Solving the Multiple Travelling Officers Problem to optimize the patrolling schedules in urban cities using Mathematical Programming Model

Presented by : Yoshita Banerjee and Monali Gupta
Submitted to : Dr. Prakash Kotecha
Institute : IIT Guwahati

Date : 28.11.2024

Table of Contents

Introduction

Urbanisation has amplified the need for efficient public service management, especially in the CBDs. Inefficiency in parking management can disrupt traffic and burden enforcement. Leveraging IoT- enabled parking sensors, this study tackles the multi-travelling officers problem using a Mixed Integer Linear Programming Model. This framework offers actionable insights for a smarter and more efficient urban parking management in smart city environments.

Objective

- Efficiently allocate officers to parking bays to monitor overstaying vehicles
- Balance travel costs by minimization of the distances travelled and maximization of fines collected

Challenges

- Multiple parking bays and time slots
- Limited officer availability and travel time
- Probabilistic overstay behaviour

Literature Review

Efficient urban parking management is a growing challenge in densely populated areas, particularly CBDs. Classical optimization problems like **Travelling Salesman Problem (TSP)** have been extended to address multi-agent systems such as **Multiple Travelling Officers Problem (MTOP).** Studies by ***Qin et al. (2020)*** have demonstrated the effectiveness of population-based algorithm in optimizing multi-agent routes. However, these approaches often lack real-time adaptability or inclusion of probabilistic factors.

The rise of **IoT enabled sensors** has made it possible to incorporate dynamic factors like overstaying probabilities into optimization framework. This study focuses on using the mathematical programming model to optimize patrol schedules by considering travel distances, overstaying probabilities, and officer working constraints.

The integration of **probabilistic models**, such are overstaying likelihoods has further enhanced the applicability of optimization methods in urban systems. Studies like ***Gendreau et al. (1994)*** introduced probabilistic considerations in routing problems, demonstrating improved outcomes in resource allocation.

Mixed Integer Linear Programming has emerged as a robust tool for such problems due to its ability to handle both continuous and binary variables. Recent studies, like ***Bektas et al. (2006),*** emphasize MILP's flexibility in addressing multi-agent scheduling problems. The proposed MTOP framework combines these elements, offering a novel approach to patrol scheduling that is both efficient and adaptable.

Model Formulation

- Objective Function :
  - *Single Objective* – To maximize the fines collected at the parking bays

$$Maximize\ Z = \left\{ \sum_{i=1}^{N} \sum_{k=1}^{M} \sum_{t=0}^{T} f_i * x_{ikt} - \sum_{k=1}^{M} y_k * sal \right\}$$

  - *Multi-Objective* – To find a trade-off between minimizing travel distance and maximizing fines collected

$$Z_o = \left\{ \alpha \sum_{i,j,k,t} d_{ij} * x_{ikt} - \beta \sum_{i,k,t} f_i * os_i * x_{ikt} - \sum_{k=1}^{M} y_k * sal \right\}$$

- Decision Variables
  - *Binary variable* for assignment of officer at the parking bay at a specific time x_{ikt}
  - *Binary variable* to determine whether an officer is active or not y_k

- Constraint Overview :
  - *Single Visit Constraint* – Ensures each parking bay is visited at most once across all officers and time periods

$$\sum_{k=1}^{M}\sum_{t=0}^{T} x_{ikt} \leq 1 \quad \forall\, i \in N$$

  - *Time Limit Constraint* – Ensures that the total travel time of each officer, on duty, is within their maximum working time

$$\sum_{i=1}^{N}\sum_{j=1}^{N} d_{ij} \,.\, \sum_{t=0}^{T} x_{ikt} \leq T * y(k) \qquad \forall\, k \in M$$

  - *Officer Availability Constraint* – Ensures that an officer can visit a parking bay only if they are actively employed

$$x_{ikt} \leq y_k \qquad \forall \;\; i \in N, k \in M, t \in T$$

  - *Minimum Probability Threshold* – Ensures that the parking bay is visited for which the overstay probability exceeds the threshold probability value

$$\sum_{k=1}^{M}\sum_{t=0}^{T} x_{ikt} * os_i \;\geq\; mp \,.\, \sum_{k=1}^{M}\sum_{t=0}^{T} x_{ikt} \quad \forall\;\, i \in N$$

MILP Formulation (Single Objective)

