

# Case – Scheduling Student Volunteers for the INFORMS Annual Meeting

Ruiwei Gao, Costa Huang, Giang Nguyen

October 30, 2017

## 1 Introduction

The Institute for Operations Research and the Management Sciences (INFORMS) holds its Annual Meeting each fall. Every year, student volunteers get stationed throughout the conference facility/ies each day of the conference to help out with the meeting. For this year’s conference, our consulting group is contracted to assist INFORMS in scheduling 14 students across eight shifts. Two shifts are covered each day (Sunday, Monday, Tuesday, Wednesday): an AM shift from 7:45 am to 12:30 pm and a PM shift from 1:00 pm to 6:00 pm. In addition, a specific shift is referred to by “shift number,” numbered sequentially from one through eight. We have received a spreadsheet named “Case Information” from INFORMS, including:

- “Shift list” worksheet: The “shifts” and the number of volunteers required
- “Student Volunteer Availability” worksheet: Volunteer availability
- Special scheduling requirements

Those information will be shown in details in the sections that are yet to come. Your job is to devise a schedule that meets organization needs (e.g. places the necessary number of students throughout the conference on each day of the conference) and the volunteers needs (e.g. meet the volunteers’ availability). Furthermore, you need to satisfy all other constraints.

## 2 Selection of Optimization Software

After researching into available optimization softwares, we have decided to use **Gurobipy**[1] package. It is fairly to use. To start off, let’s create an empty LP model:

```
In [90]: from gurobipy import *
import pandas

# create a LP model
model = Model()
```

## 3 Variables

In the end, we want to know how to assign students to each shift. For this purpose, let’s set up binary variable  $X_{i \sim j}$  to denote whether student  $i$  will work at  $j$ , where  $j$  encodes the shifts and the location of the shifts, for

$$i \in A = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14\}$$

$$j \in B = \{1s1, 2s1, 3m1, 4m1, 5t1, 6t1, 5t2, 6t2, 5t3, 6t3, 7w1, 8w1, 7w2, 8w2, 5b, 6b, 7b, 8b\}$$

For example,  $X_{2\sim 6t2} = 1$  means that student 2 is assigned to volunteer during shift 6 at location T2. Notice we also have  $5b, 6b, 7b, 8b \in B$  because we want to deal with the special constraints. In this case,  $b$  means the backup location. For example,  $X_{2\sim 7b} = 1$  means that student 2 is assigned to be the backup personel at shift 7

Now, let's add it to our model.

```
In [91]: # Set Variables
i_set = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14]
j_set = ['1s1', '2s1', '3m1', '4m1', '5t1', '6t1', '5t2', '6t2',
         '5t3', '6t3', '7w1', '8w1', '7w2', '8w2', '5b', '6b', '7b', '8b']

x = {}
for i in i_set:
    for j in j_set:
        x[str(i)+'~'+j] = model.addVar(
            vtype=GRB.BINARY,
            name=str(i) + '~' + j
        )

model.update()
```

A typical variable will look like the following outputs. If you are interested in a complete list of variables, refer to **Variable Lists** in the Appendix section

```
In [92]: x['2~1s1']
```

```
Out[92]: <gurobi.Var 2~1s1>
```

## 4 Objective function

Along with this variable, our objective is to minimize the total number of shifts assigned to volunteer because INFORMS is a considerate organization that don't overwork its volunteers. Mathematically, our objective function is

$$\min \sum_{i \in A} \sum_{j \in B} X_{i \sim j}$$

Now, we are ready to set the objective function for the model:

```
In [93]: # Set objective
model.setObjective(
    quicksum(x[key] for key in x.keys()),
    GRB.MINIMIZE
)
```

## 5 Constraints

Now, let's deal with the constraints one by one. For each of the constraint formulation, we will print out one example constraint. But if you are interested in the exhaustive list of constraints, please refer to the Appendix.

### 5.1 Meet the conference demands

The INFORM annual meeting requires certain amount of volunteers during each shifts. Namely:

$$\begin{aligned} \sum_{i \in A} X_{i \sim 1s1} &\geq 2 & \sum_{i \in A} X_{i \sim 2s1} &\geq 2 \\ \sum_{i \in A} X_{i \sim 3m1} &\geq 2 & \sum_{i \in A} X_{i \sim 4m1} &\geq 2 \\ \sum_{i \in A} X_{i \sim 5t1} &\geq 2 & \sum_{i \in A} X_{i \sim 6t1} &\geq 2 \\ \sum_{i \in A} X_{i \sim 5t2} &\geq 1 & \sum_{i \in A} X_{i \sim 6t2} &\geq 1 \\ \sum_{i \in A} X_{i \sim 5t3} &\geq 1 & \sum_{i \in A} X_{i \sim 6t3} &\geq 1 \\ \sum_{i \in A} X_{i \sim 7w1} &\geq 2 & \sum_{i \in A} X_{i \sim 8w1} &\geq 2 \\ \sum_{i \in A} X_{i \sim 7w2} &\geq 1 & \sum_{i \in A} X_{i \sim 8w2} &\geq 1 \end{aligned}$$

Now, let's add the constraints into the model:

```
In [94]: # Use c to keep track of all constraints
c = {}
# use c1 to keep track constraints for meeting the conference demands
c1 = {}

# 1. meet the conference demands
conference_demands = [2,2,2,2,2,2,1,1,1,1,1,2,2,1,1]
c_i = 0 # conference index
conference_shift = j_set[:-4] # without '5b'...
for j in conference_shift:
    c['x_i~' + j] = model.addConstr(
        quicksum(x[str(i)+'~' + j] for i in i_set) >= conference_demands[c_i],
        name='x_i~' + j + ' >= ' + str(conference_demands[c_i])
    )
    c1['x_i~' + j] = c['x_i~' + j]
    c_i += 1

model.update()
c1
```

```
Out[94]: {'x_i~1s1': <gurobi.Constr x_i~1s1 >= 2>,
 'x_i~2s1': <gurobi.Constr x_i~2s1 >= 2>,
 'x_i~3m1': <gurobi.Constr x_i~3m1 >= 2>,
 'x_i~4m1': <gurobi.Constr x_i~4m1 >= 2>,
 'x_i~5t1': <gurobi.Constr x_i~5t1 >= 2>,
 'x_i~5t2': <gurobi.Constr x_i~5t2 >= 1>,
 'x_i~5t3': <gurobi.Constr x_i~5t3 >= 1>,
 'x_i~6t1': <gurobi.Constr x_i~6t1 >= 2>,
 'x_i~6t2': <gurobi.Constr x_i~6t2 >= 1>,
 'x_i~6t3': <gurobi.Constr x_i~6t3 >= 1>,
 'x_i~7w1': <gurobi.Constr x_i~7w1 >= 2>,
 'x_i~7w2': <gurobi.Constr x_i~7w2 >= 1>,
 'x_i~8w1': <gurobi.Constr x_i~8w1 >= 2>,
 'x_i~8w2': <gurobi.Constr x_i~8w2 >= 1>}
```

## 5.2 Meet the volunteers availability

We did two preprocessings to the availability matrix.

- First, the original availability use 0 to denote a volunteer is available and 1 to denote otherwise. Such notation is counter-intuitive and we just reverse the matrix (swap the 0's and 1's).
- Second, we also add columns for  $5b, 6b, 7b, 8b$  for programming convenience, since if a student is available for shift  $i$ , he/she must be available for becoming the "back up" at shift  $i$

Now, let's read the processed matrix:

```
In [95]: # 2. meet the volunteers availability
availability = pandas.read_excel('Availability.xlsx').transpose()
pandas.read_excel('Availability.xlsx')
```

```
Out[95]:
```

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1s1	1	0	0	0	0	1	0	0	1	0	1	0	1	0
2s1	1	1	0	0	0	1	0	1	1	1	1	0	1	0
3m1	0	1	0	1	1	0	0	1	1	1	1	1	1	1
4m1	1	1	1	1	1	0	1	1	1	1	1	1	1	1
5t1	0	0	1	1	1	1	0	1	1	1	1	1	1	1
5t2	0	0	1	1	1	1	0	1	1	1	1	1	1	1
5t3	0	0	1	1	1	1	0	1	1	1	1	1	1	1
5b	0	0	1	1	1	1	0	1	1	1	1	1	1	1
6t1	0	1	1	1	0	0	1	1	1	0	1	1	0	1
6t2	0	1	1	1	0	0	1	1	1	0	1	1	0	1
6t3	0	1	1	1	0	0	1	1	1	0	1	1	0	1
6b	0	1	1	1	0	0	1	1	1	0	1	1	0	1
7w1	0	0	1	1	1	1	0	0	1	1	1	1	0	1
7w2	0	0	1	1	1	1	0	0	1	1	1	1	0	1
7b	0	0	1	1	1	1	0	0	1	1	1	1	0	1
8w1	1	1	1	0	1	1	1	0	1	0	1	0	0	0

8w2	1	1	1	0	1	1	1	0	1	0	1	0	0	0
8b	1	1	1	0	1	1	1	0	1	0	1	0	0	0

In order to satisfy the second constraints, we need to have

$$X_{i\sim j} \leq A_{i\sim j} \quad \text{for } i \in A, j \in B$$

Now, let's add those constraints to the models

```
In [96]: for i in i_set:
          for j in j_set:
              constraint_name = str(i)+'~'+j + " <= A_" + str(i) + '~' + j
              c[constraint_name] = model.addConstr(
                  x[str(i)+'~'+j] <= availability.loc[i, j],
                  name=str(i)+'~'+j + " <= A_" + str(i) + '~' + j
              )

          model.update()
          # A typical constraint look like this:
          c['10~1s1 <= A_10~1s1']
```

```
Out[96]: <gurobi.Constr 10~1s1 <= A_10~1s1>
```

Some typical constraints will look like the following outputs. If you are interested in a complete list of variables, refer to **Constraints List** in the Appendix section

### 5.3 Special Constraints

There are some extra constraints to consider:

- Student 2 and 5 will at most serve one shifts, instead of two shifts.
- There must be one extra students assigned to each shift on Tuesday and Wednesday (Shifts 5,6,7,8) as "backup". This backup students will not be assigned to a location.

