ME308 PROJECT – TEAM 36

Bus Route Optimization

<u>Members</u>

Sumeet Ranjan : 190040120
Aditya Chopra : 190020034
Rezeena Chinthamalla : 190100099
Thurkapally Manogna : 190100130

CONTENTS

MODEL

- As the first approach we try to model the bus routing problem for the campus, to transport the students from their respective residence to lecture hall.

- This model incorporates bus capacity and time constrained vehicle problem.

- The model optimises for the least operating cost for both operating cost and amortized servicing cost of the busses, giving us the least number of bus required to cover all the bus stops/

Parameters, Variables and Objective Function

```
param N;                              # Number of Intermediate Nodes + 1;
set N1 := {0..N};                     # 0 U {I} U d
set N2 := {1..N-1} ;                  #  I
set N3 := {0..N-1};                   # 0 U {I}
set N4 := {1..N};                     # {I} U d
set S := 1 .. N;
set SS := 1 .. 2**N-1;
set POW {k in SS} := {i in S: (k div 2**(i-1)) mod 2 = 1};


# Part1: Decision Variables
var x{N1,N1} binary;                  #Select the path_{ij}
var k integer;                        #Number of buses
var u{N2};
var v{N2};
var c{N1,N1};                         #Cost_{i,j}



param d{N1,N1};                       #Distance_{i,j}
param f;                              #Fixed Cost
param Q;
param T;
param a;

param b{N2};                          # q_i


# Part2: Objective Function
minimize z: f*k + sum{ i in N1 , j in N1 }c[i,j]*x[i,j];
```

VARIABLES

- $x_{ij} = \begin{cases} 1 & if\ any\ bus\ traverses\ through\ i, j \\ 0 & otherwise \end{cases}$
- $k = number\ of buses$
- $u_i = Total\ number\ of\ students\ picked\ just\ after\ leaving\ node\ i.$
- $v_i = distance\ travelled\ by\ bus\ uptil\ node\ i.$

PARAMETERS

- $N = Number\ of\ intermediate\ nodes + 1.$
- *{0} : origin ; {d}: dummy node.*
- *I = set of intermediate nodes.*
- $Sets : N_1 = I \cup \{0\} \cup \{d\};\ N_2 = I\ ;\ N_3 = \{0\} \cup I\ ;\ N_4 = I \cup \{d\}$
- $d_{ij} = distance\ between\ nodes\ i\ and\ j$
- $f = amortized\ cost\ for\ each\ bus$
- $Q = Upper\ bound\ on\ total\ capacity\ of\ each\ bus.$
- $T = Upper\ bound\ on\ distance\ travlled\ by\ each\ bus.$
- $b_i = number\ of\ students\ to\ be\ picked\ up\ at\ each\ node\ i$

OBJECTIVE FUNCTION

$$minimise: \quad f * k + \sum_{i \in N_1} \sum_{j \in N_1} c_{ij} * x_{ij}$$

The objective function is the total cost of travel and fixed amortized cost for k buses used. The aim of this formulation is to minimise the objective function.

Constraints

```
# Part 3: Constraints
s.t. M0 {i in N1, j in N1}: c[i,j] =a*d[i,j];

s.t. M1: sum{i in N2}x[0,i] = k;
s.t. M2: sum{i in N2}x[i,N] = k;

s.t. M3 { i in N2}: sum{ j in N4 }x[i,j] = 1;
s.t. M4 { j in N2 }: sum{ i in N3 }x[i,j] = 1;


## alternate capacity constraint ##
#s.t. MA1 {i in N2 , j in N2: i<>j}: u[i]-u[j]+Q*x[i,j]+(Q-b[i]-b[j])*x[j,i] <= Q- b[j];

###### capacity constraints ######

s.t. M5 {i in N2}: b[i]<=u[i];
s.t. M6 {i in N2, j in N2: i<>j}: u[i]+b[j]<=1000000*(1-x[i,j])+u[j];
s.t. M7 {i in N2, j in N2: i<>j}: u[i]+b[j]+1000000*(1-x[i,j])>=u[j];

s.t. M8 {i in N2}: u[i]<=Q;
s.t. M9 {i in N2}: u[i]-b[i]*x[0,i]+Q*x[0,i] <= Q;

###### travelling distance cap ######

s.t. M10 {i in N2, j in N2: i<>j}: v[i]+d[i,j]<=1000000*(1-x[i,j])+v[j];
s.t. M11 {i in N2, j in N2: i<>j}: v[i]+d[i,j]+1000000*(1-x[i,j])>=v[j];

s.t. M12 {i in N2}: v[i] - d[0,i]*x[0,i] >= 0;
s.t. M13 {i in N2}: v[i]-d[0,i]*x[0,i]+T*x[0,i]<=T;
#s.t. MA2 {i in N2}: v[i]<=T;
#s.t. MA3 {i in N2}: x[i,i] = 0;

## alternate distance cap constraint ##
#s.t. MA4 {i in N2 , j in N2 : i!=j}: v[i] - v[j] + (T - d[i,N] - d[0,j] + d[i,j])*x[i,j]+ (T-d[i,N]-d[0,j]-d[j,i])*x[j,i] <= T-d[i,N]-d[0,j];
#s.t. MA5 {i in N2}: v[i] - d[0,i]*x[0,i] >= 0;
#s.t. MA6 {i in N2}: v[i]-d[0,i]*x[0,i]+T*x[0,i]<=T;

###### subtour elimination Dantzig-Fulkerson-Johnson formulation #####
s.t. M14 {s in SS: card(POW[s])>=2}: sum{i in POW[s], j in POW[s]: i<>j} x[i,j]<=card(POW[s])-1;
```

