= {{0,800,6,4},{1,800,6,8},{2,800,8,4},{3,800,8,10},{4,1000,8,8},{5,600,6,10}};

//initializing path info
std::vector<od_pair_type> DATA_SET ={{{arc[2],arc[5]},{arc[0],arc[1],arc[5]},{arc[0],arc[1],arc[3],arc[4]},{arc[2],arc[3],arc[4]}},
                        {{arc[5]},{arc[3],arc[4]}}};
```

Example: Small graph

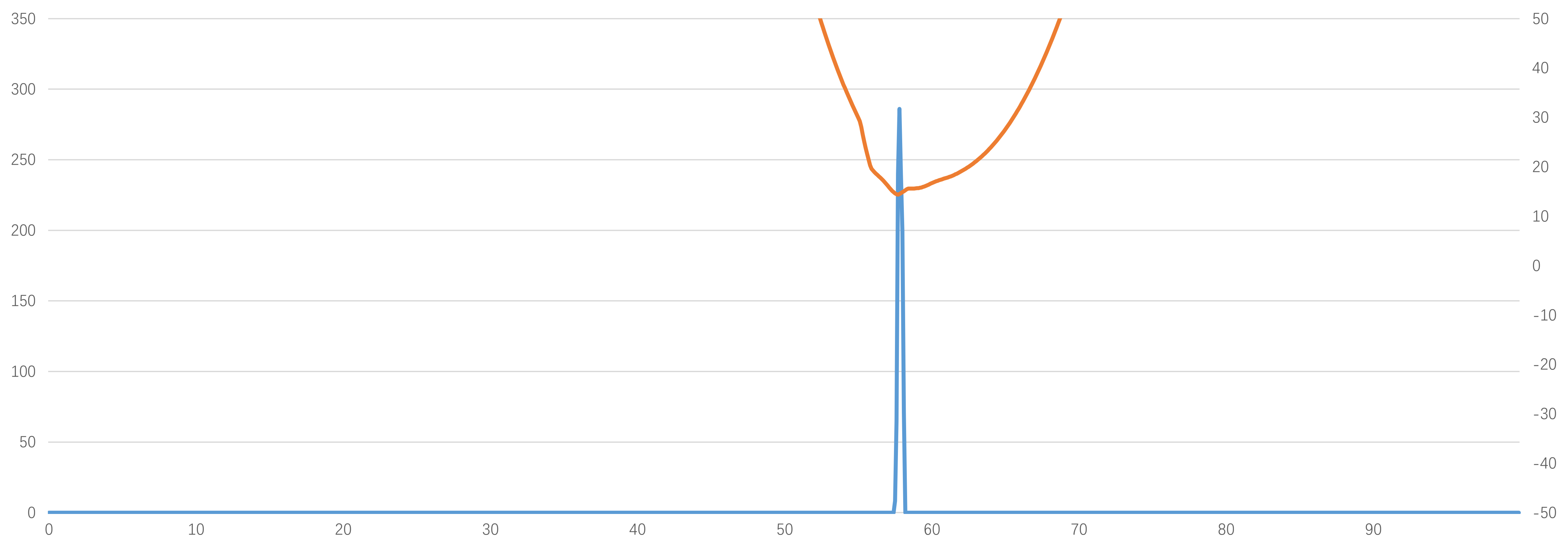
```
Printing the range of v for each od pair:
14.807292,14.807347
9.542000,9.545713
for 0 th h, ND/NX is 0.000004    NX is 77217.074806    ND is 0.301055    Q is 386.794661
for 1 th h, ND/NX is nan    NX is 0.000000    ND is 0.000000    Q is 0.000000
for 2 th h, ND/NX is nan    NX is 0.000000    ND is 0.000000    Q is 0.000000
for 3 th h, ND/NX is 0.000010    NX is 454.750382    ND is 0.004661    Q is 13.142970
for 4 th h, ND/NX is 0.000001    NX is 32669.432778    ND is 0.037689    Q is 199.743774
for 5 th h, ND/NX is nan    NX is 0.000000    ND is 0.000000    Q is 0.000000
```

THE UNIVERSITY OF TENNESSEE OAK RIDGE National Laboratory NSF JICS Joint Institute for Computational Sciences

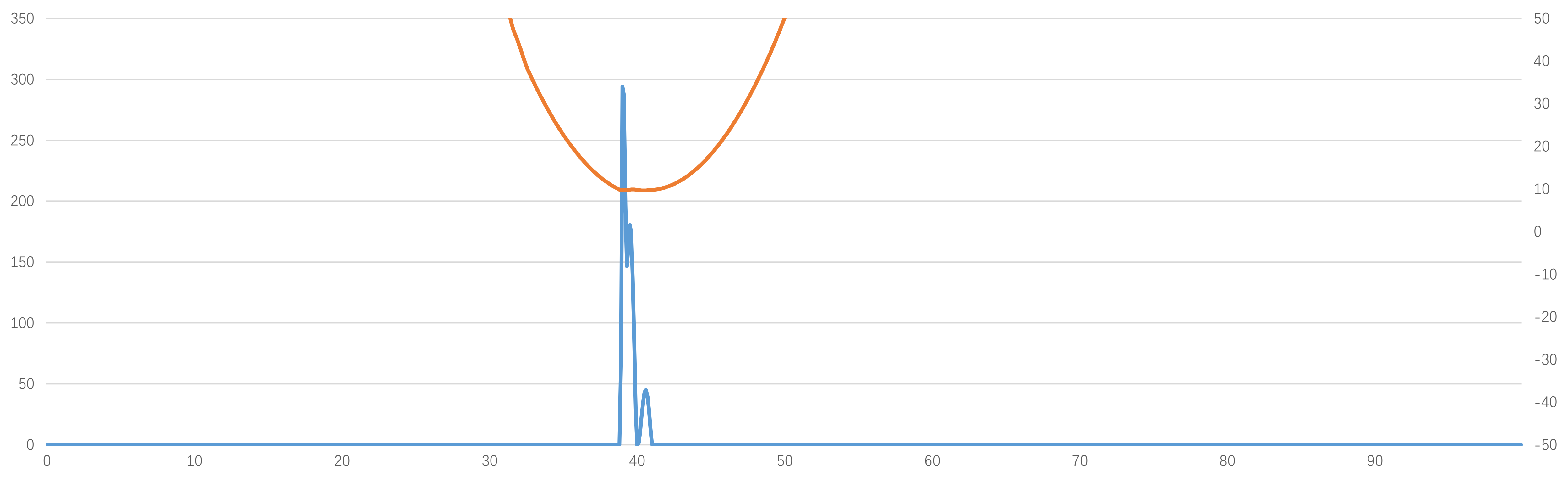
Example: Path 1



Example: Path 4



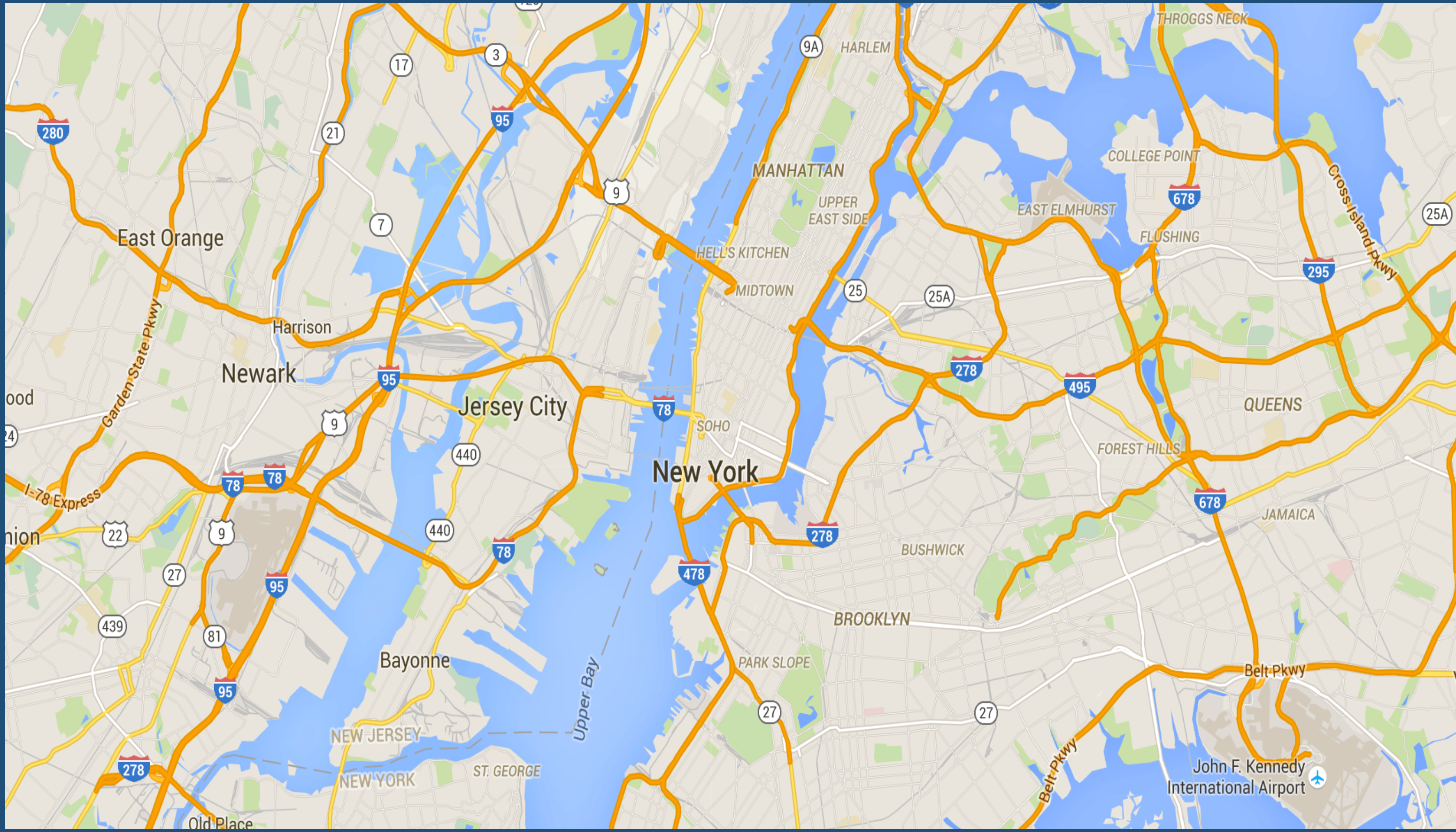
Example: Path 5



Bottleneck

If given a large network

Not fast enough!