#### 5.3.1 Special constraint 1

We simply say

$$\sum_j X_{2\sim j} \leq 1 \quad \sum_j X_{5\sim j} \leq 1$$

for  $j \in B$ . Also notice this also indicate that each other volunteer can work up to two shifts:

$$\sum_j X_{1\sim j} \leq 2 \quad \sum_j X_{3\sim j} \leq 2$$

$$\sum_j X_{4\sim j} \leq 2 \quad \sum_j X_{6\sim j} \leq 2$$

$$\sum_j X_{7\sim j} \leq 2 \quad \sum_j X_{8\sim j} \leq 2$$

$$\begin{aligned} \sum_j X_{9\sim j} &\leq 2 & \sum_j X_{10\sim j} &\leq 2 \\ \sum_j X_{11\sim j} &\leq 2 & \sum_j X_{12\sim j} &\leq 2 \\ \sum_j X_{13\sim j} &\leq 2 & \sum_j X_{14\sim j} &\leq 2 \end{aligned}$$

Now let's add them to the model:

```
In [97]: c["x_2~j <= 1"] = model.addConstr(
    quicksum(x['2~' + j] for j in j_set) <= 1,
    name="x_2~j <= 1"
)
c["x_5~j <= 1"] = model.addConstr(
    quicksum(x['5~' + j] for j in j_set) <= 1,
    name="x_5~j <= 1"
)

all_other_volunteers = [1,3,4,6,7,8,9,10,11,12,13,14]
for i in all_other_volunteers:
    c["x_"+str(i)+"~j <= 2"] = model.addConstr(
        quicksum(x[str(i)+ '~' + j] for j in j_set) <= 2,
        name="x_"+str(i)+"~j <= 2"
    )

model.update()
# A typical constraint look like this:
c["x_5~j <= 1"]
```

```
Out[97]: <gurobi.Constr x_5~j <= 1>
```

### 5.3.2 Special constraint 2

This is where our extra  $5b, 6b, 7b, 8b$  become very useful, we can simply say

$$\begin{aligned} \sum_i X_{i\sim 5b} &\geq 1 & \sum_i X_{i\sim 6b} &\geq 1 \\ \sum_i X_{i\sim 7b} &\geq 1 & \sum_i X_{i\sim 8b} &\geq 1 \end{aligned}$$

Now let's add them to the model

```
In [98]: c["x_i~5b >= 1"] = model.addConstr(
    quicksum(x[str(i)+'~5b'] for i in i_set) >= 1,
    name="x_i~5b >= 1"
)
c["x_i~6b >= 1"] = model.addConstr(
    quicksum(x[str(i)+'~6b'] for i in i_set) >= 1,
```

```

        name="x_i~6b >= 1"
    )
    c["x_i~7b >= 1"] = model.addConstr(
        quicksum(x[str(i)+'~7b'] for i in i_set) >= 1,
        name="x_i~7b >= 1"
    )
    c["x_i~8b >= 1"] = model.addConstr(
        quicksum(x[str(i)+'~8b'] for i in i_set) >= 1,
        name="x_i~8b >= 1"
    )

    model.update()
    # A typical constraint look like this:
    c["x_i~8b >= 1"]

```

Out [98]: <gurobi.Constr x\_i~8b >= 1>

## 5.4 Implicit constraints

### 5.4.1 No volunteers can be assigned to different location during the same shift

Notice that volunteers can only be assigned to different location during shift 5,6,7,8. So we need to prevent such absurd situation. Thus we should have

$$X_{i\sim 5t1} + X_{i\sim 5t2} + X_{i\sim 5t3} + X_{i\sim 5b} \leq 1$$

$$X_{i\sim 6t1} + X_{i\sim 6t2} + X_{i\sim 6t3} + X_{i\sim 6b} \leq 1$$

$$X_{i\sim 7w1} + X_{i\sim 7w2} + X_{i\sim 7b} \leq 1$$

$$X_{i\sim 8w1} + X_{i\sim 8w2} + X_{i\sim 8b} \leq 1$$

for  $i \in A$ . Now let's add them to the model:

```

In [99]: # 4. n
        for i in i_set:
            all_location = ['5t1', '5t2', '5t3', '5b']
            c["x_" + str(i) + "~5(k) <= 1"] = model.addConstr(
                quicksum(x[str(i)+'~' + k] for k in all_location) <= 1,
                name="x_" + str(i) + "~5(k) <= 1"
            )
            all_location = ['6t1', '6t2', '6t3', '6b']
            c["x_" + str(i) + "~6(k) <= 1"] = model.addConstr(
                quicksum(x[str(i)+'~' + k] for k in all_location) <= 1,
                name="x_" + str(i) + "~6(k) <= 1"
            )
            all_location = ['7w1', '7w2', '7b']
            c["x_" + str(i) + "~7(k) <= 1"] = model.addConstr(
                quicksum(x[str(i)+'~' + k] for k in all_location) <= 1,
                name="x_" + str(i) + "~7(k) <= 1"
            )

```

```

all_location = ['8w1', '8w2', '8b']
c["x_" + str(i) + "~8(k) <= 1"] = model.addConstr(
    quicksum(x[str(i)+'~' + k] for k in all_location) <= 1,
    name="x_" + str(i) + "~8(k) <= 1"
)

model.update()
# A typical constraint look like this:
c["x_1~8(k) <= 1"]

```

Out[99]: <gurobi.Constr x\_1~8(k) <= 1>

## 6 Run the model

Now, let's get the results

```

In [100]: model.optimize()
print()
print("*****Thus the objective value is " + str(model.ObjVal))

```

Optimize a model with 340 rows, 252 columns and 952 nonzeros

Variable types: 0 continuous, 252 integer (252 binary)

Coefficient statistics:

```

Matrix range      [1e+00, 1e+00]
Objective range   [1e+00, 1e+00]
Bounds range      [1e+00, 1e+00]
RHS range         [1e+00, 2e+00]

```

Found heuristic solution: objective 26

Presolve removed 276 rows and 85 columns

Presolve time: 0.00s

Presolved: 64 rows, 167 columns, 448 nonzeros

Variable types: 0 continuous, 167 integer (167 binary)

Root relaxation: cutoff, 20 iterations, 0.00 seconds

Nodes		Current Node			Objective Bounds			Work	
Expl	Unexpl	Obj	Depth	IntInf	Incumbent	BestBd	Gap	It/Node	Time
0	0	cutoff	0		26.00000	26.00000	0.00%	-	0s

Explored 0 nodes (20 simplex iterations) in 0.03 seconds

Thread count was 8 (of 8 available processors)

Solution count 1: 26

Optimal solution found (tolerance 1.00e-04)

Best objective 2.600000000000e+01, best bound 2.600000000000e+01, gap 0.0000%



\*\*\*\*\*Thus the objective value is 26.0

## 7 Solution

Thus the minimal objective value is 26, and the resulting schedule matrix is:

```
In [101]: solution = pandas.DataFrame(index = i_set, columns=j_set)
         for key in x.keys():
             key_split = key.split('~')
             row = int(key_split[0])
             col = key_split[1]
             solution.set_value(row, col, x[key].X)
```

solution

```
Out[101]:
```

	1s1	2s1	3m1	4m1	5t1	6t1	5t2	6t2	5t3	6t3	7w1	8w1	7w2	8w2	5b	6b	7b	8b
1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
3	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1
4	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0
5	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0
7	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0
8	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0
9	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0
11	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0
12	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0
13	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0

## 8 For Fun

Remember I said INFORMS is a considerate organization? Sorry, that was a lie. INFORMS is actually a cruel capitalist machine that exploits its volunteers! Therefore its objective function is to maximize the number of shifts that is delegated to the employees:

$$\max \sum_{i \in A} \sum_{j \in B} X_{i \sim j}$$

Then, let's run the model again.

```
In [102]: model.setObjective(
           quicksum(x[key] for key in x.keys()),
           GRB.MINIMIZE
           )
```

```
model.update()
model.optimize()
```

Optimize a model with 340 rows, 252 columns and 952 nonzeros

Variable types: 0 continuous, 252 integer (252 binary)

Coefficient statistics:

```
Matrix range      [1e+00, 1e+00]
Objective range   [1e+00, 1e+00]
Bounds range      [1e+00, 1e+00]
RHS range         [1e+00, 2e+00]
```

Loaded MIP start with objective 26

Presolve removed 276 rows and 85 columns

Presolve time: 0.00s

Presolved: 64 rows, 167 columns, 448 nonzeros

Variable types: 0 continuous, 167 integer (167 binary)

Root relaxation: cutoff, 20 iterations, 0.00 seconds

Nodes		Current Node			Objective Bounds			Work	
Expl	Unexpl	Obj	Depth	IntInf	Incumbent	BestBd	Gap	It/Node	Time
0	0	cutoff	0		26.00000	26.00000	0.00%	-	0s

Explored 0 nodes (20 simplex iterations) in 0.02 seconds

Thread count was 8 (of 8 available processors)

Solution count 1: 26

Optimal solution found (tolerance 1.00e-04)

Best objective 2.600000000000e+01, best bound 2.600000000000e+01, gap 0.0000%

Notice that we still reach 26 because of the constraint that each volunteer can only work up to 2 shifts. Oh well.