CONSTRAINTS

$$M0: \quad c_{ij} = a * d_{ij} \quad \forall\, i,j \in N_1$$

Setting the cost matrix proportional to distance matrix.

$$M1: \sum_{i \in N_2} x_{0i} = k$$

Sum of buses leaving the origin (o) is equal to number of buses.

$$M2: \sum_{i \in N_2} x_{id} = k$$

Sum of buses entering the dummy node (d) is equal to number of buses.

$$M3: \sum_{j \in N_4} x_{ij} = 1\,; \quad \forall i \in N_2$$

Sum of buses leaving an intermediate node = 1.

$$M4: \sum_{i \in N_3} x_{ij} = 1\,; \quad \forall j \in N_2$$

Sum of buses entering an intermediate node = 1.

$$M5: \quad b_i \leq u_i \quad ; \ \forall i \in N_2$$

Total number of students picked up just after leaving node i is greater than or equal to number of students picked upt at node i.

$$M6: \ u_i + b_j \leq 1000000 * (1 - x_{ij}) + u_j \quad ; \forall i, j \in N_2 \ , i \neq j$$

$$M7: u_i + b_j + 1000000 * (1 - x_{ij}) \geq \ u_j \quad ; \forall i, j \in N_2, \ i \neq j$$

Trivial for $x_{ij} = 0$. *For* $x_{ij} = 1$, *M6 and M7 combine to give* $u_i + b_j = u_j$

i.e. total number of students picked up till node i + number of students at node j = total number of students picked up till node j , as bus moves from i to j

$$M8: u_i \leq Q \ ; \ \forall \ i \in N_2$$

Total number of students picked up just after leaving node i is less than or equal to maximum capacity possible

$$M9:\ u_i - b_i * x_{0i} + Q * x_{oi} \leq Q \quad \forall\, i \in N_2$$

$Trivial\ for\ nodes\ for\ which\ x_{0i} = 0.\ For\ x_{0i} = 1;\ M9\ and\ M5\ combine\ to\ give\ b_i = u_i.$

$$M10: v_i + d_{ij} \leq 1000000 * \left(1 - x_{ij}\right) + v_j \quad ; \forall i, j \in N_2\ , i \neq j$$

$$M11: v_i + d_{ij} + 1000000 * \left(1 - x_{ij}\right) \geq v_j \quad ; \forall i, j \in N_2\ , i \neq j$$

$Trivial\ for\ x_{ij} = 0\ .For\ x_{ij} = 1\ , M10\ and\ M11\ combine\ to\ give\ v_i + d_{ij} = v_j.$
$i.e. distance\ travelled\ by\ bus\ uptil\ node\ i + distance\ between\ nodes\ i\ and\ j =$
$distance\ travelled\ by\ bus\ uptil\ node\ j\ , as\ bus\ moves\ from\ i\ to\ j.$

$$M12: v_i - d_{oi} * x_{oi} \geq 0;\ \forall i \in N_2$$

Trivial for $x_{0i} = 0$. *For* $x_{oi} = 1$; *M12 and M13 combine to give* $v_i = d_{oi}$.

$$M13: v_i - d_{oi} * x_{oi} + T * x_{oi} \leq T\ ;\ \forall i \in N_2$$

For $x_{oi} = 0$; v_i *is limited by maximum distance that the bus can travel. For* $x_{0i} = 1$,
M12 and M13 combine to give $v_i = d_{oi}$

$$M14: \sum_i \sum_j x_{ij} \leq |POW(s)| - 1\ ;\forall\ s \in \{1,2,..(2^n - 1)\}, i, j \in POW(s), i \neq j, |POW(s)| \geq 2$$

To ensure that cycles do not occur.