Using openMP to implement parallel computing

```
//Initializing from 2m to 3m-1
m_counter=0;
h_counter=0;
for(int a=0;a<DATA_SET.size();a++)
{
  for(int b=0;b<DATA_SET[a].size();b++)
  {
    for(int c=0;c<DATA_SET[a][b].size();c++)
    {
      double total_x=0;
      double total_dx=0;
      int arc_num=int(DATA_SET[a][b][c][0]);
      for(int i=0;i<ARC_MAP[arc_num].size();i++)
      {
        total_x+=x[ARC_MAP[arc_num][i]];
        total_dx+=dxdt[ARC_MAP[arc_num][i]];
      }
      if(c==0)
      {
        dxdt[m_counter+2*m]=R(total_x,total_dx,x[m_counter+m],x[m_counter+2*m],
          h_spline[h_counter](t),DATA_SET[a][b][c]);
        h_counter++;
      }else{
        dxdt[m_counter+2*m]=R(total_x,total_dx,x[m_counter+m],x[m_counter+2*m],
          x[m_counter+m-1],DATA_SET[a][b][c]);
      }
      m_counter++;
    }
  }
}
```

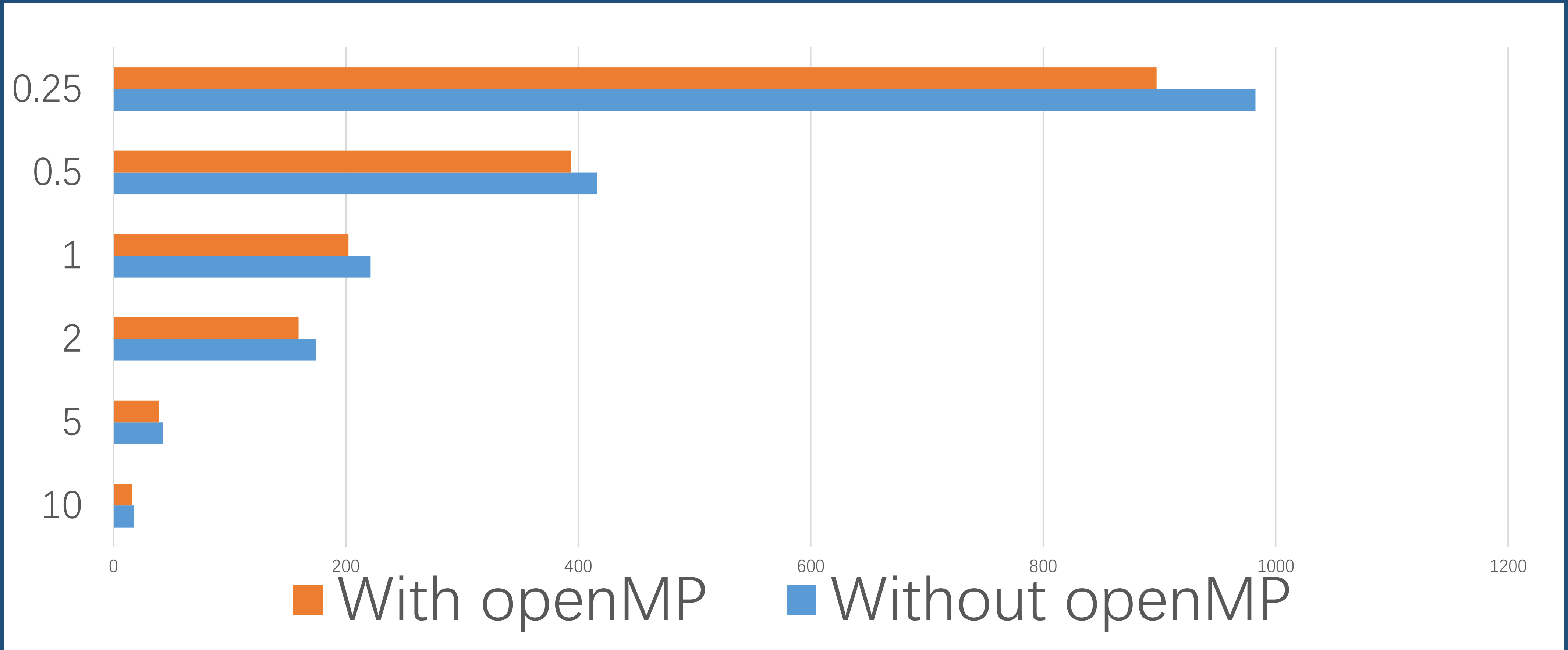
```
//Initializing from 2m to 3m-1
#pragma omp parallel for
for(int i=0;i<m;i++)
{
  double total_x=0;
  double total_dx=0;
  std::vector<double> current_arc=DATA_SET[M_MAP[i][0]][M_MAP[i][1]][M_MAP[i][2]];
  int arc_num=int(current_arc[0]);
  for(int j=0;j<ARC_MAP[arc_num].size();j++)
  {
    total_x+=x[ARC_MAP[arc_num][j]];
    total_dx+=dxdt[ARC_MAP[arc_num][j]];
  }
  if(M_MAP[i][2]==0)
  {
    dxdt[i+2*m]=R(total_x,total_dx,x[i+m],x[i+2*m],
      h_spline[INDEX_MAP[M_MAP[i][0]][M_MAP[i][1]]](t),current_arc);
  }else{
    dxdt[i+2*m]=R(total_x,total_dx,x[i+m],x[i+2*m],
      x[i+m-1],current_arc);
  }
}
```

Using openMP to implement parallel computing

- ◆ODE calculation
- ◆Linear search for v

- ◆12 cores, 24 threads

Performance on small graph



THE UNIVERSITY OF TENNESSEE

OAK RIDGE National Laboratory

NSF

JICS Joint Institute for Computational Sciences

Sioux Falls Network

Considering 10 OD pair and 30 paths

```
double TA[10] = {75.0,50.0,75.0,75.0,75.0,50.0,60.0,75.0,75.0,80.0};
double Q[10] = {400,400,400,400,400,400,400,400,400,400};
//initializing arc info
std::vector< arc_type > arc= {{0,800,6,4},{1,800,6,8},{2,800,8,4},{3,800,8,10},{4,1000,8,8},{5,600,6,10},
                              {6,800,6,4},{7,800,6,8},{8,800,8,4},{9,800,8,10},{10,1000,8,8},{11,600,6,10},
                              {12,800,6,4},{13,800,6,8},{14,800,8,4},{15,800,8,10},{16,1000,8,8},{17,600,6,10},
                              {18,800,6,4},{19,800,6,8},{20,800,8,4},{21,800,8,10},{22,1000,8,8},{23,600,6,10},
                              {24,800,6,4},{25,800,6,8},{26,800,8,4},{27,800,8,10},{28,1000,8,8},{29,600,6,10},
                              {30,800,6,4},{31,800,6,8},{32,800,8,4},{33,800,8,10},{34,1000,8,8},{35,600,6,10},
                              {36,800,6,4},{37,800,6,8},{38,800,8,4},{39,800,8,10},{40,1000,8,8},{41,600,6,10},
                              {42,800,6,4},{43,800,6,8},{44,800,8,4},{45,800,8,10},{46,1000,8,8},{47,600,6,10},