## 9 Reference

[1] <http://www.gurobi.com/downloads/get-anaconda>

## 10 Appendix

### 10.1 Variables List

In [103]: x

```
Out[103]: {'10~1s1': <gurobi.Var 10~1s1 (value 0.0)>,
           '10~2s1': <gurobi.Var 10~2s1 (value 0.0)>,
           '10~3m1': <gurobi.Var 10~3m1 (value 0.0)>,
           '10~4m1': <gurobi.Var 10~4m1 (value 1.0)>,
           '10~5b': <gurobi.Var 10~5b (value 0.0)>,
           '10~5t1': <gurobi.Var 10~5t1 (value 0.0)>,
           '10~5t2': <gurobi.Var 10~5t2 (value 0.0)>,
           '10~5t3': <gurobi.Var 10~5t3 (value 0.0)>,
           '10~6b': <gurobi.Var 10~6b (value 0.0)>,
           '10~6t1': <gurobi.Var 10~6t1 (value 0.0)>,
           '10~6t2': <gurobi.Var 10~6t2 (value 0.0)>,
           '10~6t3': <gurobi.Var 10~6t3 (value 0.0)>,
           '10~7b': <gurobi.Var 10~7b (value 0.0)>,
           '10~7w1': <gurobi.Var 10~7w1 (value 1.0)>,
           '10~7w2': <gurobi.Var 10~7w2 (value 0.0)>,
           '10~8b': <gurobi.Var 10~8b (value 0.0)>,
           '10~8w1': <gurobi.Var 10~8w1 (value 0.0)>,
           '10~8w2': <gurobi.Var 10~8w2 (value 0.0)>,
           '11~1s1': <gurobi.Var 11~1s1 (value 0.0)>,
           '11~2s1': <gurobi.Var 11~2s1 (value 0.0)>,
           '11~3m1': <gurobi.Var 11~3m1 (value 0.0)>,
           '11~4m1': <gurobi.Var 11~4m1 (value 1.0)>,
           '11~5b': <gurobi.Var 11~5b (value 0.0)>,
           '11~5t1': <gurobi.Var 11~5t1 (value 0.0)>,
           '11~5t2': <gurobi.Var 11~5t2 (value 0.0)>,
           '11~5t3': <gurobi.Var 11~5t3 (value 0.0)>,
           '11~6b': <gurobi.Var 11~6b (value 0.0)>,
           '11~6t1': <gurobi.Var 11~6t1 (value 0.0)>,
           '11~6t2': <gurobi.Var 11~6t2 (value 0.0)>,
           '11~6t3': <gurobi.Var 11~6t3 (value 0.0)>,
           '11~7b': <gurobi.Var 11~7b (value 1.0)>,
           '11~7w1': <gurobi.Var 11~7w1 (value 0.0)>,
           '11~7w2': <gurobi.Var 11~7w2 (value 0.0)>,
           '11~8b': <gurobi.Var 11~8b (value 0.0)>,
           '11~8w1': <gurobi.Var 11~8w1 (value 0.0)>,
           '11~8w2': <gurobi.Var 11~8w2 (value 0.0)>,
           '12~1s1': <gurobi.Var 12~1s1 (value 0.0)>,
           '12~2s1': <gurobi.Var 12~2s1 (value 0.0)>,
           '12~3m1': <gurobi.Var 12~3m1 (value 0.0)>,
           '12~4m1': <gurobi.Var 12~4m1 (value 0.0)>,
           '12~5b': <gurobi.Var 12~5b (value 0.0)>,
           '12~5t1': <gurobi.Var 12~5t1 (value 0.0)>,
           '12~5t2': <gurobi.Var 12~5t2 (value 0.0)>,}
```

'12~5t3': <gurobi.Var 12~5t3 (value 0.0)>,  
'12~6b': <gurobi.Var 12~6b (value 0.0)>,  
'12~6t1': <gurobi.Var 12~6t1 (value 0.0)>,  
'12~6t2': <gurobi.Var 12~6t2 (value 0.0)>,  
'12~6t3': <gurobi.Var 12~6t3 (value 1.0)>,  
'12~7b': <gurobi.Var 12~7b (value 0.0)>,  
'12~7w1': <gurobi.Var 12~7w1 (value 1.0)>,  
'12~7w2': <gurobi.Var 12~7w2 (value 0.0)>,  
'12~8b': <gurobi.Var 12~8b (value 0.0)>,  
'12~8w1': <gurobi.Var 12~8w1 (value 0.0)>,  
'12~8w2': <gurobi.Var 12~8w2 (value 0.0)>,  
'13~1s1': <gurobi.Var 13~1s1 (value 0.0)>,  
'13~2s1': <gurobi.Var 13~2s1 (value 0.0)>,  
'13~3m1': <gurobi.Var 13~3m1 (value 1.0)>,  
'13~4m1': <gurobi.Var 13~4m1 (value 0.0)>,  
'13~5b': <gurobi.Var 13~5b (value 0.0)>,  
'13~5t1': <gurobi.Var 13~5t1 (value 1.0)>,  
'13~5t2': <gurobi.Var 13~5t2 (value 0.0)>,  
'13~5t3': <gurobi.Var 13~5t3 (value 0.0)>,  
'13~6b': <gurobi.Var 13~6b (value 0.0)>,  
'13~6t1': <gurobi.Var 13~6t1 (value 0.0)>,  
'13~6t2': <gurobi.Var 13~6t2 (value 0.0)>,  
'13~6t3': <gurobi.Var 13~6t3 (value 0.0)>,  
'13~7b': <gurobi.Var 13~7b (value 0.0)>,  
'13~7w1': <gurobi.Var 13~7w1 (value 0.0)>,  
'13~7w2': <gurobi.Var 13~7w2 (value 0.0)>,  
'13~8b': <gurobi.Var 13~8b (value 0.0)>,  
'13~8w1': <gurobi.Var 13~8w1 (value 0.0)>,  
'13~8w2': <gurobi.Var 13~8w2 (value 0.0)>,  
'14~1s1': <gurobi.Var 14~1s1 (value 0.0)>,  
'14~2s1': <gurobi.Var 14~2s1 (value 0.0)>,  
'14~3m1': <gurobi.Var 14~3m1 (value 0.0)>,  
'14~4m1': <gurobi.Var 14~4m1 (value 0.0)>,  
'14~5b': <gurobi.Var 14~5b (value 1.0)>,  
'14~5t1': <gurobi.Var 14~5t1 (value 0.0)>,  
'14~5t2': <gurobi.Var 14~5t2 (value 0.0)>,  
'14~5t3': <gurobi.Var 14~5t3 (value 0.0)>,  
'14~6b': <gurobi.Var 14~6b (value 0.0)>,  
'14~6t1': <gurobi.Var 14~6t1 (value 1.0)>,  
'14~6t2': <gurobi.Var 14~6t2 (value 0.0)>,  
'14~6t3': <gurobi.Var 14~6t3 (value 0.0)>,  
'14~7b': <gurobi.Var 14~7b (value 0.0)>,  
'14~7w1': <gurobi.Var 14~7w1 (value 0.0)>,  
'14~7w2': <gurobi.Var 14~7w2 (value 0.0)>,  
'14~8b': <gurobi.Var 14~8b (value 0.0)>,  
'14~8w1': <gurobi.Var 14~8w1 (value 0.0)>,  
'14~8w2': <gurobi.Var 14~8w2 (value 0.0)>,  
'1~1s1': <gurobi.Var 1~1s1 (value 1.0)>,