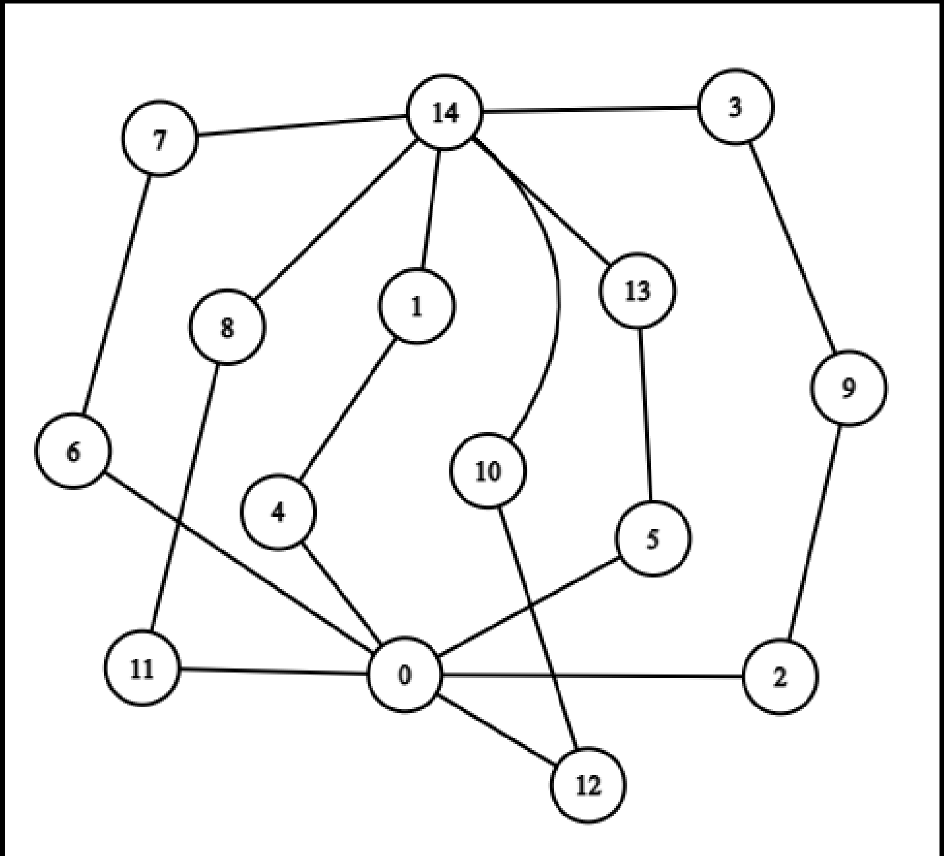
DATA SET

```
param N = 14;
param a = 1;
param f = 20000;
param Q = 30;
param T = 10000;
param d :
    0   1   2   3   4   5   6   7   8   9   10 11 12 13 14 :=
0   50000   401 102 245 34  22  15  27  338 12  397 78  37  95  50000
1   401 50000   501 179 369 411 414 374 85  391 5   363 426 494 0
2   102 501 50000   346 135 100 92  127 439 112 497 176 104 10  0
3   245 179 346 50000   214 256 259 219 95  236 175 205 271 339 0
4   34  369 135 214 50000   45  48  10  303 25  365 57  59  128 0
5   22  411 100 256 45  50000   13  37  348 23  406 98  52  93  0
6   15  414 92  259 48  13  50000   40  352 26  410 102 39  85  0
7   27  374 127 219 10  37  40  50000   309 17  370 62  54  120 0
8   338 85  439 95  303 348 352 309 50000   329 78  298 364 432 0
9   12  391 112 236 25  23  26  17  329 50000   387 78  47  106 0
10  397 5   497 175 365 406 410 370 78  387 50000   359 422 490 0
11  78  363 176 205 57  98  102 62  298 78  359 50000   75  170 0
12  37  426 104 271 59  52  39  54  364 47  422 75  50000   99  0
13  95  494 10  339 128 93  85  120 432 106 490 170 99  50000   0
14  50000   0   0   0   0   0   0   0   0   0   0   0   0   0   50000;



param b :=
1  5
2  6
3  9
4  25
5  11
6  13
7  17
8  14
9  15
10 11
11 16
12 18
13 19;
```