MILP Formulation

SINGLE – OBJECTIVE FUNCTION: $Maximize \left\{ \sum_{i=1}^{N} \sum_{k=1}^{M} \sum_{t=0}^{T} f_i * x_{ikt} - \sum_{k=1}^{M} y_k * sal \right\}$

Subject to the following constraints

1. $\textit{Single Visit Constraint} \sum_{k=1}^{M} \sum_{t=0}^{T} x_{ikt} \leq 1 \quad \forall\, i \in N$

2. $\textit{Time Limit Constraint} \sum_{i=1}^{N} \sum_{j=1}^{N} d_{ij} \,.\, \sum_{t=0}^{T} x_{ikt} \leq T * y(k) \quad \forall\, k \in M$

3. $\textit{Officer Availability Constraint} \quad x_{ikt} \leq y_k \quad \forall\; i \in N, k \in M, t \in T$

4. $\textit{Minimum Probability Threshold}$

$$\sum_{k=1}^{M} \sum_{t=0}^{T} x_{ikt} * os_i \;\geq\; mp \,.\, \sum_{k=1}^{M} \sum_{t=0}^{T} x_{ikt} \quad \forall\; i \in N$$

where,

N : number of parking bays (nodes)

M : number of officers

T : maximum working time for each officer

f_i : fine collected at parking bay i

x_{ikt} : whether parking bay i is visited by officer k at time t (Binary variable)

y_k : whether officer k is actively employed for patrolling (Binary variable)

sal : base salary paid to every officer

d_{ij} : distance between parking bay i and j

os_i : overstaying probability of car at parking bay i

mp : minimum overstay probability for visiting the parking bay

MILP Formulation (Multi-Objective)

MILP Formulation

MULTI – OBJECTIVE FUNCTION $Z_o = \left\{ \alpha \sum_{i,j,k,t} d_{ij} * x_{ikt} - \beta \sum_{i,k,t} f_i * os_i * x_{ikt} - \sum_{k=1}^{M} y_k * sal \right\}$

Subject to the following constraints

1. ***Single Visit Constraint*** $\sum_{k=1}^{M} \sum_{t=0}^{T} x_{ikt} \leq 1 \quad \forall\, i \in N$

2. ***Time Limit Constraint*** $\sum_{i=1}^{N} \sum_{j=1}^{N} d_{ij} \cdot \sum_{t=0}^{T} x_{ikt} \leq T * y(k) \quad \forall\, k \in M$

3. ***Officer Availability Constraint*** $x_{ikt} \leq y_k \quad \forall\; i \in N, k \in M, t \in T$

4. ***Minimum Probability Threshold***

$$\sum_{k=1}^{M} \sum_{t=0}^{T} x_{ikt} * os_i \;\geq\; mp \,.\, \sum_{k=1}^{M} \sum_{t=0}^{T} x_{ikt} \quad \forall\; i \in N$$

where,

N : number of parking bays (nodes)

M : number of officers

T : maximum working time for each officer

f_i : fine collected at parking bay i

x_{ikt} : whether parking bay i is visited by officer k at time t (Binary variable)

y_k : whether officer k is actively employed for patrolling (Binary variable)

sal : base salary paid to every officer

d_{ij} : distance between parking bay i and j

os_i : overstaying probability of car at parking bay i

mp : minimum overstay probability for visiting the parking bay

α : weight associated with the minimization of travel distance

β : weight associated with the maximization of fines collected

Data Inputs

- Parameters :
  - Distance between parking bays d_{ij}
  - Fines generated at each parking bay f_i and overstaying probabilities os_i
  - Base salary of the officers sal and Maximum working time of the officers T