                              {48,800,6,4},{49,800,6,8},{50,800,8,4},{51,800,8,10},{52,1000,8,8},{53,600,6,10},
                              {54,800,6,4},{55,800,6,8},{56,800,8,4},{57,800,8,10},{58,1000,8,8},{59,600,6,10},
                              {60,800,6,4},{61,800,6,8},{62,800,8,4},{63,800,8,10},{64,1000,8,8},{65,600,6,10},
                              {66,800,6,4},{67,800,6,8},{68,800,8,4},{69,800,8,10},{70,1000,8,8},{71,600,6,10},
                              {72,800,6,4},{73,800,6,8},{74,800,8,4},{75,800,8,10},{76,1000,8,8},{77,600,6,10},
                              {78,800,6,4},{79,800,6,8},{80,800,8,4},{81,800,8,10}};

//initializing path info
std::vector<od_pair_type> DATA_SET ={
  {{arc[2],arc[7],arc[37],arc[39],arc[75],arc[64]},{arc[2],arc[7],arc[36],arc[34],arc[41],arc[45],arc[59]},
    {arc[2],arc[7],arc[36],arc[32],arc[29],arc[49],arc[53],arc[59]},{arc[1],arc[4],arc[16],arc[22],arc[49],arc[53],arc[59]},
    {arc[3],arc[4],arc[16],arc[20],arc[18],arc[56]}},//5
  {{arc[4],arc[16],arc[22],arc[49],arc[53],arc[59]},{arc[4],arc[16],arc[20],arc[18],arc[56]},{arc[4],arc[16],arc[22],arc[55],arc[56]}},//3
  {{arc[10],arc[34],arc[42],arc[73],arc[75],arc[64]},{arc[10],arc[27],arc[30],arc[53],arc[59]},{arc[9],arc[12],arc[16],arc[22],arc[49],arc[53]
    {arc[9],arc[13],arc[25],arc[30],arc[53],arc[59]}},//4
  {{arc[13],arc[25],arc[28],arc[46],arc[69],arc[64]},{arc[12],arc[16],arc[22],arc[49],arc[53],arc[59]}},//2
  {{arc[54],arc[56]},{arc[17],arc[22],arc[49],arc[53],arc[59]},{arc[18],arc[55],arc[49],arc[53],arc[59]}},//3
  {{arc[25],arc[28],arc[46],arc[69],arc[64]},{arc[25],arc[28],arc[46],arc[68]},{arc[24],arc[22],arc[49],arc[53],arc[59]}},//3
  {{arc[34],arc[42],arc[73],arc[75],arc[64]},{arc[32],arc[29],arc[49],arc[53],arc[59]},{arc[32],arc[28],arc[46],arc[68]}},//3
  {{arc[39],arc[75],arc[64]},{arc[39],arc[75],arc[69],arc[68]}},//2
  {{arc[46],arc[68]},{arc[46],arc[69],arc[64]},{arc[45],arc[59]}},//3
  {{arc[49],arc[53],arc[59]},{arc[50],arc[56]}}//2
};
```

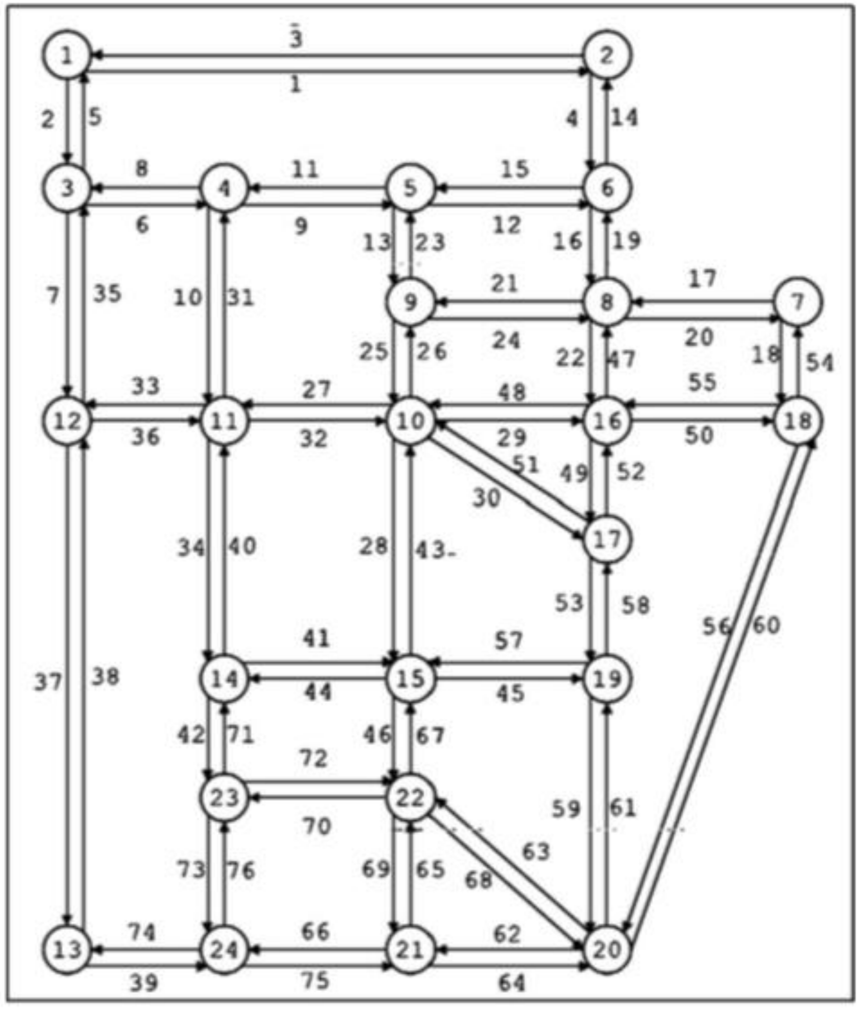


Fig. 1. Sioux Falls network.

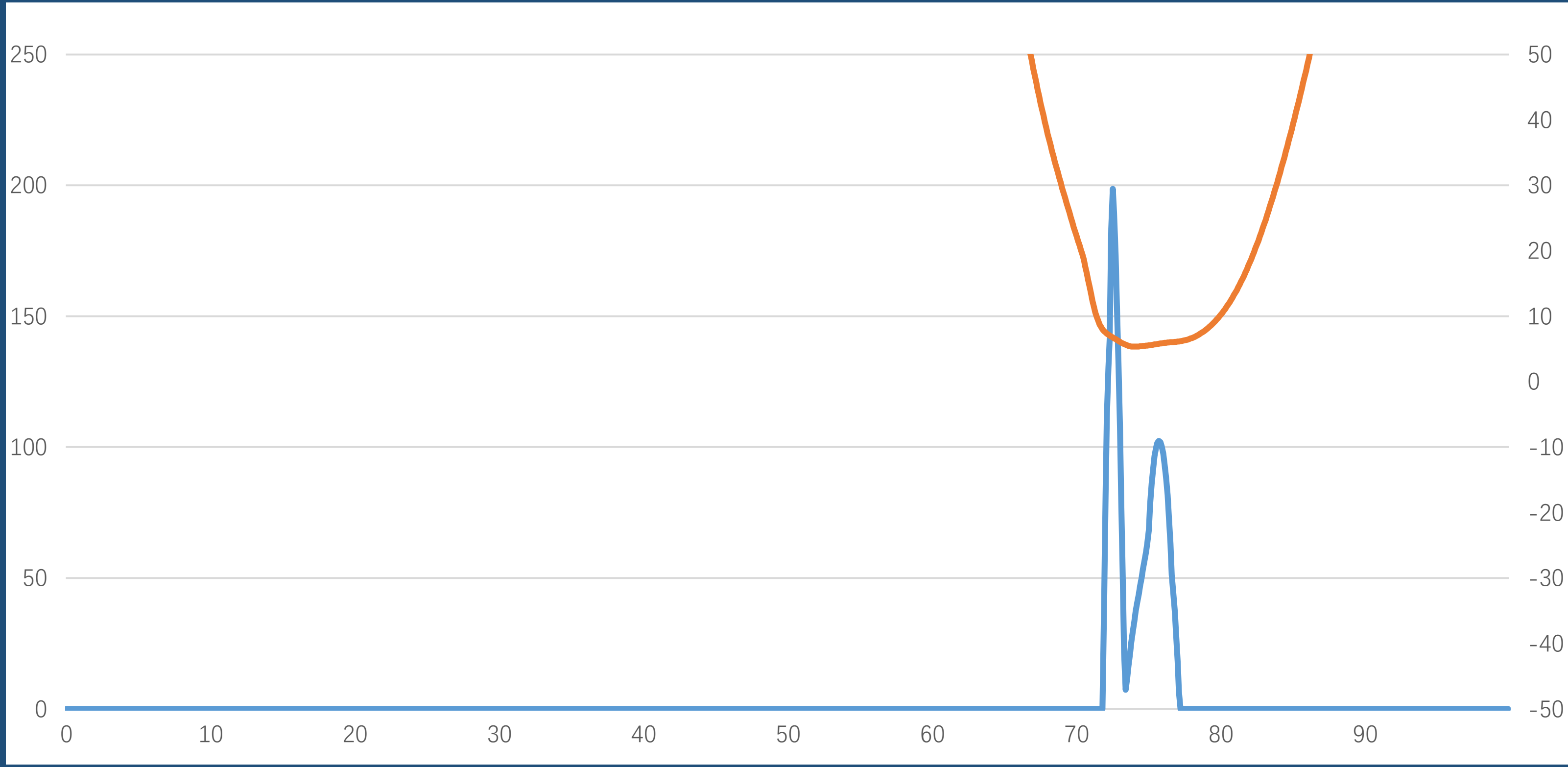
Sioux Falls Network

Considering 10 OD pair and 30 paths

```
For the 0 th path, the integral of departure rate of final hk is 0.000000
For the 1 th path, the integral of departure rate of final hk is 0.000000
For the 2 th path, the integral of departure rate of final hk is 0.000000
For the 3 th path, the integral of departure rate of final hk is 0.000000
For the 4 th path, the integral of departure rate of final hk is 399.945611
For the 5 th path, the integral of departure rate of final hk is 0.000000
For the 6 th path, the integral of departure rate of final hk is 394.704573
For the 7 th path, the integral of departure rate of final hk is 4.835398
For the 8 th path, the integral of departure rate of final hk is 235.110113
For the 9 th path, the integral of departure rate of final hk is 164.270061