RESULTS




```
7764 branch-and-bound nodes
absmipgap = 11.7178, relmipgap = 9.61591e-05
k = 6

x[i,j] [*,*]
:    0   1   2   3   4   5   6   7   8   9  10  11  12  13  14    :=
0    0   0   1   0   1   1   1   0   0   0   0   1   1   0   0
1    0   0   0   0   0   0   0   0   0   0   0   0   0   0   1
2    0   0   0   0   0   0   0   0   0   1   0   0   0   0   0
3    0   0   0   0   0   0   0   0   0   0   0   0   0   0   1
4    0   1   0   0   0   0   0   0   0   0   0   0   0   0   0
5    0   0   0   0   0   0   0   0   0   0   0   0   0   1   0
6    0   0   0   0   0   0   0   1   0   0   0   0   0   0   0
7    0   0   0   0   0   0   0   0   0   0   0   0   0   0   1
8    0   0   0   0   0   0   0   0   0   0   0   0   0   0   1
9    0   0   0   1   0   0   0   0   0   0   0   0   0   0   0
10   0   0   0   0   0   0   0   0   0   0   0   0   0   0   1
11   0   0   0   0   0   0   0   0   1   0   0   0   0   0   0
12   0   0   0   0   0   0   0   0   0   0   1   0   0   0   0
13   0   0   0   0   0   0   0   0   0   0   0   0   0   0   1
14   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
;

:  u[i]  v[i]    :=
1    30   403
2     6   102
3    30   450
4    25    34
5    11    22
6    13    15
7    30    55
8    30   376
9    21   214
10   29   459
11   16    78
12   18    37
13   30   115
;

z = 121858
```

MODEL 2

- This model deals with the case of variable initial and final stations.
- For each of the {R} routes, the initial and final stations are fixed.
- Number of Passengers demanding to travel from node o to destination d are denoted as D^{od}.
- Demand D^{od} can make transfer at any node i, from one route to the other.
- For such system, we need to specify the distance between nodes, initial and final stations for each route and demand od.

Parameters, Variables and Objective Function

```
#Part1 Parameters
param r0;
set R = {1..r0};

param n;
set N = {1..n};
set S := 1 .. n;
set SS := 1 .. 2**n;
set POW {k in SS} := {i in S: (k div 2**(i-1)) mod 2 = 1};

param cap;
param D{N,N};
param e{R};
param f{R};
param l{N,N};                    #C_bus
param trmax;
param M;
param Ca{N,N};

# Part2: Decision Variables
var x{N,N,R} binary;            #[I][J][R]
var p{N,N,N,N,R} binary;        #[O][D][I][J][R]
var t{N,N,N} binary;            #[O][D][I]
var m{N,N} binary;
var Cb{N,N};
var u{N,N,R};

# Part3: Objective Function
maximize z: sum {o in N, d in N} (D[o,d]*m[o,d]) -sum{o in N, d in N}Cb[o,d] -sum{i in N, j in N, r in R} (l[i,j]*x[i,j,r])-  0.001*sum{o in N, d in N, i in N} (m[o,d]*D[o,d]*t[o,d,i]);
```

VARIABLES

- $x_{ijr} = \begin{cases} 1 & if\ route\ r\ passes\ through\ segment\ ij \\ 0 & otherwise \end{cases}$
- $p_{ijr}^{od} = \begin{cases} 1 & if\ demand\ \ od\ uses\ segment\ ij\ on\ route\ r \\ 0 & otherwise \end{cases}$
- $t_i^{od} = \begin{cases} 1 & if\ demand\ od\ stops\ at\ node\ i\ \ for\ transfer \\ 0 & otherwise \end{cases}$
- $m^{od} = \begin{cases} 1 & if\ demand\ od\ uses\ bus\ for\ travelling \\ 0 & otherwise \end{cases}$
- $C_b^{od} = Cost\ for\ demand\ od, for\ using\ bus\ system.$
- $u_{ijr} = Number\ of\ passengers\ travelling\ from\ node\ i\ to\ node\ j\ , using\ route\ r.$

PARAMETERS

- r_0 : $Number\ of\ routes$
- n : $Number\ of\ nodes$
- D^{od}: $Demand\ for\ travelling\ from\ node\ o\ to\ d.$
- $e(r)$: $initial\ station\ for\ route\ r.$
- $f(r)$: $final\ station\ for\ route\ r.$
- l_{ij}: $length\ between\ node\ i\ and\ j.$
- tr_{max}: $maximum\ number\ of\ transfers\ desired.$
- M: $constant\ with\ high\ value$
- cap: $maximum\ capacity\ the\ bus\ can\ fill.$

SETS

- $R = \{1,2,3 \ldots r_0\}$
- $N = \{1,2,3 \ldots n\}$
- $S = \{1,2, \ldots n\}$
- $SS = \{1,2, \ldots 2^n\}$
- $POW = power\ set\ of\ S$

OBJECTIVE FUNCTION

$$maximize : \sum_{o}\sum_{d} D^{od}m^{od} - \sum_{o}\sum_{d} C_b^{od} - \sum_{i}\sum_{j}\sum_{r} l_{ij} * x_{ijr} - 0.001\sum_{o}\sum_{d}\sum_{i} m^{od} * D^{od} * t_i^{od}$$