- Snapshots of Data Input:

```
Sets
    i Parking bays /1*10/
    k Officers /1*3/
    t Time periods /1*10/;

Alias(i,j);

Table d(i,j) Distance between parking bays
         1      2      3      4      5      6      7      8      9     10
1        0    1.2    2.3    1.4    2.1    2.6    1.7    2.8    1.9    2.0
2      1.2      0    1.5    1.3    1.8    1.6    2.1    2.4    2.5    2.7
3      2.3    1.5      0    2.1    1.2    2.7    1.6    2.0    2.3    2.9
4      1.4    1.3    2.1      0    1.7    2.2    1.8    2.6    2.4    2.8
5      2.1    1.8    1.2    1.7      0    2.3    1.5    2.2    2.7    2.1
6      2.6    1.6    2.7    2.2    2.3      0    1.9    2.5    2.8    2.4
7      1.7    2.1    1.6    1.8    1.5    1.9      0    2.2    2.4    2.6
8      2.8    2.4    2.0    2.6    2.2    2.5    2.2      0    2.3    2.1
9      1.9    2.5    2.3    2.4    2.7    2.8    2.4    2.3      0    2.0
10     2.0    2.7    2.9    2.8    2.1    2.4    2.6    2.1    2.0      0;

Parameters
    f(i)  Fine for parking violation at each bay /1 50, 2 40, 3 60, 4 70, 5 30, 6 80, 7 50, 8 90, 9 30, 10 40/
    os(i) Probability of car overstaying at each parking bay /1 0.6, 2 0.5, 3 0.7, 4 0.4, 5 0.8, 6 0.3, 7 0.9, 8 0.2, 9 0.7, 10 0.6/
    max_time(t) Maximum time for officer tours /1 600, 2 600, 3 600, 4 600, 5 600, 6 600, 7 600, 8 600, 9 600, 10 600/
    mp Minimum probability threshold for visiting the parking bay /0.4/
    sal Base salary of each officer /100/;
```

Solution Approach

- *Mixed Integer Linear Programming (MILP)* is used as the Optimization Technique to model this version of multiple travelling officers problem. GAMS (General Algebraic Modelling System) and IBM ILOG CPLEX are the tools used to achieve the solution.

```
Binary Variables
    x(i,k,t) whether officer k visits bay i at time t
    y(k) whether officer k is on patrol duty;

Variables
    z Objective function (maximize fines - salary);

Equations
    obj Objective function
    E1(i) Each parking bay visited at most once
    E2(k) Total travel time for each officer within the limit
    E3(i,k,t) Officer can only visit a node if active
    E4(i) Parking bay is only visited if overstay probability is sufficient;

* Objective: Maximize fines and minimize officer costs
obj..z =e= sum(i,sum(k,sum(t, f(i) * x(i,k,t)))) - sum(k, y(k) * sal);

* Constraint 1: Each parking bay visited at most once
E1(i)..sum(k,sum(t, x(i,k,t))) =l= 1;

* Constraint 2: Total travel time for each officer must not exceed max_time
E2(k)..sum(i,sum(j, d(i,j))* sum(t, x(i,k,t))) =l= sum(t,max_time(t) * y(k));

* Constraint 3: Officer can only visit only if he is active
E3(i,k,t)..x(i,k,t) =l= y(k);

* Constraint 4: Parking bay visited only if probability threshold is met
E4(i)..sum(k,sum(t, x(i,k,t)*os(i))) =g= mp * sum(k,sum(t, x(i,k,t)));

Model mtop /all/;
Solve mtop using mip maximizing z;
Display x.l,y.l,z.l;
```

Implementation in GAMS

```
// Objective Function
maximize
  sum(i in Nodes, k in Officers, t in Time) fine[i] * x[i][k][t] -
  sum(k in Officers) base_salary * y[k];

// Constraints

// Each parking bay is visited at most once across all officers and time periods
subject to {
  forall(i in Nodes)
    sum(k in Officers, t in Time) x[i][k][t] <= 1;

  // Travel time for each officer must not exceed working duration
  forall(k in Officers)
    sum(i in Nodes, j in Nodes, t in Time)
      distance[i][j] * x[i][k][t] <= working_duration * y[k];

  // An officer can only visit a node if they are active
  forall(i in Nodes, k in Officers, t in Time)
    x[i][k][t] <= y[k];

  // Nodes are visited only if the overstaying probability threshold is met
  forall(i in Nodes)
    sum(k in Officers, t in Time) x[i][k][t]*overstay[i] >= min_prob * sum(k in Officers, t in Time) x[i][k][t];
}
```