For the 10 th path, the integral of departure rate of final hk is 0.000000
For the 11 th path, the integral of departure rate of final hk is 0.000000
For the 12 th path, the integral of departure rate of final hk is 201.004450
For the 13 th path, the integral of departure rate of final hk is 198.743123
For the 14 th path, the integral of departure rate of final hk is 399.917744
For the 15 th path, the integral of departure rate of final hk is 0.000000
For the 16 th path, the integral of departure rate of final hk is 0.000000
For the 17 th path, the integral of departure rate of final hk is 0.000000
For the 18 th path, the integral of departure rate of final hk is 400.014653
For the 19 th path, the integral of departure rate of final hk is 0.000000
For the 20 th path, the integral of departure rate of final hk is 0.000000
For the 21 th path, the integral of departure rate of final hk is 0.000000
For the 22 th path, the integral of departure rate of final hk is 399.659963
For the 23 th path, the integral of departure rate of final hk is 270.721623
For the 24 th path, the integral of departure rate of final hk is 128.991122
For the 25 th path, the integral of departure rate of final hk is 399.821417
For the 26 th path, the integral of departure rate of final hk is 0.000000
For the 27 th path, the integral of departure rate of final hk is 0.000000
For the 28 th path, the integral of departure rate of final hk is 0.000000
For the 29 th path, the integral of departure rate of final hk is 399.826259
```

Sioux Falls Network :path no.30

Considering 10 OD pair and 30 paths



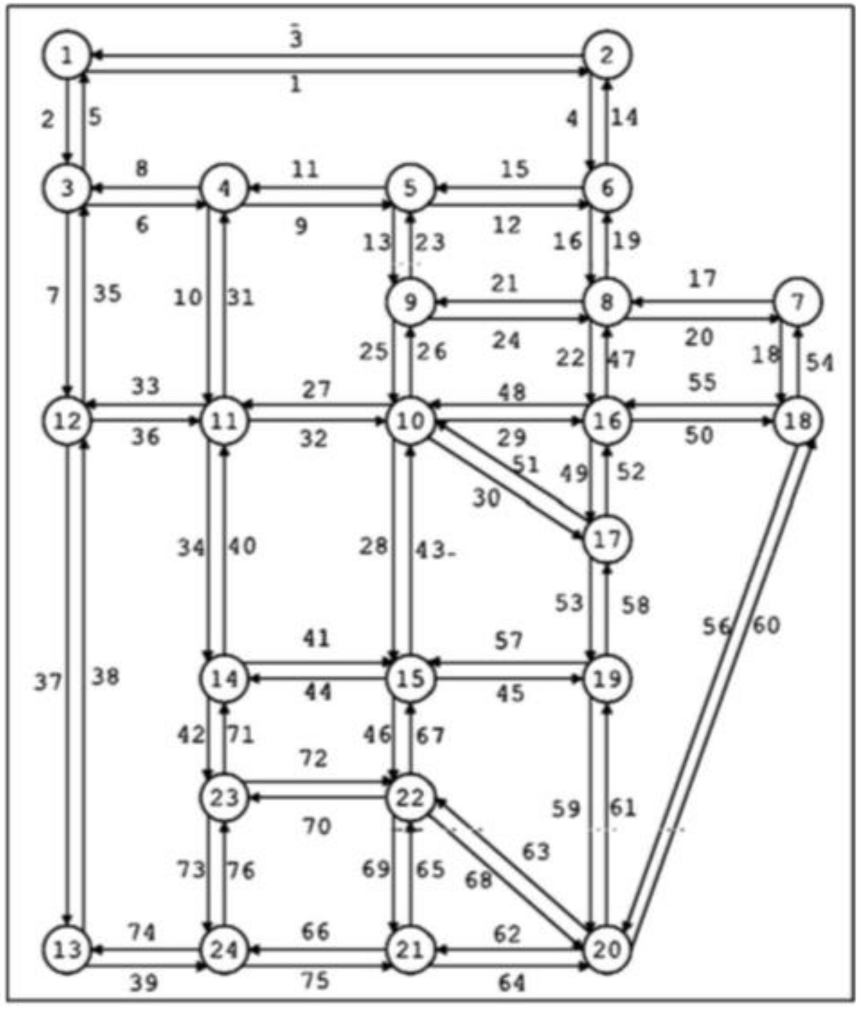
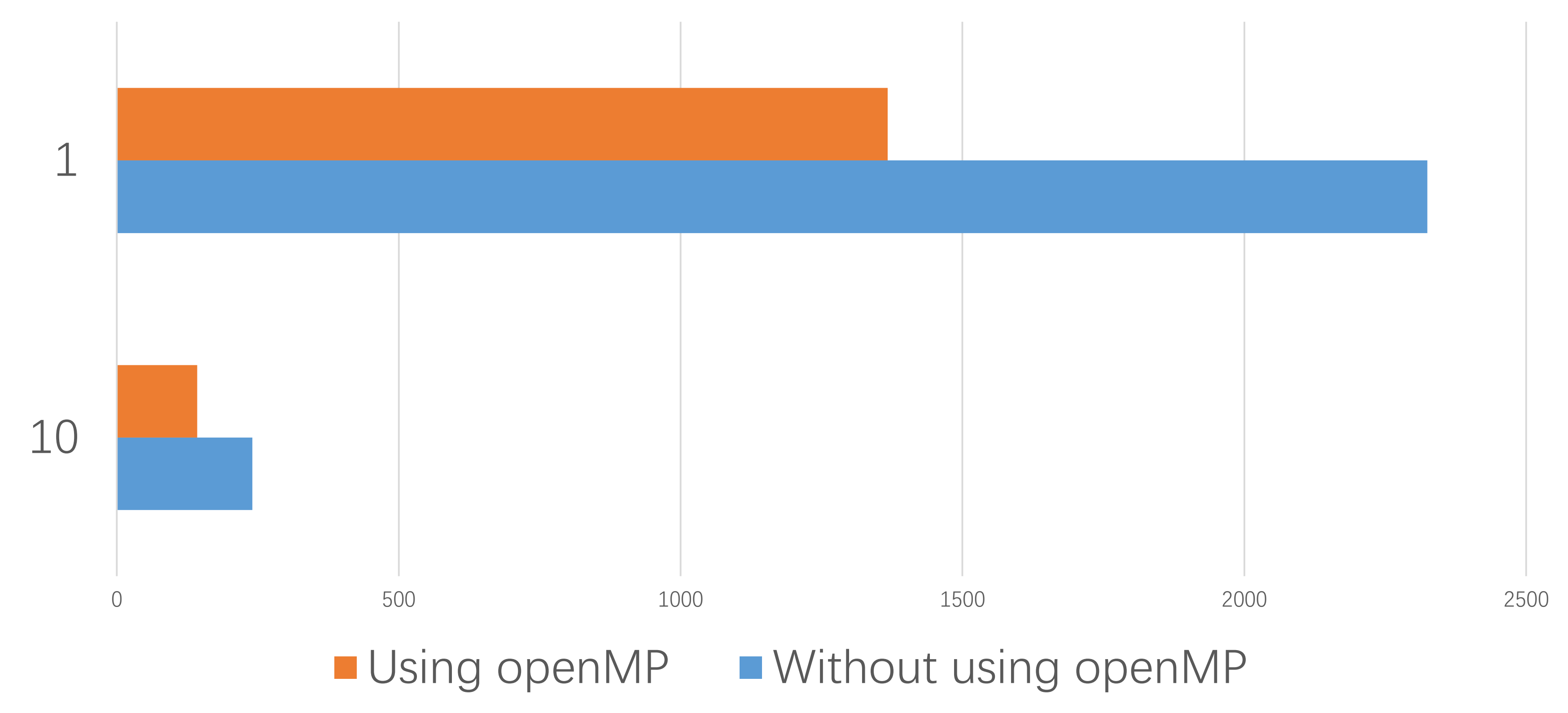


Fig. 1. Sioux Falls network.

Sioux Falls Network

Considering 10 OD pairs and 30 paths



Sioux Falls Network

Considering 23 OD pairs and 232 paths

17	19	14	3								
17	19	15	11	8	5						
17	21	23	11	8	5						
17	21	25	27	31	8	5					
18	55	47	19	14	3						
18	55	47	21	23	11	8	5				
18	55	48	26	23	11	8	5				
18	55	48	26	23	12	14	3				
18	55	48	27	31	8	5					
18	55	48	27	33	35	5					
18	55	49	51	27	31	9	12	14	3		
18	55	49	51	27	33	35	5				
18	56	61	57	43	26	23	11	8	5		
18	56	61	57	43	27	31	8	5			
18	56	62	65	67	43	27	31	8	5		
18	56	62	66	74	38	35	6	9	12	14	3
18	56	63	67	43	27	33	35	5			
18	56	63	7	71	4	31	8	5			
18	56	63	7	71	41	43	27	33	35	5	

17	19	14						
17	19	15	11	8	5	1		
17	21	23	11	8	5	1		
17	21	23	12	14				
17	21	25	27	31	8	5	1	
18	55	47	19	14				
18	55	47	21	23	11	8	5	1
18	55	48	26	23	11	8	5	1
18	55	48	26	23	12	14		
18	55	48	27	31	8	5	1	
18	55	48	27	33	35	5	1	

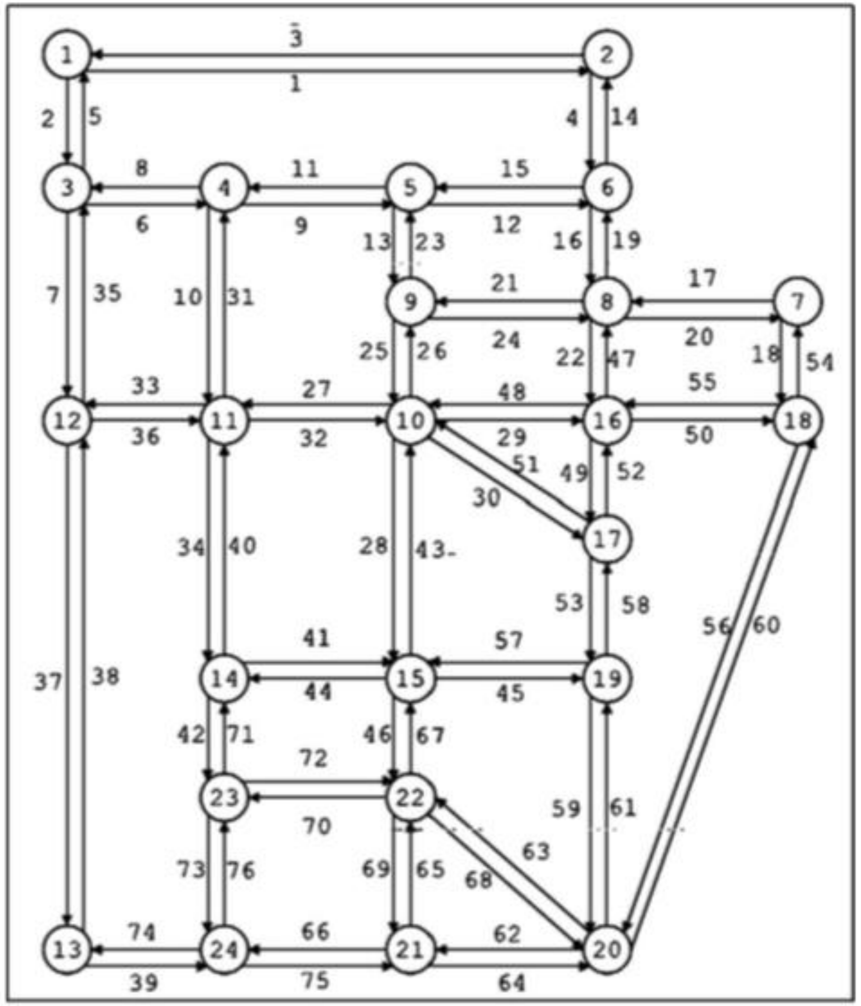
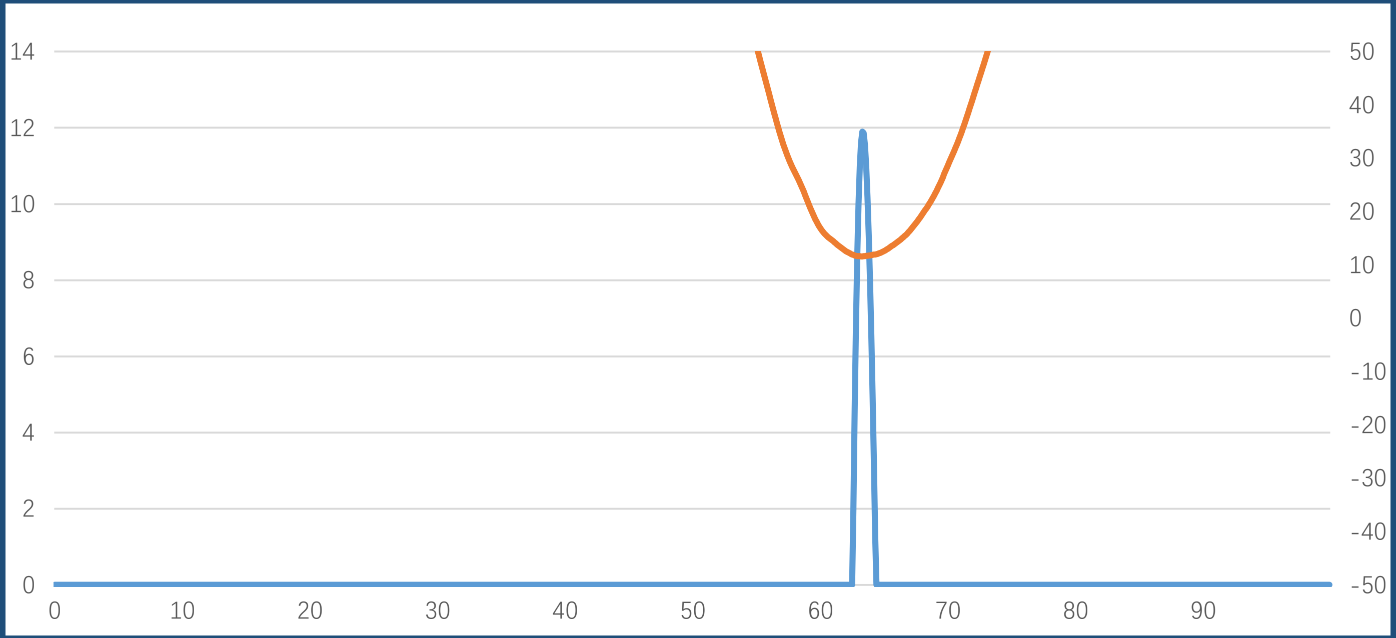


Fig. 1. Sioux Falls network.

Sioux Falls Network

Considering 23 OD pairs and 232 paths

Path no.40:




```
For the 0 th path, the integral of departure rate of final hk is 20.008372
For the 1 th path, the integral of departure rate of final hk is 0.000000