The objective function is combination of smaller objective functions : $z = w_1 * z_1 - w_2 * z_2 - w_3 * z_3 - w_4 * z_4$;
$Here\ w_i\ is\ the\ weight\ of\ i^{th}(smaller)\ objective\ function$

z_1 : $maximizing\ demand\ served\ by\ bus\ system.$
z_2 : $minimizing\ cost\ of\ bus\ system\ for\ demand\ od.$
z_3 : $minimizing\ total\ bus\ route\ length.$
z_4 : $minimizing\ the\ total\ number\ of\ transfers\ made\ by\ all\ users.$

Constraints

```
# Part 4: Constraints

s.t. M01 {r in R}: sum{j in N} x[e[r],j,r]=1;
s.t. M02 {r in R}: sum{j in N} x[j,f[r],r]=1;
s.t. M03 {i in N, r in R: i!=e[r] and i!=f[r]}: sum{j in N}x[j,i,r]=sum{k in N}x[i,k,r];

s.t. M04{o in N, d in N}: Cb[o,d]=sum{i in N, j in N, r in R}l[i,j]*p[o,d,i,j,r];

#s.t. MA1 {i in N, r in R}: x[i,i,r] = 0;
s.t. M05 {s in SS, r in R:card(POW[s])>=2 }:sum{i in POW[s], j in POW[s]: i<>j}x[i,j,r]<=card(POW[s])-1;

#s.t. MA2 {r in R, i in N , j in N : i<j}: x[j,i,r] = x[i,j,r];

s.t. M06 {o in N, d in N}: sum{j in N, r in R}p[o,d,o,j,r] = 1;
s.t. M07 {o in N, d in N}: sum{j in N, r in R}p[o,d,j,d,r] = 1;

#s.t. MA3 {o in N, d in N, i in N , j in N, r in R}: p[o,d,i,j,r]*p[o,d,j,i,r]<=0;

s.t. M08 {o in N, d in N, i in N: i!=o and i!=d}: sum{j in N, r in R:i<>j}p[o,d,j,i,r] = sum{k in N, r in R:i<>k}p[o,d,i,k,r] ;

#s.t. MA4 {o in N, d in N}: Cb[o,d]<=Ca[o,d]+M*(1-m[o,d]);

s.t. M09 {o in N, d in N, r in R, i in N: i!=o and i!=d}: sum{j in N}p[o,d,j,i,r] - sum{k in N}p[o,d,i,k,r]<=t[o,d,i];
#s.t. MA5 {o in N, d in N}: sum{i in N} t[o,d,i] <= trmax;
s.t. M10 {r in R , i in N, j in N}: sum{o in N, d in N} (p[o,d,i,j,r]+p[o,d,j,i,r]) <= M*(x[i,j,r]+x[j,i,r]);

s.t. M11 {i in N,j in N, r in R}: (sum{o in N,d in N}D[o,d]*p[o,d,i,j,r]*m[o,d])<=cap;
s.t. M12 {i in N,j in N, r in R}: u[i,j,r] <= (sum{o in N,d in N}D[o,d]*p[o,d,i,j,r]*m[o,d]);
s.t. M13 {i in N,j in N, r in R}: u[i,j,r] >= (sum{o in N,d in N}D[o,d]*p[o,d,i,j,r]*m[o,d]);
#s.t. MA6 {i in N, r in R}:u[i,r]<=cap;
```

CONSTRAINTS

$M1 : \sum_j x_{e(r)jr} = 1 \; ; \forall \, r$

For each route 'r', number of buses leaving e(r) is 1.

$$M2 : \sum_j x_{jf(r)r} = 1 \; ; \forall \, r$$

For each route 'r', number of buses entering f(r) is 1.

$$M3 : \sum_j x_{jir} = \sum_k x_{ikr} \quad ; \forall r \, , i \neq e(r) \neq f(r)$$

For each node i (other than e(r) and f(r)), : a segment of route r arrives at node i, then a segment must also leave from it.

$$M4: C_b^{od} = \sum_i \sum_j \sum_r l_{ij} * p_{ijr}^{od} \quad ; \forall\ o, d$$

Cost of bus for demand od is proportinal to total length travelled by the bus, via all routes possible.

$$M5: \sum_{i \in POW(s)} \sum_{j \in POW(s)} x_{ijr} \leq |POW(s)| - 1 \ ; \ \forall r, s \in SS \ ; |POW(s)| \geq 2$$

To ensure that no cycles will occur, we use this constraint.

$$M6: \sum_j \sum_r p_{ojr}^{od} = 1 \ ; \ \forall\ o, d$$

From the trip origin o of each demand od, the demand should move from o to j via 1 route.

$$M7:\sum_{j}\sum_{r}p_{jdr}^{od}=1\,;\ \ \forall\, o,d$$

Towards the final destination trip d of each demand od, the demand enters d via 1 route.

$$M8:\sum_{\substack{j \\ i\neq j}}\sum_{r}p_{jir}^{od}=\sum_{\substack{k \\ i\neq k}}\sum_{r}p_{ikr}^{od}\ \ \forall o,d,i\neq o\neq d$$