'1~2s1': <gurobi.Var 1~2s1 (value 1.0)>,  
'1~3m1': <gurobi.Var 1~3m1 (value 0.0)>,  
'1~4m1': <gurobi.Var 1~4m1 (value 0.0)>,  
'1~5b': <gurobi.Var 1~5b (value 0.0)>,  
'1~5t1': <gurobi.Var 1~5t1 (value 0.0)>,  
'1~5t2': <gurobi.Var 1~5t2 (value 0.0)>,  
'1~5t3': <gurobi.Var 1~5t3 (value 0.0)>,  
'1~6b': <gurobi.Var 1~6b (value 0.0)>,  
'1~6t1': <gurobi.Var 1~6t1 (value 0.0)>,  
'1~6t2': <gurobi.Var 1~6t2 (value 0.0)>,  
'1~6t3': <gurobi.Var 1~6t3 (value 0.0)>,  
'1~7b': <gurobi.Var 1~7b (value 0.0)>,  
'1~7w1': <gurobi.Var 1~7w1 (value 0.0)>,  
'1~7w2': <gurobi.Var 1~7w2 (value 0.0)>,  
'1~8b': <gurobi.Var 1~8b (value 0.0)>,  
'1~8w1': <gurobi.Var 1~8w1 (value 0.0)>,  
'1~8w2': <gurobi.Var 1~8w2 (value 0.0)>,  
'2~1s1': <gurobi.Var 2~1s1 (value 0.0)>,  
'2~2s1': <gurobi.Var 2~2s1 (value 0.0)>,  
'2~3m1': <gurobi.Var 2~3m1 (value 0.0)>,  
'2~4m1': <gurobi.Var 2~4m1 (value 0.0)>,  
'2~5b': <gurobi.Var 2~5b (value 0.0)>,  
'2~5t1': <gurobi.Var 2~5t1 (value 0.0)>,  
'2~5t2': <gurobi.Var 2~5t2 (value 0.0)>,  
'2~5t3': <gurobi.Var 2~5t3 (value 0.0)>,  
'2~6b': <gurobi.Var 2~6b (value 0.0)>,  
'2~6t1': <gurobi.Var 2~6t1 (value 0.0)>,  
'2~6t2': <gurobi.Var 2~6t2 (value 0.0)>,  
'2~6t3': <gurobi.Var 2~6t3 (value 0.0)>,  
'2~7b': <gurobi.Var 2~7b (value 0.0)>,  
'2~7w1': <gurobi.Var 2~7w1 (value 0.0)>,  
'2~7w2': <gurobi.Var 2~7w2 (value 0.0)>,  
'2~8b': <gurobi.Var 2~8b (value 0.0)>,  
'2~8w1': <gurobi.Var 2~8w1 (value 1.0)>,  
'2~8w2': <gurobi.Var 2~8w2 (value 0.0)>,  
'3~1s1': <gurobi.Var 3~1s1 (value 0.0)>,  
'3~2s1': <gurobi.Var 3~2s1 (value 0.0)>,  
'3~3m1': <gurobi.Var 3~3m1 (value 0.0)>,  
'3~4m1': <gurobi.Var 3~4m1 (value 0.0)>,  
'3~5b': <gurobi.Var 3~5b (value 0.0)>,  
'3~5t1': <gurobi.Var 3~5t1 (value 0.0)>,  
'3~5t2': <gurobi.Var 3~5t2 (value 0.0)>,  
'3~5t3': <gurobi.Var 3~5t3 (value 0.0)>,  
'3~6b': <gurobi.Var 3~6b (value 0.0)>,  
'3~6t1': <gurobi.Var 3~6t1 (value 0.0)>,  
'3~6t2': <gurobi.Var 3~6t2 (value 1.0)>,  
'3~6t3': <gurobi.Var 3~6t3 (value 0.0)>,  
'3~7b': <gurobi.Var 3~7b (value 0.0)>,

'3~7w1': <gurobi.Var 3~7w1 (value 0.0)>,  
'3~7w2': <gurobi.Var 3~7w2 (value 0.0)>,  
'3~8b': <gurobi.Var 3~8b (value 1.0)>,  
'3~8w1': <gurobi.Var 3~8w1 (value 0.0)>,  
'3~8w2': <gurobi.Var 3~8w2 (value 0.0)>,  
'4~1s1': <gurobi.Var 4~1s1 (value 0.0)>,  
'4~2s1': <gurobi.Var 4~2s1 (value 0.0)>,  
'4~3m1': <gurobi.Var 4~3m1 (value 0.0)>,  
'4~4m1': <gurobi.Var 4~4m1 (value 0.0)>,  
'4~5b': <gurobi.Var 4~5b (value 0.0)>,  
'4~5t1': <gurobi.Var 4~5t1 (value 0.0)>,  
'4~5t2': <gurobi.Var 4~5t2 (value 0.0)>,  
'4~5t3': <gurobi.Var 4~5t3 (value 1.0)>,  
'4~6b': <gurobi.Var 4~6b (value 0.0)>,  
'4~6t1': <gurobi.Var 4~6t1 (value 0.0)>,  
'4~6t2': <gurobi.Var 4~6t2 (value 0.0)>,  
'4~6t3': <gurobi.Var 4~6t3 (value 0.0)>,  
'4~7b': <gurobi.Var 4~7b (value 0.0)>,  
'4~7w1': <gurobi.Var 4~7w1 (value 0.0)>,  
'4~7w2': <gurobi.Var 4~7w2 (value 1.0)>,  
'4~8b': <gurobi.Var 4~8b (value 0.0)>,  
'4~8w1': <gurobi.Var 4~8w1 (value 0.0)>,  
'4~8w2': <gurobi.Var 4~8w2 (value 0.0)>,  
'5~1s1': <gurobi.Var 5~1s1 (value 0.0)>,  
'5~2s1': <gurobi.Var 5~2s1 (value 0.0)>,  
'5~3m1': <gurobi.Var 5~3m1 (value 1.0)>,  
'5~4m1': <gurobi.Var 5~4m1 (value 0.0)>,  
'5~5b': <gurobi.Var 5~5b (value 0.0)>,  
'5~5t1': <gurobi.Var 5~5t1 (value 0.0)>,  
'5~5t2': <gurobi.Var 5~5t2 (value 0.0)>,  
'5~5t3': <gurobi.Var 5~5t3 (value 0.0)>,  
'5~6b': <gurobi.Var 5~6b (value 0.0)>,  
'5~6t1': <gurobi.Var 5~6t1 (value 0.0)>,  
'5~6t2': <gurobi.Var 5~6t2 (value 0.0)>,  
'5~6t3': <gurobi.Var 5~6t3 (value 0.0)>,  
'5~7b': <gurobi.Var 5~7b (value 0.0)>,  
'5~7w1': <gurobi.Var 5~7w1 (value 0.0)>,  
'5~7w2': <gurobi.Var 5~7w2 (value 0.0)>,  
'5~8b': <gurobi.Var 5~8b (value 0.0)>,  
'5~8w1': <gurobi.Var 5~8w1 (value 0.0)>,  
'5~8w2': <gurobi.Var 5~8w2 (value 0.0)>,  
'6~1s1': <gurobi.Var 6~1s1 (value 0.0)>,  
'6~2s1': <gurobi.Var 6~2s1 (value 0.0)>,  
'6~3m1': <gurobi.Var 6~3m1 (value 0.0)>,  
'6~4m1': <gurobi.Var 6~4m1 (value 0.0)>,  
'6~5b': <gurobi.Var 6~5b (value 0.0)>,  
'6~5t1': <gurobi.Var 6~5t1 (value 1.0)>,  
'6~5t2': <gurobi.Var 6~5t2 (value 0.0)>,

'6~5t3': <gurobi.Var 6~5t3 (value 0.0)>,  
'6~6b': <gurobi.Var 6~6b (value 0.0)>,  
'6~6t1': <gurobi.Var 6~6t1 (value 0.0)>,  
'6~6t2': <gurobi.Var 6~6t2 (value 0.0)>,  
'6~6t3': <gurobi.Var 6~6t3 (value 0.0)>,  
'6~7b': <gurobi.Var 6~7b (value 0.0)>,  
'6~7w1': <gurobi.Var 6~7w1 (value 0.0)>,  
'6~7w2': <gurobi.Var 6~7w2 (value 0.0)>,  
'6~8b': <gurobi.Var 6~8b (value 0.0)>,  
'6~8w1': <gurobi.Var 6~8w1 (value 0.0)>,  
'6~8w2': <gurobi.Var 6~8w2 (value 1.0)>,  
'7~1s1': <gurobi.Var 7~1s1 (value 0.0)>,  
'7~2s1': <gurobi.Var 7~2s1 (value 0.0)>,  
'7~3m1': <gurobi.Var 7~3m1 (value 0.0)>,  
'7~4m1': <gurobi.Var 7~4m1 (value 0.0)>,  
'7~5b': <gurobi.Var 7~5b (value 0.0)>,  
'7~5t1': <gurobi.Var 7~5t1 (value 0.0)>,  
'7~5t2': <gurobi.Var 7~5t2 (value 0.0)>,  
'7~5t3': <gurobi.Var 7~5t3 (value 0.0)>,  
'7~6b': <gurobi.Var 7~6b (value 0.0)>,  
'7~6t1': <gurobi.Var 7~6t1 (value 1.0)>,  
'7~6t2': <gurobi.Var 7~6t2 (value 0.0)>,  
'7~6t3': <gurobi.Var 7~6t3 (value 0.0)>,  
'7~7b': <gurobi.Var 7~7b (value 0.0)>,  
'7~7w1': <gurobi.Var 7~7w1 (value 0.0)>,  
'7~7w2': <gurobi.Var 7~7w2 (value 0.0)>,  
'7~8b': <gurobi.Var 7~8b (value 0.0)>,  
'7~8w1': <gurobi.Var 7~8w1 (value 1.0)>,  
'7~8w2': <gurobi.Var 7~8w2 (value 0.0)>,  
'8~1s1': <gurobi.Var 8~1s1 (value 0.0)>,  
'8~2s1': <gurobi.Var 8~2s1 (value 0.0)>,  
'8~3m1': <gurobi.Var 8~3m1 (value 0.0)>,  
'8~4m1': <gurobi.Var 8~4m1 (value 0.0)>,  
'8~5b': <gurobi.Var 8~5b (value 0.0)>,  
'8~5t1': <gurobi.Var 8~5t1 (value 0.0)>,  
'8~5t2': <gurobi.Var 8~5t2 (value 1.0)>,  
'8~5t3': <gurobi.Var 8~5t3 (value 0.0)>,  
'8~6b': <gurobi.Var 8~6b (value 1.0)>,  
'8~6t1': <gurobi.Var 8~6t1 (value 0.0)>,  
'8~6t2': <gurobi.Var 8~6t2 (value 0.0)>,  
'8~6t3': <gurobi.Var 8~6t3 (value 0.0)>,  
'8~7b': <gurobi.Var 8~7b (value 0.0)>,  
'8~7w1': <gurobi.Var 8~7w1 (value 0.0)>,  
'8~7w2': <gurobi.Var 8~7w2 (value 0.0)>,  
'8~8b': <gurobi.Var 8~8b (value 0.0)>,  
'8~8w1': <gurobi.Var 8~8w1 (value 0.0)>,  
'8~8w2': <gurobi.Var 8~8w2 (value 0.0)>,  
'9~1s1': <gurobi.Var 9~1s1 (value 1.0)>,

```

'9~2s1': <gurobi.Var 9~2s1 (value 1.0)>,
'9~3m1': <gurobi.Var 9~3m1 (value 0.0)>,
'9~4m1': <gurobi.Var 9~4m1 (value 0.0)>,
'9~5b': <gurobi.Var 9~5b (value 0.0)>,
'9~5t1': <gurobi.Var 9~5t1 (value 0.0)>,
'9~5t2': <gurobi.Var 9~5t2 (value 0.0)>,
'9~5t3': <gurobi.Var 9~5t3 (value 0.0)>,
'9~6b': <gurobi.Var 9~6b (value 0.0)>,
'9~6t1': <gurobi.Var 9~6t1 (value 0.0)>,
'9~6t2': <gurobi.Var 9~6t2 (value 0.0)>,
'9~6t3': <gurobi.Var 9~6t3 (value 0.0)>,
'9~7b': <gurobi.Var 9~7b (value 0.0)>,
'9~7w1': <gurobi.Var 9~7w1 (value 0.0)>,
'9~7w2': <gurobi.Var 9~7w2 (value 0.0)>,
'9~8b': <gurobi.Var 9~8b (value 0.0)>,
'9~8w1': <gurobi.Var 9~8w1 (value 0.0)>,
'9~8w2': <gurobi.Var 9~8w2 (value 0.0)>}