For the 2 th path, the integral of departure rate of final hk is 0.000000
For the 3 th path, the integral of departure rate of final hk is 20.003733
For the 4 th path, the integral of departure rate of final hk is 0.000000
For the 5 th path, the integral of departure rate of final hk is 0.000000
For the 6 th path, the integral of departure rate of final hk is 19.992994
For the 7 th path, the integral of departure rate of final hk is 0.000000
For the 8 th path, the integral of departure rate of final hk is 0.000000
For the 9 th path, the integral of departure rate of final hk is 20.022541
For the 10 th path, the integral of departure rate of final hk is 19.987229
For the 11 th path, the integral of departure rate of final hk is 0.000000
For the 12 th path, the integral of departure rate of final hk is 0.000000
For the 13 th path, the integral of departure rate of final hk is 20.010012
For the 14 th path, the integral of departure rate of final hk is 0.000000
For the 15 th path, the integral of departure rate of final hk is 0.000000
For the 16 th path, the integral of departure rate of final hk is 0.000000
For the 17 th path, the integral of departure rate of final hk is 0.000000
For the 18 th path, the integral of departure rate of final hk is 0.000000
For the 19 th path, the integral of departure rate of final hk is 0.000000
For the 20 th path, the integral of departure rate of final hk is 0.000000
For the 21 th path, the integral of departure rate of final hk is 0.000000
For the 22 th path, the integral of departure rate of final hk is 0.000000
For the 23 th path, the integral of departure rate of final hk is 0.000000
For the 24 th path, the integral of departure rate of final hk is 0.000000
For the 25 th path, the integral of departure rate of final hk is 0.000000
For the 26 th path, the integral of departure rate of final hk is 0.000000
For the 27 th path, the integral of departure rate of final hk is 0.000000
For the 28 th path, the integral of departure rate of final hk is 0.000000
For the 29 th path, the integral of departure rate of final hk is 19.999220
For the 30 th path, the integral of departure rate of final hk is 0.000000
For the 31 th path, the integral of departure rate of final hk is 0.000000
For the 32 th path, the integral of departure rate of final hk is 0.000000
For the 33 th path, the integral of departure rate of final hk is 0.000000
For the 34 th path, the integral of departure rate of final hk is 0.000000
For the 35 th path, the integral of departure rate of final hk is 0.000000
For the 36 th path, the integral of departure rate of final hk is 0.000000
For the 37 th path, the integral of departure rate of final hk is 0.000000
For the 38 th path, the integral of departure rate of final hk is 5.664875
For the 39 th path, the integral of departure rate of final hk is 14.337562
For the 40 th path, the integral of departure rate of final hk is 0.000000