In nodes i, different from the trip origin or destination of each demand od, if a traveler arrives at that node, he must continue his journey to another node.

$$M9:\sum_{j}p_{jir}^{od}-\sum_{k}p_{ikr}^{od}\leq t_{i}^{od}\ \ \ \forall o,d,r,i\neq o\neq d$$

Constraint to track transfer at each node i, for demand od.

$$M10: \sum_{o} \sum_{d} p_{ijr}^{od} + p_{jir}^{od} \leq M * \left(x_{ijr} + x_{jir}\right); \ \forall r, i, j$$

Trivial for either or both x_{ijr} or $x_{jir} = 1$. For $x_{ijr} = x_{jir} = 0$, for every demand od, $p_{ijr}^{od} = p_{jir}^{od} = 0$

$$M11: u_{ijr} \leq cap; \ \forall r, i, j$$

The number of passengers in the bus is bounded by the maximum capacity of bus.

$$M12: u_{ijr} = \sum_{o} \sum_{d} D^{od} * p_{ijr}^{od} * m^{od} \ ; \ \forall r, i, j$$

Number of passengers travelling from node i to node j, using route r.

Data set

```
param r0 = 3;
param n = 5;
param cap = 50;

param D:
       1     2     3    4    5     :=
 1    0    50    20   11   24
 2    17    0     3    21    27
 3    1     8     0    31    32
 4    6     8     3     0    9
 5    15   18    19     5    0;

 param e:=
 1  1
 2  2
 3  3;

 param f:=
 1 3
 2 4
 3 5;

 param l:
     1   2     3    4     5      :=
 1   99999   1.6  3.8   5.6  4.9
 2  1.6  99999    99   9.1  6.3
 3  3.8  4.8    99999   2.1  9.0
 4  5.6  9.1  2.1    99999   12.3
 5  4.9  6.3  9.0   12.3  99999;

 param trmax = 10;
 param M = 1000;
```

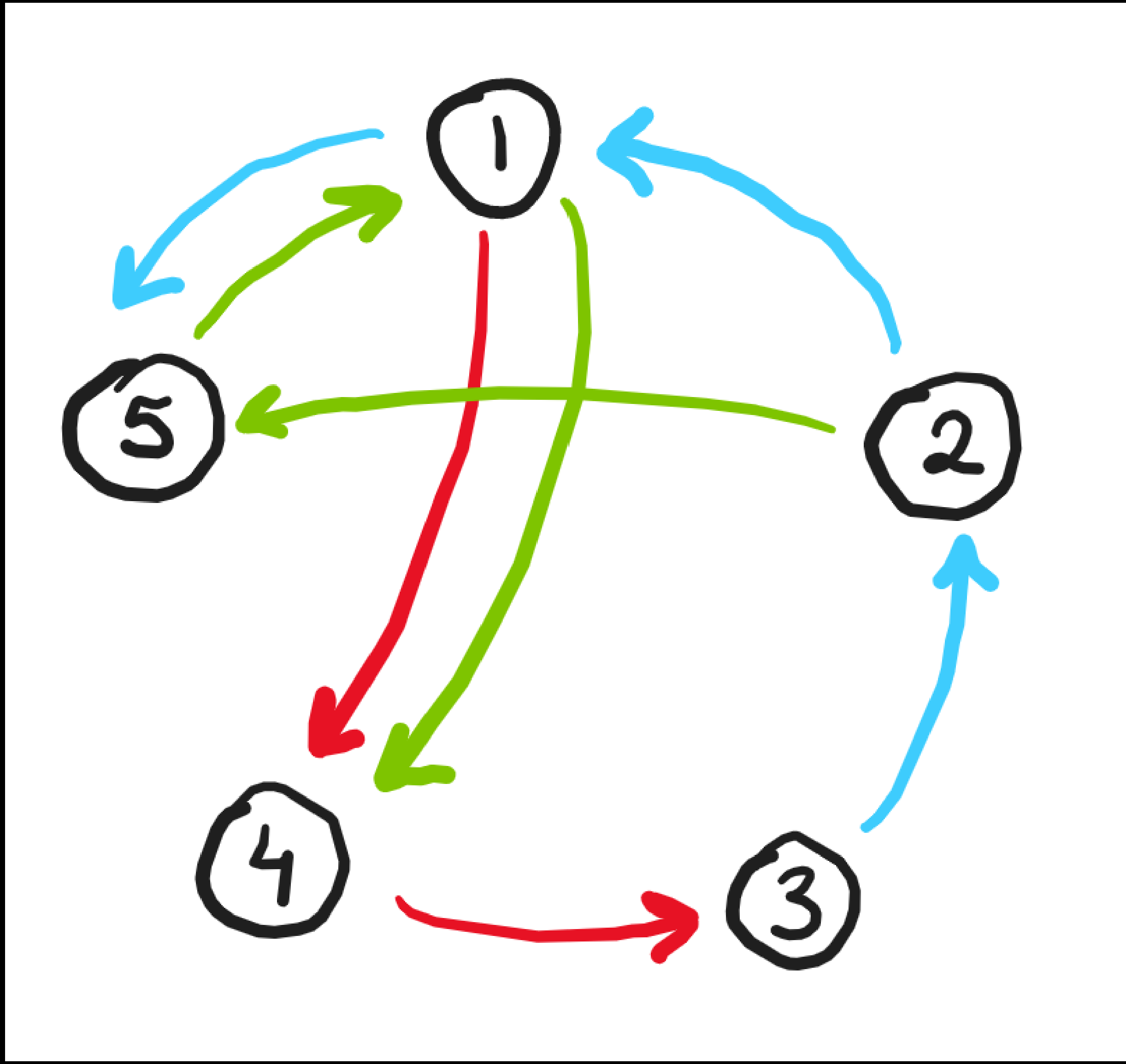
RESULTS

```
x[i,j,r] [*,*,1]
:   1   2   3   4   5    :=
1   0   0   0   1   0
2   0   0   0   0   0
3   0   0   0   0   0
4   0   0   1   0   0
5   0   0   0   0   0

 [*,*,2]
:   1   2   3   4   5    :=
1   0   0   1   0   0
2   0   0   0   0   1
3   0   0   0   1   0
4   0   0   0   0   0
5   1   0   0   0   0

 [*,*,3]
:   1   2   3   4   5    :=
1   0   0   0   0   1
2   1   0   0   0   0
3   0   1   0   0   0
4   0   0   0   0   0
5   0   0   0   0   0
;

z = 155.954
```