```

## 10.2 Constraints list

In [104]: c

```

Out[104]: {'10~1s1 <= A_10~1s1': <gurobi.Constr 10~1s1 <= A_10~1s1>,
'10~2s1 <= A_10~2s1': <gurobi.Constr 10~2s1 <= A_10~2s1>,
'10~3m1 <= A_10~3m1': <gurobi.Constr 10~3m1 <= A_10~3m1>,
'10~4m1 <= A_10~4m1': <gurobi.Constr 10~4m1 <= A_10~4m1>,
'10~5b <= A_10~5b': <gurobi.Constr 10~5b <= A_10~5b>,
'10~5t1 <= A_10~5t1': <gurobi.Constr 10~5t1 <= A_10~5t1>,
'10~5t2 <= A_10~5t2': <gurobi.Constr 10~5t2 <= A_10~5t2>,
'10~5t3 <= A_10~5t3': <gurobi.Constr 10~5t3 <= A_10~5t3>,
'10~6b <= A_10~6b': <gurobi.Constr 10~6b <= A_10~6b>,
'10~6t1 <= A_10~6t1': <gurobi.Constr 10~6t1 <= A_10~6t1>,
'10~6t2 <= A_10~6t2': <gurobi.Constr 10~6t2 <= A_10~6t2>,
'10~6t3 <= A_10~6t3': <gurobi.Constr 10~6t3 <= A_10~6t3>,
'10~7b <= A_10~7b': <gurobi.Constr 10~7b <= A_10~7b>,
'10~7w1 <= A_10~7w1': <gurobi.Constr 10~7w1 <= A_10~7w1>,
'10~7w2 <= A_10~7w2': <gurobi.Constr 10~7w2 <= A_10~7w2>,
'10~8b <= A_10~8b': <gurobi.Constr 10~8b <= A_10~8b>,
'10~8w1 <= A_10~8w1': <gurobi.Constr 10~8w1 <= A_10~8w1>,
'10~8w2 <= A_10~8w2': <gurobi.Constr 10~8w2 <= A_10~8w2>,
'11~1s1 <= A_11~1s1': <gurobi.Constr 11~1s1 <= A_11~1s1>,
'11~2s1 <= A_11~2s1': <gurobi.Constr 11~2s1 <= A_11~2s1>,
'11~3m1 <= A_11~3m1': <gurobi.Constr 11~3m1 <= A_11~3m1>,
'11~4m1 <= A_11~4m1': <gurobi.Constr 11~4m1 <= A_11~4m1>,
'11~5b <= A_11~5b': <gurobi.Constr 11~5b <= A_11~5b>,
'11~5t1 <= A_11~5t1': <gurobi.Constr 11~5t1 <= A_11~5t1>,
'11~5t2 <= A_11~5t2': <gurobi.Constr 11~5t2 <= A_11~5t2>,
'11~5t3 <= A_11~5t3': <gurobi.Constr 11~5t3 <= A_11~5t3>,

```



```

'11~6b <= A_11~6b': <gurobi.Constr 11~6b <= A_11~6b>,
'11~6t1 <= A_11~6t1': <gurobi.Constr 11~6t1 <= A_11~6t1>,
'11~6t2 <= A_11~6t2': <gurobi.Constr 11~6t2 <= A_11~6t2>,
'11~6t3 <= A_11~6t3': <gurobi.Constr 11~6t3 <= A_11~6t3>,
'11~7b <= A_11~7b': <gurobi.Constr 11~7b <= A_11~7b>,
'11~7w1 <= A_11~7w1': <gurobi.Constr 11~7w1 <= A_11~7w1>,
'11~7w2 <= A_11~7w2': <gurobi.Constr 11~7w2 <= A_11~7w2>,
'11~8b <= A_11~8b': <gurobi.Constr 11~8b <= A_11~8b>,
'11~8w1 <= A_11~8w1': <gurobi.Constr 11~8w1 <= A_11~8w1>,
'11~8w2 <= A_11~8w2': <gurobi.Constr 11~8w2 <= A_11~8w2>,
'12~1s1 <= A_12~1s1': <gurobi.Constr 12~1s1 <= A_12~1s1>,
'12~2s1 <= A_12~2s1': <gurobi.Constr 12~2s1 <= A_12~2s1>,
'12~3m1 <= A_12~3m1': <gurobi.Constr 12~3m1 <= A_12~3m1>,
'12~4m1 <= A_12~4m1': <gurobi.Constr 12~4m1 <= A_12~4m1>,
'12~5b <= A_12~5b': <gurobi.Constr 12~5b <= A_12~5b>,
'12~5t1 <= A_12~5t1': <gurobi.Constr 12~5t1 <= A_12~5t1>,
'12~5t2 <= A_12~5t2': <gurobi.Constr 12~5t2 <= A_12~5t2>,
'12~5t3 <= A_12~5t3': <gurobi.Constr 12~5t3 <= A_12~5t3>,
'12~6b <= A_12~6b': <gurobi.Constr 12~6b <= A_12~6b>,
'12~6t1 <= A_12~6t1': <gurobi.Constr 12~6t1 <= A_12~6t1>,
'12~6t2 <= A_12~6t2': <gurobi.Constr 12~6t2 <= A_12~6t2>,
'12~6t3 <= A_12~6t3': <gurobi.Constr 12~6t3 <= A_12~6t3>,
'12~7b <= A_12~7b': <gurobi.Constr 12~7b <= A_12~7b>,
'12~7w1 <= A_12~7w1': <gurobi.Constr 12~7w1 <= A_12~7w1>,
'12~7w2 <= A_12~7w2': <gurobi.Constr 12~7w2 <= A_12~7w2>,
'12~8b <= A_12~8b': <gurobi.Constr 12~8b <= A_12~8b>,
'12~8w1 <= A_12~8w1': <gurobi.Constr 12~8w1 <= A_12~8w1>,
'12~8w2 <= A_12~8w2': <gurobi.Constr 12~8w2 <= A_12~8w2>,
'13~1s1 <= A_13~1s1': <gurobi.Constr 13~1s1 <= A_13~1s1>,
'13~2s1 <= A_13~2s1': <gurobi.Constr 13~2s1 <= A_13~2s1>,
'13~3m1 <= A_13~3m1': <gurobi.Constr 13~3m1 <= A_13~3m1>,
'13~4m1 <= A_13~4m1': <gurobi.Constr 13~4m1 <= A_13~4m1>,
'13~5b <= A_13~5b': <gurobi.Constr 13~5b <= A_13~5b>,
'13~5t1 <= A_13~5t1': <gurobi.Constr 13~5t1 <= A_13~5t1>,
'13~5t2 <= A_13~5t2': <gurobi.Constr 13~5t2 <= A_13~5t2>,
'13~5t3 <= A_13~5t3': <gurobi.Constr 13~5t3 <= A_13~5t3>,
'13~6b <= A_13~6b': <gurobi.Constr 13~6b <= A_13~6b>,
'13~6t1 <= A_13~6t1': <gurobi.Constr 13~6t1 <= A_13~6t1>,
'13~6t2 <= A_13~6t2': <gurobi.Constr 13~6t2 <= A_13~6t2>,
'13~6t3 <= A_13~6t3': <gurobi.Constr 13~6t3 <= A_13~6t3>,
'13~7b <= A_13~7b': <gurobi.Constr 13~7b <= A_13~7b>,
'13~7w1 <= A_13~7w1': <gurobi.Constr 13~7w1 <= A_13~7w1>,
'13~7w2 <= A_13~7w2': <gurobi.Constr 13~7w2 <= A_13~7w2>,
'13~8b <= A_13~8b': <gurobi.Constr 13~8b <= A_13~8b>,
'13~8w1 <= A_13~8w1': <gurobi.Constr 13~8w1 <= A_13~8w1>,
'13~8w2 <= A_13~8w2': <gurobi.Constr 13~8w2 <= A_13~8w2>,
'14~1s1 <= A_14~1s1': <gurobi.Constr 14~1s1 <= A_14~1s1>,
'14~2s1 <= A_14~2s1': <gurobi.Constr 14~2s1 <= A_14~2s1>,