For the 41 th path, the integral of departure rate of final hk is 0.000000
For the 42 th path, the integral of departure rate of final hk is 0.000000
For the 43 th path, the integral of departure rate of final hk is 0.000000
For the 44 th path, the integral of departure rate of final hk is 0.000000
For the 45 th path, the integral of departure rate of final hk is 8.331890
For the 46 th path, the integral of departure rate of final hk is 0.000000
For the 47 th path, the integral of departure rate of final hk is 11.654051
For the 48 th path, the integral of departure rate of final hk is 0.000000
For the 49 th path, the integral of departure rate of final hk is 0.000000
For the 50 th path, the integral of departure rate of final hk is 0.000000
For the 51 th path, the integral of departure rate of final hk is 7.753155
For the 52 th path, the integral of departure rate of final hk is 12.247956
For the 53 th path, the integral of departure rate of final hk is 0.000000
For the 54 th path, the integral of departure rate of final hk is 0.000000
For the 55 th path, the integral of departure rate of final hk is 20.007114
For the 56 th path, the integral of departure rate of final hk is 0.000000
For the 57 th path, the integral of departure rate of final hk is 0.000000
For the 58 th path, the integral of departure rate of final hk is 20.008997
```

Sioux Falls Network

Considering 23 OD pairs and 232 paths

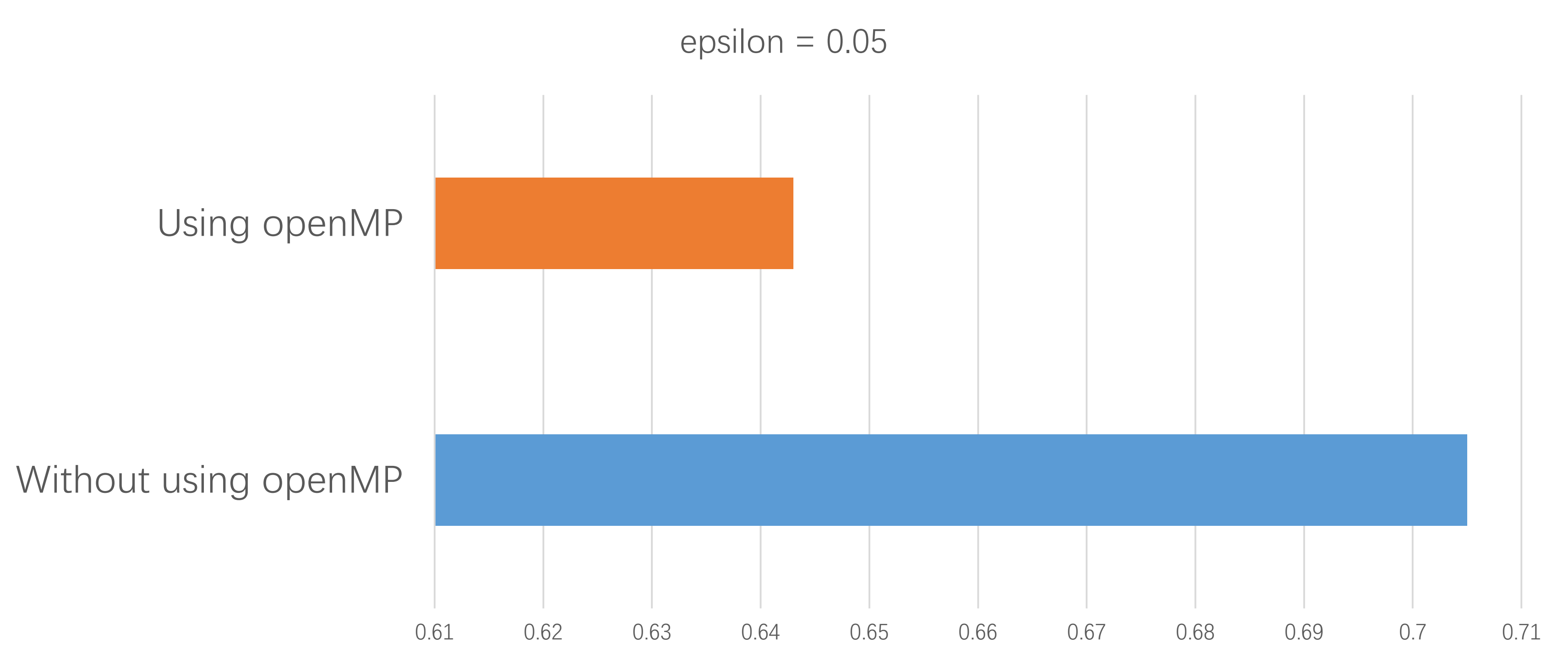
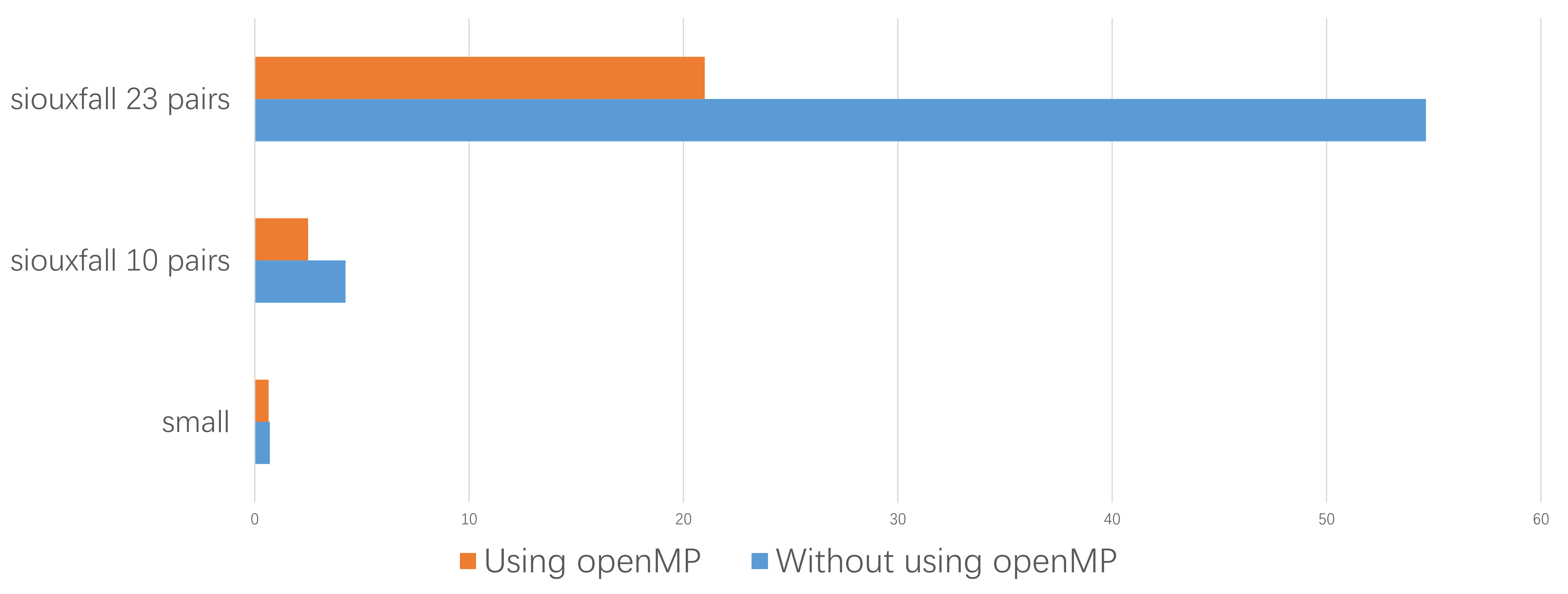


Table of average time for a single iteration



With a larger graph, comes with a higher efficiency of openMP.

THE UNIVERSITY OF TENNESSEE OAK RIDGE National Laboratory NSF JICS Joint Institute for Computational Sciences

Sioux Falls Network (Larger graph)

Considering 552 OD pair and 6255 paths

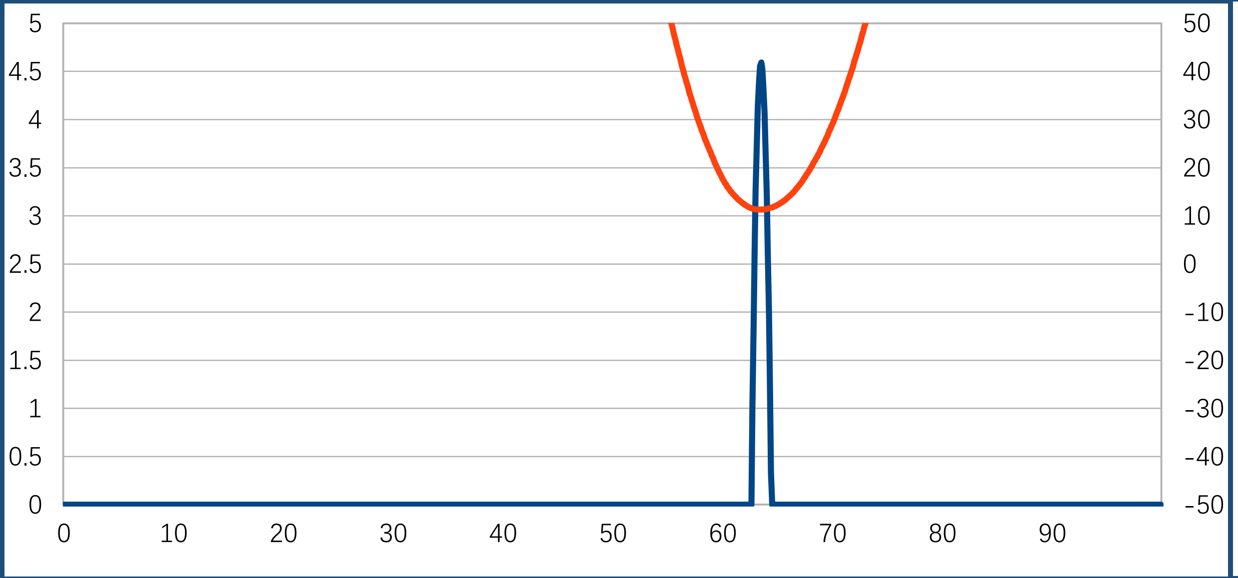
- Iterations:12
- Running time: 10968.9 s
- Average time for a iteration: 914.1 s(15.2 min)
- Epsilon: 0.5

```
For the 6208 th path, the integral of departure rate of final hk is 5.628382
For the 6209 th path, the integral of departure rate of final hk is 0.000000
For the 6210 th path, the integral of departure rate of final hk is 0.000000
For the 6211 th path, the integral of departure rate of final hk is 0.000000
For the 6212 th path, the integral of departure rate of final hk is 0.000000
For the 6213 th path, the integral of departure rate of final hk is 0.000000
For the 6214 th path, the integral of departure rate of final hk is 0.000000
For the 6215 th path, the integral of departure rate of final hk is 0.000000
For the 6216 th path, the integral of departure rate of final hk is 4.373205
For the 6217 th path, the integral of departure rate of final hk is 0.000000
```

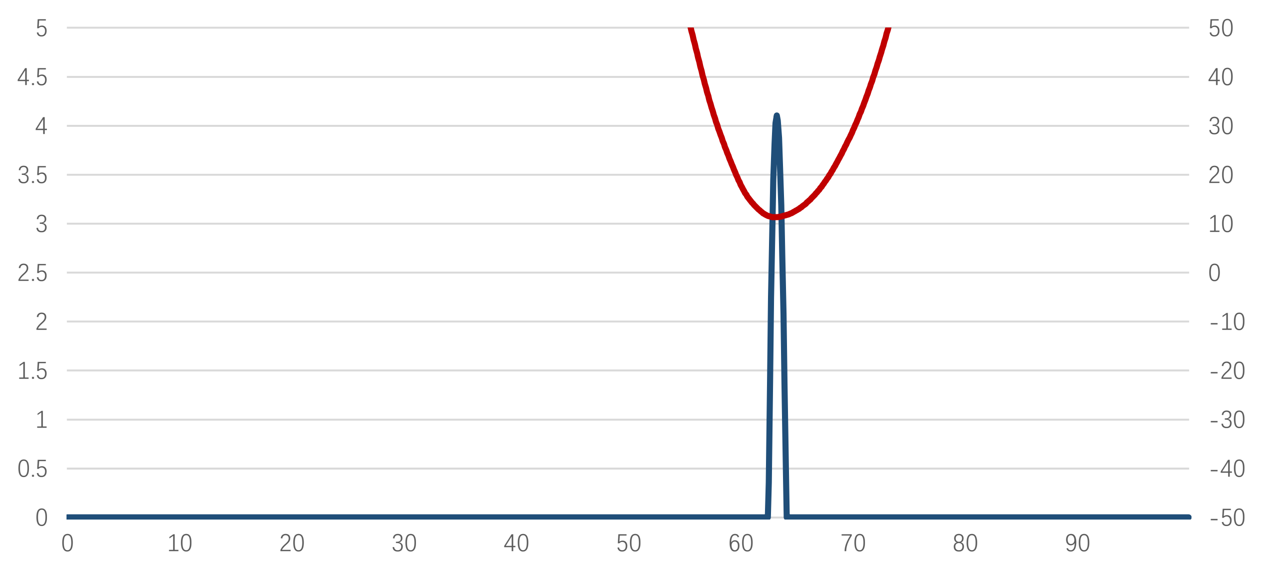
Sioux Falls Network (Larger graph)

Considering 552 OD pair and 6255 paths

Graph of path 6208



Graph of path 6216



Future work

Implementation on CUDA or GPU.