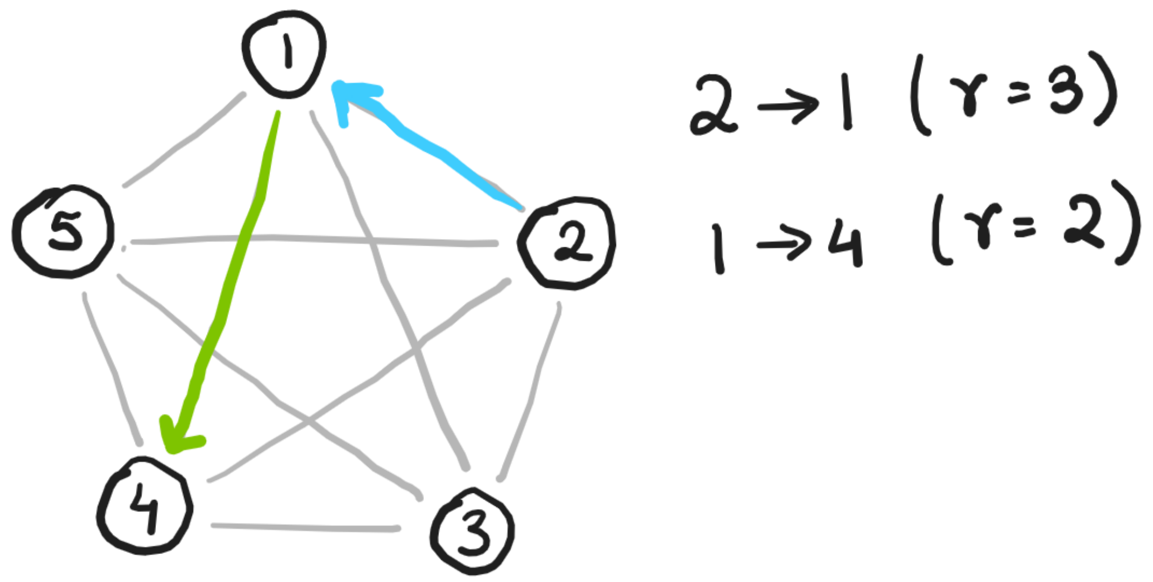


```
p[2,4,i,j,r] [*,*,1]
:   1   2   3   4   5    :=
1   0   0   0   1   0
2   0   0   0   0   0
3   0   0   0   0   0
4   0   0   0   0   0
5   0   0   0   0   0

 [*,*,2]
:   1   2   3   4   5    :=
1   0   0   0   0   0
2   0   0   0   0   0
3   0   0   0   0   0
4   0   0   0   0   0
5   0   0   0   0   0

 [*,*,3]
:   1   2   3   4   5    :=
1   0   0   0   0   0
2   1   0   0   0   0
3   0   0   0   0   0
4   0   0   0   0   0
5   0   0   0   0   0
;

t[2,4,i] [*] :=
1  1
2  0
3  0
4  0
5  0
;
```