```

'14~3m1 <= A\_14~3m1': <gurobi.Constr 14~3m1 <= A\_14~3m1>,  
'14~4m1 <= A\_14~4m1': <gurobi.Constr 14~4m1 <= A\_14~4m1>,  
'14~5b <= A\_14~5b': <gurobi.Constr 14~5b <= A\_14~5b>,  
'14~5t1 <= A\_14~5t1': <gurobi.Constr 14~5t1 <= A\_14~5t1>,  
'14~5t2 <= A\_14~5t2': <gurobi.Constr 14~5t2 <= A\_14~5t2>,  
'14~5t3 <= A\_14~5t3': <gurobi.Constr 14~5t3 <= A\_14~5t3>,  
'14~6b <= A\_14~6b': <gurobi.Constr 14~6b <= A\_14~6b>,  
'14~6t1 <= A\_14~6t1': <gurobi.Constr 14~6t1 <= A\_14~6t1>,  
'14~6t2 <= A\_14~6t2': <gurobi.Constr 14~6t2 <= A\_14~6t2>,  
'14~6t3 <= A\_14~6t3': <gurobi.Constr 14~6t3 <= A\_14~6t3>,  
'14~7b <= A\_14~7b': <gurobi.Constr 14~7b <= A\_14~7b>,  
'14~7w1 <= A\_14~7w1': <gurobi.Constr 14~7w1 <= A\_14~7w1>,  
'14~7w2 <= A\_14~7w2': <gurobi.Constr 14~7w2 <= A\_14~7w2>,  
'14~8b <= A\_14~8b': <gurobi.Constr 14~8b <= A\_14~8b>,  
'14~8w1 <= A\_14~8w1': <gurobi.Constr 14~8w1 <= A\_14~8w1>,  
'14~8w2 <= A\_14~8w2': <gurobi.Constr 14~8w2 <= A\_14~8w2>,  
'1~1s1 <= A\_1~1s1': <gurobi.Constr 1~1s1 <= A\_1~1s1>,  
'1~2s1 <= A\_1~2s1': <gurobi.Constr 1~2s1 <= A\_1~2s1>,  
'1~3m1 <= A\_1~3m1': <gurobi.Constr 1~3m1 <= A\_1~3m1>,  
'1~4m1 <= A\_1~4m1': <gurobi.Constr 1~4m1 <= A\_1~4m1>,  
'1~5b <= A\_1~5b': <gurobi.Constr 1~5b <= A\_1~5b>,  
'1~5t1 <= A\_1~5t1': <gurobi.Constr 1~5t1 <= A\_1~5t1>,  
'1~5t2 <= A\_1~5t2': <gurobi.Constr 1~5t2 <= A\_1~5t2>,  
'1~5t3 <= A\_1~5t3': <gurobi.Constr 1~5t3 <= A\_1~5t3>,  
'1~6b <= A\_1~6b': <gurobi.Constr 1~6b <= A\_1~6b>,  
'1~6t1 <= A\_1~6t1': <gurobi.Constr 1~6t1 <= A\_1~6t1>,  
'1~6t2 <= A\_1~6t2': <gurobi.Constr 1~6t2 <= A\_1~6t2>,  
'1~6t3 <= A\_1~6t3': <gurobi.Constr 1~6t3 <= A\_1~6t3>,  
'1~7b <= A\_1~7b': <gurobi.Constr 1~7b <= A\_1~7b>,  
'1~7w1 <= A\_1~7w1': <gurobi.Constr 1~7w1 <= A\_1~7w1>,  
'1~7w2 <= A\_1~7w2': <gurobi.Constr 1~7w2 <= A\_1~7w2>,  
'1~8b <= A\_1~8b': <gurobi.Constr 1~8b <= A\_1~8b>,  
'1~8w1 <= A\_1~8w1': <gurobi.Constr 1~8w1 <= A\_1~8w1>,  
'1~8w2 <= A\_1~8w2': <gurobi.Constr 1~8w2 <= A\_1~8w2>,  
'2~1s1 <= A\_2~1s1': <gurobi.Constr 2~1s1 <= A\_2~1s1>,  
'2~2s1 <= A\_2~2s1': <gurobi.Constr 2~2s1 <= A\_2~2s1>,  
'2~3m1 <= A\_2~3m1': <gurobi.Constr 2~3m1 <= A\_2~3m1>,  
'2~4m1 <= A\_2~4m1': <gurobi.Constr 2~4m1 <= A\_2~4m1>,  
'2~5b <= A\_2~5b': <gurobi.Constr 2~5b <= A\_2~5b>,  
'2~5t1 <= A\_2~5t1': <gurobi.Constr 2~5t1 <= A\_2~5t1>,  
'2~5t2 <= A\_2~5t2': <gurobi.Constr 2~5t2 <= A\_2~5t2>,  
'2~5t3 <= A\_2~5t3': <gurobi.Constr 2~5t3 <= A\_2~5t3>,  
'2~6b <= A\_2~6b': <gurobi.Constr 2~6b <= A\_2~6b>,  
'2~6t1 <= A\_2~6t1': <gurobi.Constr 2~6t1 <= A\_2~6t1>,  
'2~6t2 <= A\_2~6t2': <gurobi.Constr 2~6t2 <= A\_2~6t2>,  
'2~6t3 <= A\_2~6t3': <gurobi.Constr 2~6t3 <= A\_2~6t3>,  
'2~7b <= A\_2~7b': <gurobi.Constr 2~7b <= A\_2~7b>,  
'2~7w1 <= A\_2~7w1': <gurobi.Constr 2~7w1 <= A\_2~7w1>,