Improvement on DNL based on DTA

THE UNIVERSITY OF TENNESSEE · OAK RIDGE National Laboratory · NSF · JICS Joint Institute for Computational Sciences

Dynamic network loading by PDEs

LWR model --- hydrodynamic model

Non-linear PDE

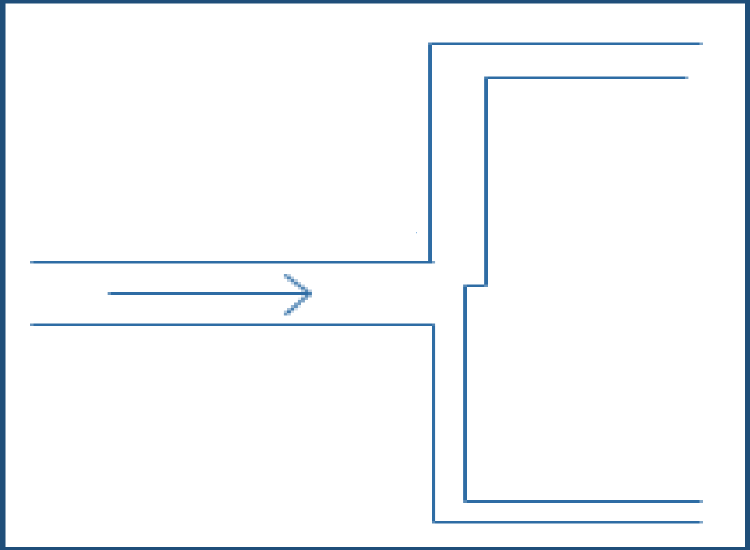
$$\frac{\partial \rho(t,x)}{\partial t} + \frac{\partial f(\rho(t,x))}{\partial x} = 0$$

Advantages

1. Queues and delay
2. Density-speed relationship
3. First-in-First-out principle
4. Route information

It depends on the downstream and upstream flow state.

Origin



Destination

Water propagate in pipe

Thickness → capacity

Water → flow

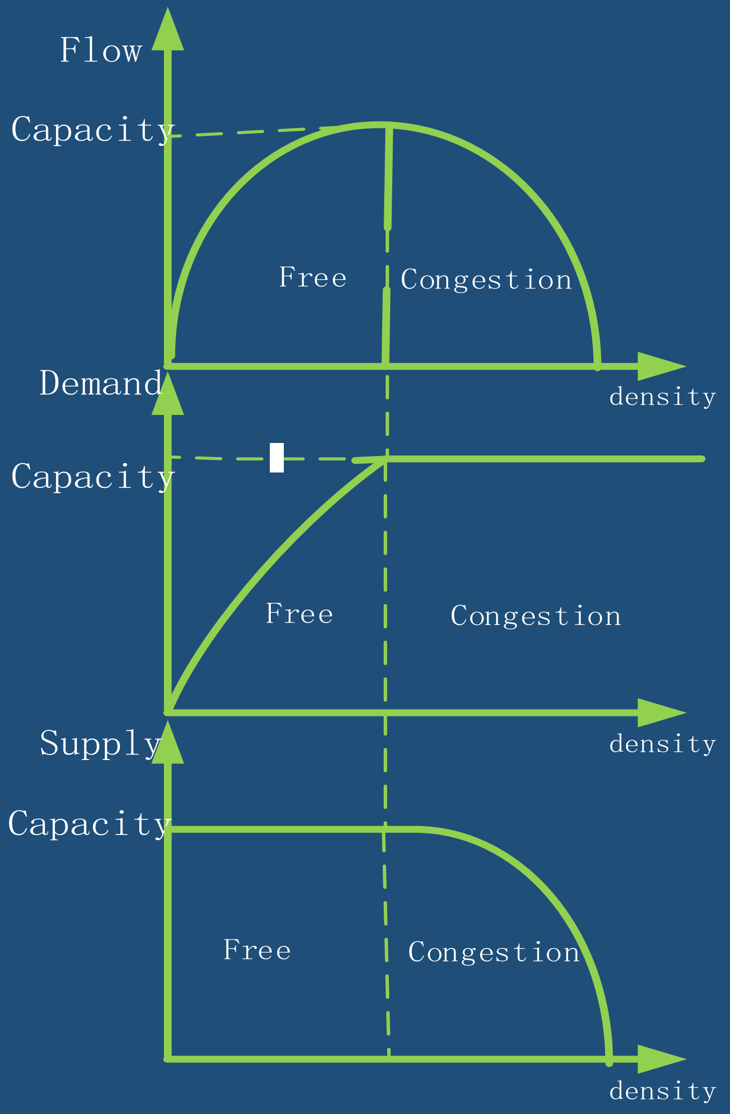
Implementation

Discretization (Han,2012)

$$Da(t) = \begin{cases} q_{in}^{a}(t - \frac{L_a}{k_a}) & if\ N_{up}^{a}(t - \frac{L_a}{k_a}) = N_{down}^{a}(t) \\ C_a & if\ N_{up}^{a}(t - \frac{L_a}{k_a}) > N_{down}^{a}(t) \end{cases}$$

Link model

$$S_a(t) = \begin{cases} q_{out}^{a}(t - \frac{L_a}{w_a}) & if\ N_{up}^{a}(t) = N_{down}^{a}(t - \frac{L_a}{w_a}) + \rho_{jam}^{a} L_a \\ C_a & if\ N_{up}^{a}(t) < N_{down}^{a}(t - \frac{L_a}{w_a}) + \rho_{jam}^{a} L_a \end{cases}$$



Implementation

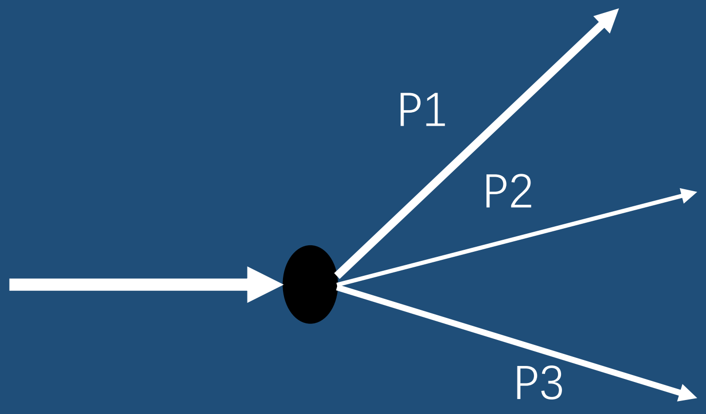
Discretization (Han,2012)

Junction model

$$\alpha_{ij}^{J}(t) = \sum_{p \ni a,b} \mu_a^p(t, L_a)$$

$$q_{out,i} = \min\{D_i(t), \frac{S_j(t)}{\alpha_i}\} j \in I^o$$

$$q_{in,j} = \sum_{i \in I^v} \alpha_{ij} \cdot q_{out,i}(t)$$



Path delay

$$N_{down}^a(t) = N_{up}^a(\tau_a(t))$$

$$travel\_time = \tau_a(t) - t$$

Moskowitz function

Algorithm 2: Computing dynamic network loading based on LWR model by C

Initialization: path,timespan, arc, h0, Q(demand), epsilon (tolerance) et.

for all i = 1 : num (OD pair)

  While condition is true ⅄ hk - hk+1 is larger than tolerance

  for t =1 : num (timesteps)

    for i = 1 : num (links)

      Solve D Get link demand ⅄equation 5.2

      Solve S Get link supply ⅄equation 5.3

        for j = 1 : num (linkin)

          for k = 1 : num (linkout)

        get turning ratio ⅄equation 5.4

        end for

      end for

    end for

    Calculate entering flow ⅄equation 5.5

    Calculate exiting flow ⅄equation 5.6

  end for

  get effective path delay

  get a v in each iteration ⅄equation 3.3

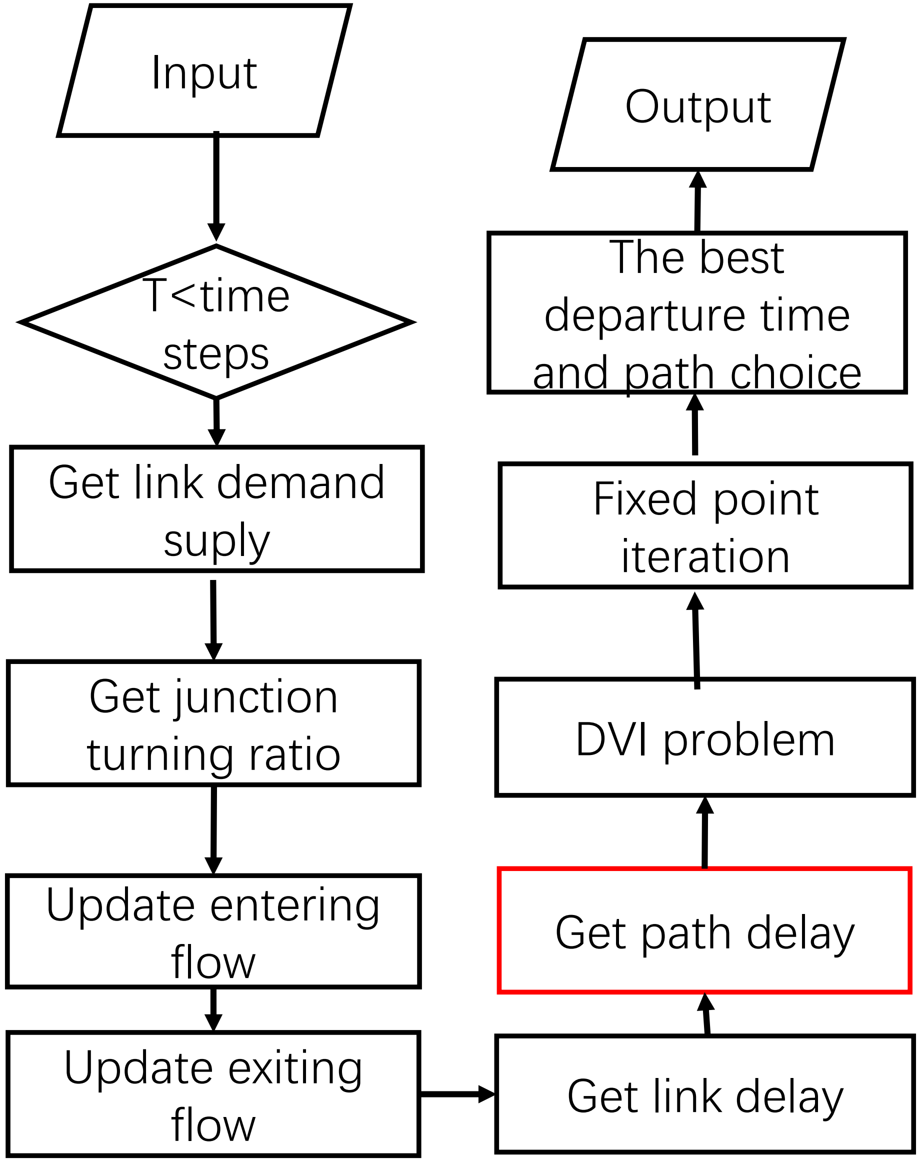
  update hk+1 according to equation 3.4 ⅄each v map to a hk

 end while

end for

  Output: Phi, hk

Flow chart



Discretization

The value in each cellular denotes the departure rate、downstream flow、upstream flow associate with different tables.