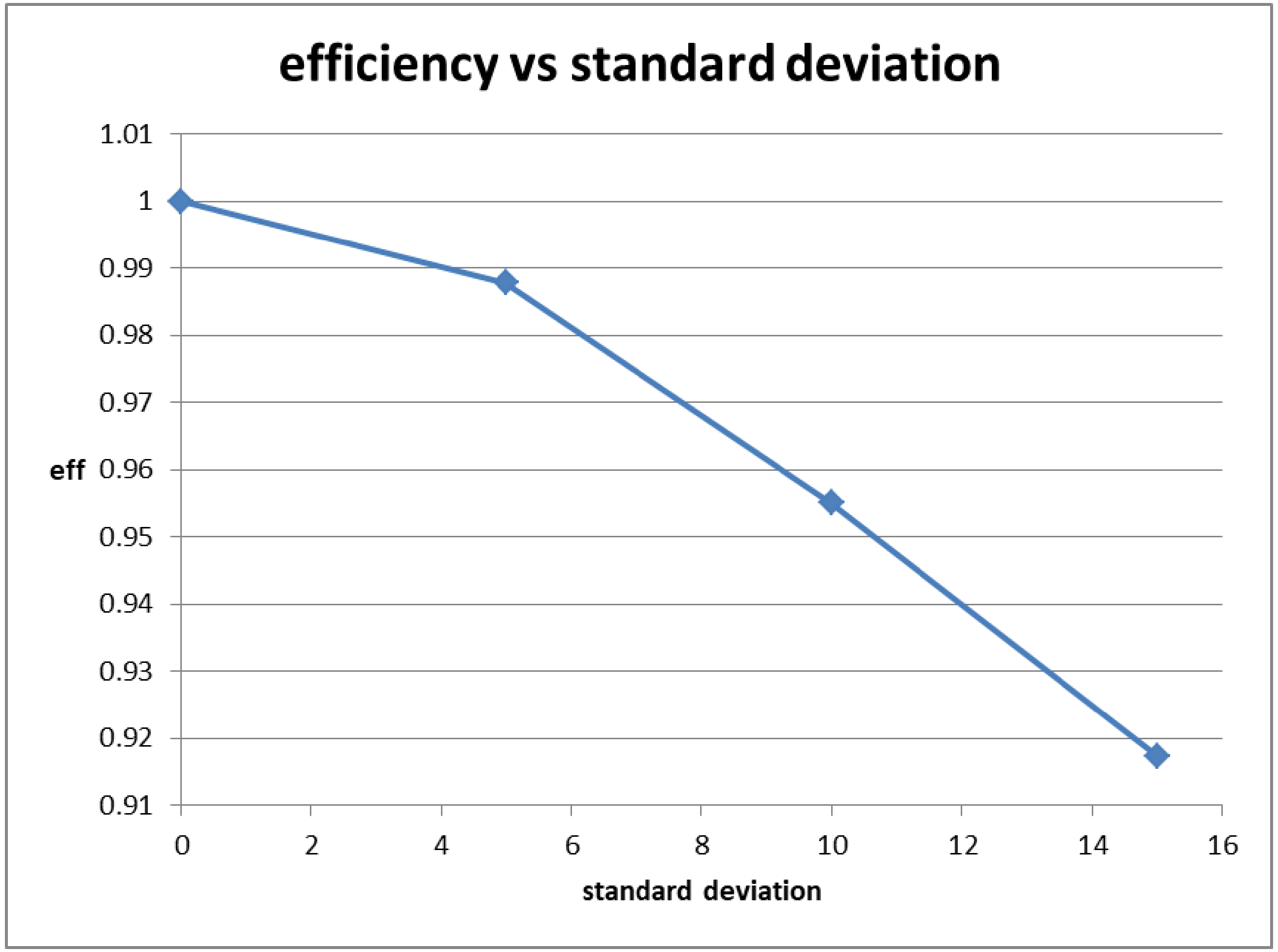


MODEL ROBUSTNESS

- In order to access the robustness of the model, we use greedy method to simulate the model in real world.
- The demand is sampled from gaussian distribution with mean as the demand used during optimization and fixed standard deviation.
- We get the fraction of total passengers served over the entire demand.
- The average of the score gives a good idea about the robustness of the model.

RESULTS

- Using greedy method, the fraction of passengers that can get on the bus depends on number of passenger already on the bus and its capacity.
- We see pretty high fraction of passengers served.
- The fraction of demand served reduces as the uncertainty in demand increase



RESULTS

- We observe that the optimised model depends on the weights assigned to each term in the objective function.

- Changing weights can aggressively

  - Minimise the number of bus changes passengers have to make.
  - Minimise the average cost of travel for passengers.
  - Maximise the number of passengers served
  - Minimise the operating cost.

MODEL

- https://github.com/sumeet-ranjan/bus-route-optimization