'2~7w2 <= A\_2~7w2': <gurobi.Constr 2~7w2 <= A\_2~7w2>,  
 '2~8b <= A\_2~8b': <gurobi.Constr 2~8b <= A\_2~8b>,  
 '2~8w1 <= A\_2~8w1': <gurobi.Constr 2~8w1 <= A\_2~8w1>,  
 '2~8w2 <= A\_2~8w2': <gurobi.Constr 2~8w2 <= A\_2~8w2>,  
 '3~1s1 <= A\_3~1s1': <gurobi.Constr 3~1s1 <= A\_3~1s1>,  
 '3~2s1 <= A\_3~2s1': <gurobi.Constr 3~2s1 <= A\_3~2s1>,  
 '3~3m1 <= A\_3~3m1': <gurobi.Constr 3~3m1 <= A\_3~3m1>,  
 '3~4m1 <= A\_3~4m1': <gurobi.Constr 3~4m1 <= A\_3~4m1>,  
 '3~5b <= A\_3~5b': <gurobi.Constr 3~5b <= A\_3~5b>,  
 '3~5t1 <= A\_3~5t1': <gurobi.Constr 3~5t1 <= A\_3~5t1>,  
 '3~5t2 <= A\_3~5t2': <gurobi.Constr 3~5t2 <= A\_3~5t2>,  
 '3~5t3 <= A\_3~5t3': <gurobi.Constr 3~5t3 <= A\_3~5t3>,  
 '3~6b <= A\_3~6b': <gurobi.Constr 3~6b <= A\_3~6b>,  
 '3~6t1 <= A\_3~6t1': <gurobi.Constr 3~6t1 <= A\_3~6t1>,  
 '3~6t2 <= A\_3~6t2': <gurobi.Constr 3~6t2 <= A\_3~6t2>,  
 '3~6t3 <= A\_3~6t3': <gurobi.Constr 3~6t3 <= A\_3~6t3>,  
 '3~7b <= A\_3~7b': <gurobi.Constr 3~7b <= A\_3~7b>,  
 '3~7w1 <= A\_3~7w1': <gurobi.Constr 3~7w1 <= A\_3~7w1>,  
 '3~7w2 <= A\_3~7w2': <gurobi.Constr 3~7w2 <= A\_3~7w2>,  
 '3~8b <= A\_3~8b': <gurobi.Constr 3~8b <= A\_3~8b>,  
 '3~8w1 <= A\_3~8w1': <gurobi.Constr 3~8w1 <= A\_3~8w1>,  
 '3~8w2 <= A\_3~8w2': <gurobi.Constr 3~8w2 <= A\_3~8w2>,  
 '4~1s1 <= A\_4~1s1': <gurobi.Constr 4~1s1 <= A\_4~1s1>,  
 '4~2s1 <= A\_4~2s1': <gurobi.Constr 4~2s1 <= A\_4~2s1>,  
 '4~3m1 <= A\_4~3m1': <gurobi.Constr 4~3m1 <= A\_4~3m1>,  
 '4~4m1 <= A\_4~4m1': <gurobi.Constr 4~4m1 <= A\_4~4m1>,  
 '4~5b <= A\_4~5b': <gurobi.Constr 4~5b <= A\_4~5b>,  
 '4~5t1 <= A\_4~5t1': <gurobi.Constr 4~5t1 <= A\_4~5t1>,  
 '4~5t2 <= A\_4~5t2': <gurobi.Constr 4~5t2 <= A\_4~5t2>,  
 '4~5t3 <= A\_4~5t3': <gurobi.Constr 4~5t3 <= A\_4~5t3>,  
 '4~6b <= A\_4~6b': <gurobi.Constr 4~6b <= A\_4~6b>,  
 '4~6t1 <= A\_4~6t1': <gurobi.Constr 4~6t1 <= A\_4~6t1>,  
 '4~6t2 <= A\_4~6t2': <gurobi.Constr 4~6t2 <= A\_4~6t2>,  
 '4~6t3 <= A\_4~6t3': <gurobi.Constr 4~6t3 <= A\_4~6t3>,  
 '4~7b <= A\_4~7b': <gurobi.Constr 4~7b <= A\_4~7b>,  
 '4~7w1 <= A\_4~7w1': <gurobi.Constr 4~7w1 <= A\_4~7w1>,  
 '4~7w2 <= A\_4~7w2': <gurobi.Constr 4~7w2 <= A\_4~7w2>,  
 '4~8b <= A\_4~8b': <gurobi.Constr 4~8b <= A\_4~8b>,  
 '4~8w1 <= A\_4~8w1': <gurobi.Constr 4~8w1 <= A\_4~8w1>,  
 '4~8w2 <= A\_4~8w2': <gurobi.Constr 4~8w2 <= A\_4~8w2>,  
 '5~1s1 <= A\_5~1s1': <gurobi.Constr 5~1s1 <= A\_5~1s1>,  
 '5~2s1 <= A\_5~2s1': <gurobi.Constr 5~2s1 <= A\_5~2s1>,  
 '5~3m1 <= A\_5~3m1': <gurobi.Constr 5~3m1 <= A\_5~3m1>,  
 '5~4m1 <= A\_5~4m1': <gurobi.Constr 5~4m1 <= A\_5~4m1>,  
 '5~5b <= A\_5~5b': <gurobi.Constr 5~5b <= A\_5~5b>,  
 '5~5t1 <= A\_5~5t1': <gurobi.Constr 5~5t1 <= A\_5~5t1>,  
 '5~5t2 <= A\_5~5t2': <gurobi.Constr 5~5t2 <= A\_5~5t2>,  
 '5~5t3 <= A\_5~5t3': <gurobi.Constr 5~5t3 <= A\_5~5t3>,

```

'5~6b <= A_5~6b': <gurobi.Constr 5~6b <= A_5~6b>,
'5~6t1 <= A_5~6t1': <gurobi.Constr 5~6t1 <= A_5~6t1>,
'5~6t2 <= A_5~6t2': <gurobi.Constr 5~6t2 <= A_5~6t2>,
'5~6t3 <= A_5~6t3': <gurobi.Constr 5~6t3 <= A_5~6t3>,
'5~7b <= A_5~7b': <gurobi.Constr 5~7b <= A_5~7b>,
'5~7w1 <= A_5~7w1': <gurobi.Constr 5~7w1 <= A_5~7w1>,
'5~7w2 <= A_5~7w2': <gurobi.Constr 5~7w2 <= A_5~7w2>,
'5~8b <= A_5~8b': <gurobi.Constr 5~8b <= A_5~8b>,
'5~8w1 <= A_5~8w1': <gurobi.Constr 5~8w1 <= A_5~8w1>,
'5~8w2 <= A_5~8w2': <gurobi.Constr 5~8w2 <= A_5~8w2>,
'6~1s1 <= A_6~1s1': <gurobi.Constr 6~1s1 <= A_6~1s1>,
'6~2s1 <= A_6~2s1': <gurobi.Constr 6~2s1 <= A_6~2s1>,
'6~3m1 <= A_6~3m1': <gurobi.Constr 6~3m1 <= A_6~3m1>,
'6~4m1 <= A_6~4m1': <gurobi.Constr 6~4m1 <= A_6~4m1>,
'6~5b <= A_6~5b': <gurobi.Constr 6~5b <= A_6~5b>,
'6~5t1 <= A_6~5t1': <gurobi.Constr 6~5t1 <= A_6~5t1>,
'6~5t2 <= A_6~5t2': <gurobi.Constr 6~5t2 <= A_6~5t2>,
'6~5t3 <= A_6~5t3': <gurobi.Constr 6~5t3 <= A_6~5t3>,
'6~6b <= A_6~6b': <gurobi.Constr 6~6b <= A_6~6b>,
'6~6t1 <= A_6~6t1': <gurobi.Constr 6~6t1 <= A_6~6t1>,
'6~6t2 <= A_6~6t2': <gurobi.Constr 6~6t2 <= A_6~6t2>,
'6~6t3 <= A_6~6t3': <gurobi.Constr 6~6t3 <= A_6~6t3>,
'6~7b <= A_6~7b': <gurobi.Constr 6~7b <= A_6~7b>,
'6~7w1 <= A_6~7w1': <gurobi.Constr 6~7w1 <= A_6~7w1>,
'6~7w2 <= A_6~7w2': <gurobi.Constr 6~7w2 <= A_6~7w2>,
'6~8b <= A_6~8b': <gurobi.Constr 6~8b <= A_6~8b>,
'6~8w1 <= A_6~8w1': <gurobi.Constr 6~8w1 <= A_6~8w1>,
'6~8w2 <= A_6~8w2': <gurobi.Constr 6~8w2 <= A_6~8w2>,
'7~1s1 <= A_7~1s1': <gurobi.Constr 7~1s1 <= A_7~1s1>,
'7~2s1 <= A_7~2s1': <gurobi.Constr 7~2s1 <= A_7~2s1>,
'7~3m1 <= A_7~3m1': <gurobi.Constr 7~3m1 <= A_7~3m1>,
'7~4m1 <= A_7~4m1': <gurobi.Constr 7~4m1 <= A_7~4m1>,
'7~5b <= A_7~5b': <gurobi.Constr 7~5b <= A_7~5b>,
'7~5t1 <= A_7~5t1': <gurobi.Constr 7~5t1 <= A_7~5t1>,
'7~5t2 <= A_7~5t2': <gurobi.Constr 7~5t2 <= A_7~5t2>,
'7~5t3 <= A_7~5t3': <gurobi.Constr 7~5t3 <= A_7~5t3>,
'7~6b <= A_7~6b': <gurobi.Constr 7~6b <= A_7~6b>,
'7~6t1 <= A_7~6t1': <gurobi.Constr 7~6t1 <= A_7~6t1>,
'7~6t2 <= A_7~6t2': <gurobi.Constr 7~6t2 <= A_7~6t2>,
'7~6t3 <= A_7~6t3': <gurobi.Constr 7~6t3 <= A_7~6t3>,
'7~7b <= A_7~7b': <gurobi.Constr 7~7b <= A_7~7b>,
'7~7w1 <= A_7~7w1': <gurobi.Constr 7~7w1 <= A_7~7w1>,
'7~7w2 <= A_7~7w2': <gurobi.Constr 7~7w2 <= A_7~7w2>,
'7~8b <= A_7~8b': <gurobi.Constr 7~8b <= A_7~8b>,
'7~8w1 <= A_7~8w1': <gurobi.Constr 7~8w1 <= A_7~8w1>,
'7~8w2 <= A_7~8w2': <gurobi.Constr 7~8w2 <= A_7~8w2>,
'8~1s1 <= A_8~1s1': <gurobi.Constr 8~1s1 <= A_8~1s1>,
'8~2s1 <= A_8~2s1': <gurobi.Constr 8~2s1 <= A_8~2s1>,