Time Steps

Path

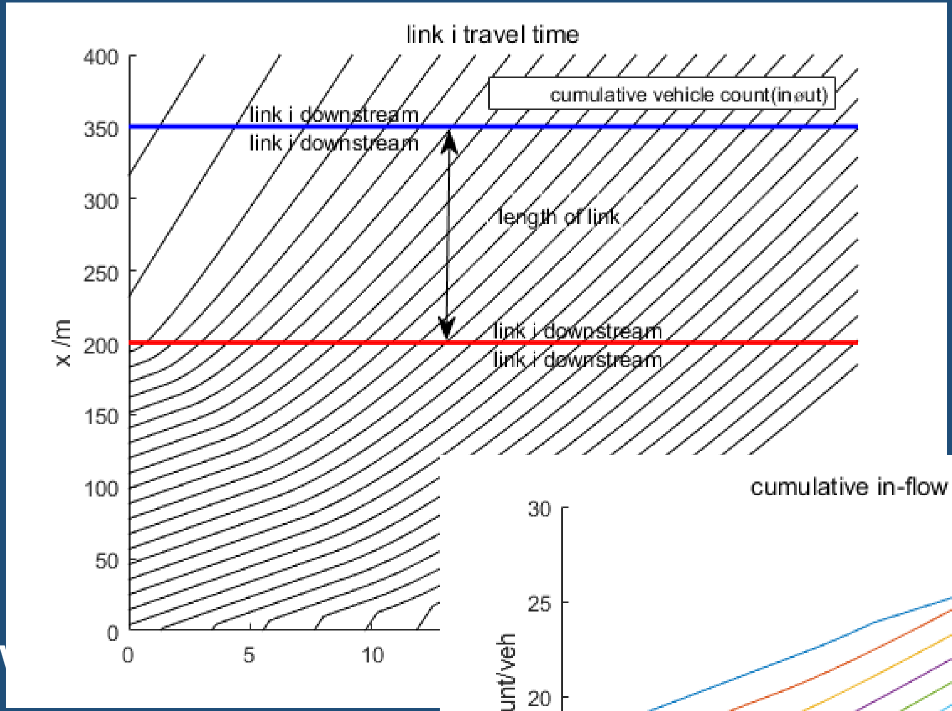
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	0	0	0	0	0.2	0.3	0.4	0.5	0.6	0.7	0.5	0.2	0	0	0	0	0	0	0	0
2	0	0	0	0	0.1	0.4	0.6	0.8	0.9	0.7	0.6	0.3	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0.6	0.2	0.3	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0.4	0.6	0.8	0.9	0.5	0.3	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0.6	0.7	0.7	0.6	0.5	0.2	0.4	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0.3	0.4	0.5	0.6	0.4	0.3	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0.2	0	0.4	0.5	0.6	0.5	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0.6	0.6	0.7	0.5	0.4	0	0	0	0	0	0	0	0

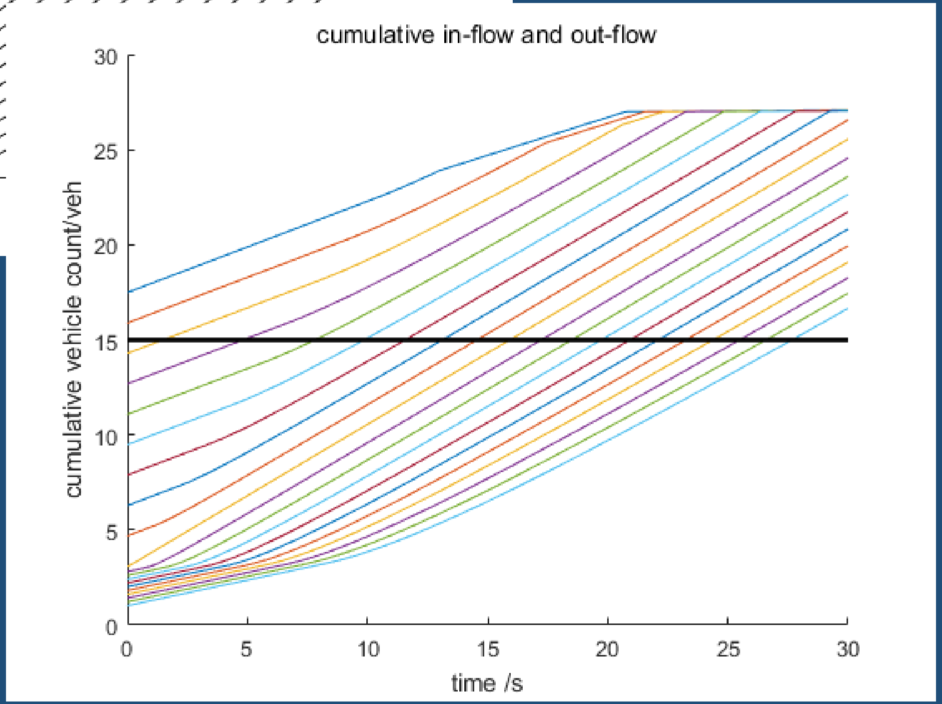
Link travel time example

Vertical difference →length of link.

Slant line → cumulative inflow and outflo

Horizontal difference → link travel time





Conclusion and remarks

For ODE model,
- We apply *openMP* in solving *ODE* and fix-point iteration
- succeed and obtain ideal assignment results.
- The parallel computing significantly fastened the solving speed as the same result compared with common computation
- In this way, it's possible to apply this technology to urban network planning.

For PDE model,
- It remains many works to do. We haven't finish the entire code
- we proposed the pseudo code, and the next step is realizing it by C.
- If possible, parallel computing will be also used to certify the high performance in dynamic traffic assignment.

Acknowledgements

I would like to express my deep gratitude to my mentor Dr. Liu and Dr. Wong, For their patient guidance and enthusiastic encouragement of this project. I would also like to thanks my partner Geyu and any other who helped me.

In addition, this project was sponsored by the National Science Foundation through Research Experience for Undergraduates (REU) award, with additional support from the Joint Institute of Computational Sciences at University of Tennessee Knoxville. This project used allocations from the Extreme Science and Engineering Discovery Environment (XSEDE), which is supported by the National Science Foundation. In addition, the computing work was also performed on technical workstations donated by the BP High Performance Computing Team.

References

Daganzo, C., 1994. The cell transmission model. Part I: a simple dynamic representation of highway traffic. Transportation Research Part B 28 (4), 269–287.
Friesz, T., Han, K., Neto, P., Meimand, A. and Yao, T. (2013). Dynamic user equilibrium based on a hydrodynamic model. Transportation Research Part B: Methodological, 47, pp.102-126.
Friesz, T., Kim, T., Kwon, C. and Rigdon, M. (2011). Approximate network loading and dual-time-scale dynamic user equilibrium. Transportation Research Part B: Methodological, 45(1), pp.176-207.
Friesz, T. (2014). Dynamic optimization and differential games. [Place of publication not identified]: Springer-Verlag New York.
Han, K., Piccoli, B., Friesz, T. L., & Yao, T. 2012. A Continuous-time Link-based Kinematic Wave Model for Dynamic Traffic Networks. Arxiv e-prints, Aug.
Han, K., Friesz, T.L., Yao, T., 2013a. A partial differential equation formulation of Vickrey's bottleneck model, part I: Methodology and theoretical analysis. Transportation Research Part B 49, 55–74.
Han, K., Friesz, T.L., Yao, T., 2013b. A partial differential equation formulation of Vickrey's bottleneck model, part II: Numerical analysis and computation. Transportation Research Part B 49, 75–93.
Han, K., Piccoli, B. and Szeto, W. (2015). Continuous-time link-based kinematic wave model: formulation, solution existence, and well-posedness. Transportmetrica B: Transport Dynamics, 4(3), pp.187-222.
Han, K., Piccoli, B. and Friesz, T. (2016). Continuity of the path delay operator for dynamic network loading with spillback. Transportation Research Part B: Methodological, 92, pp.211-233.
Han, K., Friesz, T.L., Yao, T., 2014. Vehicle spillback on dynamic traffic networks and what it means for dynamic traffic assignment models. 5th International Symposium on Dynamic Traffic Assignment. Salerno, Italy, 17-19 June 2014.
Lighthill, M. and Whitham, G. (1955). On kinematic waves. [London]: [Royal Society].
Programming Methods

End

Thank You