```

```

'8~3m1 <= A_8~3m1': <gurobi.Constr 8~3m1 <= A_8~3m1>,
'8~4m1 <= A_8~4m1': <gurobi.Constr 8~4m1 <= A_8~4m1>,
'8~5b <= A_8~5b': <gurobi.Constr 8~5b <= A_8~5b>,
'8~5t1 <= A_8~5t1': <gurobi.Constr 8~5t1 <= A_8~5t1>,
'8~5t2 <= A_8~5t2': <gurobi.Constr 8~5t2 <= A_8~5t2>,
'8~5t3 <= A_8~5t3': <gurobi.Constr 8~5t3 <= A_8~5t3>,
'8~6b <= A_8~6b': <gurobi.Constr 8~6b <= A_8~6b>,
'8~6t1 <= A_8~6t1': <gurobi.Constr 8~6t1 <= A_8~6t1>,
'8~6t2 <= A_8~6t2': <gurobi.Constr 8~6t2 <= A_8~6t2>,
'8~6t3 <= A_8~6t3': <gurobi.Constr 8~6t3 <= A_8~6t3>,
'8~7b <= A_8~7b': <gurobi.Constr 8~7b <= A_8~7b>,
'8~7w1 <= A_8~7w1': <gurobi.Constr 8~7w1 <= A_8~7w1>,
'8~7w2 <= A_8~7w2': <gurobi.Constr 8~7w2 <= A_8~7w2>,
'8~8b <= A_8~8b': <gurobi.Constr 8~8b <= A_8~8b>,
'8~8w1 <= A_8~8w1': <gurobi.Constr 8~8w1 <= A_8~8w1>,
'8~8w2 <= A_8~8w2': <gurobi.Constr 8~8w2 <= A_8~8w2>,
'9~1s1 <= A_9~1s1': <gurobi.Constr 9~1s1 <= A_9~1s1>,
'9~2s1 <= A_9~2s1': <gurobi.Constr 9~2s1 <= A_9~2s1>,
'9~3m1 <= A_9~3m1': <gurobi.Constr 9~3m1 <= A_9~3m1>,
'9~4m1 <= A_9~4m1': <gurobi.Constr 9~4m1 <= A_9~4m1>,
'9~5b <= A_9~5b': <gurobi.Constr 9~5b <= A_9~5b>,
'9~5t1 <= A_9~5t1': <gurobi.Constr 9~5t1 <= A_9~5t1>,
'9~5t2 <= A_9~5t2': <gurobi.Constr 9~5t2 <= A_9~5t2>,
'9~5t3 <= A_9~5t3': <gurobi.Constr 9~5t3 <= A_9~5t3>,
'9~6b <= A_9~6b': <gurobi.Constr 9~6b <= A_9~6b>,
'9~6t1 <= A_9~6t1': <gurobi.Constr 9~6t1 <= A_9~6t1>,
'9~6t2 <= A_9~6t2': <gurobi.Constr 9~6t2 <= A_9~6t2>,
'9~6t3 <= A_9~6t3': <gurobi.Constr 9~6t3 <= A_9~6t3>,
'9~7b <= A_9~7b': <gurobi.Constr 9~7b <= A_9~7b>,
'9~7w1 <= A_9~7w1': <gurobi.Constr 9~7w1 <= A_9~7w1>,
'9~7w2 <= A_9~7w2': <gurobi.Constr 9~7w2 <= A_9~7w2>,
'9~8b <= A_9~8b': <gurobi.Constr 9~8b <= A_9~8b>,
'9~8w1 <= A_9~8w1': <gurobi.Constr 9~8w1 <= A_9~8w1>,
'9~8w2 <= A_9~8w2': <gurobi.Constr 9~8w2 <= A_9~8w2>,
'x_10~5(k) <= 1': <gurobi.Constr x_10~5(k) <= 1>,
'x_10~6(k) <= 1': <gurobi.Constr x_10~6(k) <= 1>,
'x_10~7(k) <= 1': <gurobi.Constr x_10~7(k) <= 1>,
'x_10~8(k) <= 1': <gurobi.Constr x_10~8(k) <= 1>,
'x_10~j <= 2': <gurobi.Constr x_10~j <= 2>,
'x_11~5(k) <= 1': <gurobi.Constr x_11~5(k) <= 1>,
'x_11~6(k) <= 1': <gurobi.Constr x_11~6(k) <= 1>,
'x_11~7(k) <= 1': <gurobi.Constr x_11~7(k) <= 1>,
'x_11~8(k) <= 1': <gurobi.Constr x_11~8(k) <= 1>,
'x_11~j <= 2': <gurobi.Constr x_11~j <= 2>,
'x_12~5(k) <= 1': <gurobi.Constr x_12~5(k) <= 1>,
'x_12~6(k) <= 1': <gurobi.Constr x_12~6(k) <= 1>,
'x_12~7(k) <= 1': <gurobi.Constr x_12~7(k) <= 1>,
'x_12~8(k) <= 1': <gurobi.Constr x_12~8(k) <= 1>,

```

```

'x_12~j <= 2': <gurobi.Constr x_12~j <= 2>,
'x_13~5(k) <= 1': <gurobi.Constr x_13~5(k) <= 1>,
'x_13~6(k) <= 1': <gurobi.Constr x_13~6(k) <= 1>,
'x_13~7(k) <= 1': <gurobi.Constr x_13~7(k) <= 1>,
'x_13~8(k) <= 1': <gurobi.Constr x_13~8(k) <= 1>,
'x_13~j <= 2': <gurobi.Constr x_13~j <= 2>,
'x_14~5(k) <= 1': <gurobi.Constr x_14~5(k) <= 1>,
'x_14~6(k) <= 1': <gurobi.Constr x_14~6(k) <= 1>,
'x_14~7(k) <= 1': <gurobi.Constr x_14~7(k) <= 1>,
'x_14~8(k) <= 1': <gurobi.Constr x_14~8(k) <= 1>,
'x_14~j <= 2': <gurobi.Constr x_14~j <= 2>,
'x_1~5(k) <= 1': <gurobi.Constr x_1~5(k) <= 1>,
'x_1~6(k) <= 1': <gurobi.Constr x_1~6(k) <= 1>,
'x_1~7(k) <= 1': <gurobi.Constr x_1~7(k) <= 1>,
'x_1~8(k) <= 1': <gurobi.Constr x_1~8(k) <= 1>,
'x_1~j <= 2': <gurobi.Constr x_1~j <= 2>,
'x_2~5(k) <= 1': <gurobi.Constr x_2~5(k) <= 1>,
'x_2~6(k) <= 1': <gurobi.Constr x_2~6(k) <= 1>,
'x_2~7(k) <= 1': <gurobi.Constr x_2~7(k) <= 1>,
'x_2~8(k) <= 1': <gurobi.Constr x_2~8(k) <= 1>,
'x_2~j <= 1': <gurobi.Constr x_2~j <= 1>,
'x_3~5(k) <= 1': <gurobi.Constr x_3~5(k) <= 1>,
'x_3~6(k) <= 1': <gurobi.Constr x_3~6(k) <= 1>,
'x_3~7(k) <= 1': <gurobi.Constr x_3~7(k) <= 1>,
'x_3~8(k) <= 1': <gurobi.Constr x_3~8(k) <= 1>,
'x_3~j <= 2': <gurobi.Constr x_3~j <= 2>,
'x_4~5(k) <= 1': <gurobi.Constr x_4~5(k) <= 1>,
'x_4~6(k) <= 1': <gurobi.Constr x_4~6(k) <= 1>,
'x_4~7(k) <= 1': <gurobi.Constr x_4~7(k) <= 1>,
'x_4~8(k) <= 1': <gurobi.Constr x_4~8(k) <= 1>,
'x_4~j <= 2': <gurobi.Constr x_4~j <= 2>,
'x_5~5(k) <= 1': <gurobi.Constr x_5~5(k) <= 1>,
'x_5~6(k) <= 1': <gurobi.Constr x_5~6(k) <= 1>,
'x_5~7(k) <= 1': <gurobi.Constr x_5~7(k) <= 1>,
'x_5~8(k) <= 1': <gurobi.Constr x_5~8(k) <= 1>,
'x_5~j <= 1': <gurobi.Constr x_5~j <= 1>,
'x_6~5(k) <= 1': <gurobi.Constr x_6~5(k) <= 1>,
'x_6~6(k) <= 1': <gurobi.Constr x_6~6(k) <= 1>,
'x_6~7(k) <= 1': <gurobi.Constr x_6~7(k) <= 1>,
'x_6~8(k) <= 1': <gurobi.Constr x_6~8(k) <= 1>,
'x_6~j <= 2': <gurobi.Constr x_6~j <= 2>,
'x_7~5(k) <= 1': <gurobi.Constr x_7~5(k) <= 1>,
'x_7~6(k) <= 1': <gurobi.Constr x_7~6(k) <= 1>,
'x_7~7(k) <= 1': <gurobi.Constr x_7~7(k) <= 1>,
'x_7~8(k) <= 1': <gurobi.Constr x_7~8(k) <= 1>,
'x_7~j <= 2': <gurobi.Constr x_7~j <= 2>,
'x_8~5(k) <= 1': <gurobi.Constr x_8~5(k) <= 1>,
'x_8~6(k) <= 1': <gurobi.Constr x_8~6(k) <= 1>,

```

```

'x_8~7(k) <= 1': <gurobi.Constr x_8~7(k) <= 1>,
'x_8~8(k) <= 1': <gurobi.Constr x_8~8(k) <= 1>,
'x_8~j <= 2': <gurobi.Constr x_8~j <= 2>,
'x_9~5(k) <= 1': <gurobi.Constr x_9~5(k) <= 1>,
'x_9~6(k) <= 1': <gurobi.Constr x_9~6(k) <= 1>,
'x_9~7(k) <= 1': <gurobi.Constr x_9~7(k) <= 1>,
'x_9~8(k) <= 1': <gurobi.Constr x_9~8(k) <= 1>,
'x_9~j <= 2': <gurobi.Constr x_9~j <= 2>,
'x_i~1s1': <gurobi.Constr x_i~1s1 >= 2>,
'x_i~2s1': <gurobi.Constr x_i~2s1 >= 2>,
'x_i~3m1': <gurobi.Constr x_i~3m1 >= 2>,
'x_i~4m1': <gurobi.Constr x_i~4m1 >= 2>,
'x_i~5b >= 1': <gurobi.Constr x_i~5b >= 1>,
'x_i~5t1': <gurobi.Constr x_i~5t1 >= 2>,
'x_i~5t2': <gurobi.Constr x_i~5t2 >= 1>,
'x_i~5t3': <gurobi.Constr x_i~5t3 >= 1>,
'x_i~6b >= 1': <gurobi.Constr x_i~6b >= 1>,
'x_i~6t1': <gurobi.Constr x_i~6t1 >= 2>,
'x_i~6t2': <gurobi.Constr x_i~6t2 >= 1>,
'x_i~6t3': <gurobi.Constr x_i~6t3 >= 1>,
'x_i~7b >= 1': <gurobi.Constr x_i~7b >= 1>,
'x_i~7w1': <gurobi.Constr x_i~7w1 >= 2>,
'x_i~7w2': <gurobi.Constr x_i~7w2 >= 1>,
'x_i~8b >= 1': <gurobi.Constr x_i~8b >= 1>,
'x_i~8w1': <gurobi.Constr x_i~8w1 >= 2>,
'x_i~8w2': <gurobi.Constr x_i~8w2 >= 1>}

```