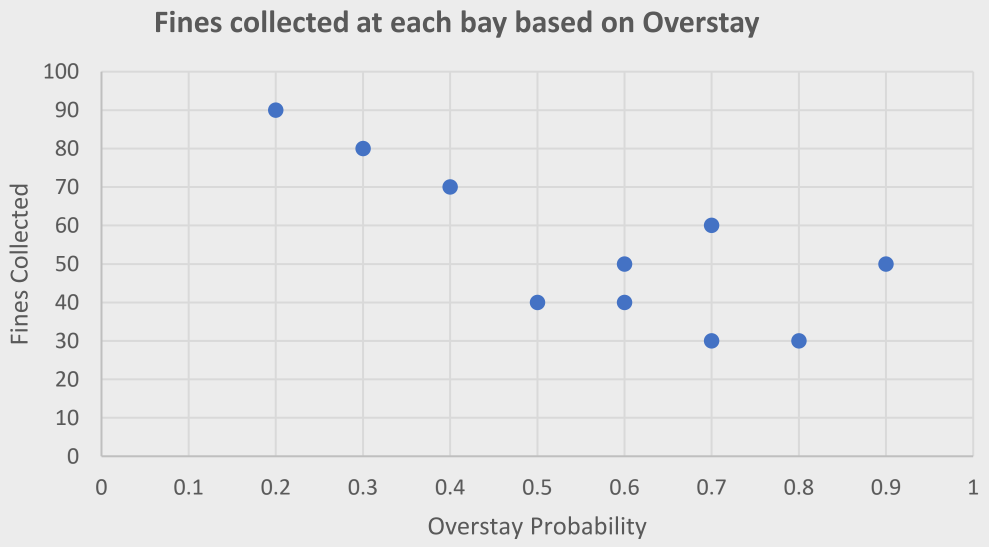
Implementation in IBM ILOG CPLEX

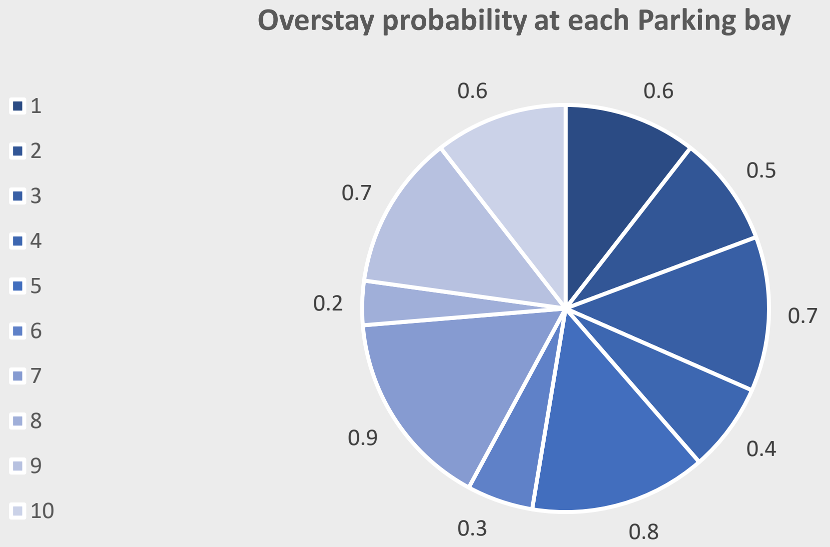
- *The problem has been solved using a single objective function and a multi-objective function*
  - In the single objective approach, the **total fines collected** from parking violations is **maximized**
  - In the multi-objective approach, a single combined objective function is formulated which **minimizes** the **travel time** and **maximizes** the **fines collected** , thus handling trade-offs with weights α and β

Results and Discussions

We have, for simplicity, assumed violations in 10 parking bays, each having its own overstay probability, and 3 patrolling officers across 10 time periods to maximize the collection of fines.

Fines generated should be correlated with higher overstay probabilities as the likelihood of violations increase. This dataset shows some alignment with this trend particularly for high probability cases. However, inconsistencies in this trend may be due to several operational factors. Bay with 0.3 probability has a fine of 80 whereas one with 0.5 probability has a fine of 40. This could imply that these bays have higher traffic volume leading to more violations despite lower probabilities. The fines collected are not only dependent on overstay probability but also on officer availability, time constraints, and travel distance between the bays.





Result Comparison (GAMS)

Single Objective

Maximizing the fines collected

Proven optimal Solution
MIP Solution: 270.000000 (26 iterations, 0 nodes)
Final Solve: 270.000000 (0 iterations)

Best Possible: 270.000000
Absolute gap: 0.000000
Relative gap: 0.000000

```
* Objective: Maximize fines and minimize officer costs
obj..z =e= sum(i,sum(k,sum(t, f(i) * x(i,k,t)))) - sum(k, y(k) * sal);


Proven optimal solution
MIP Solution:           270.000000    (26 iterations, 0 nodes)
Final Solve:            270.000000    (0 iterations)

Best possible:          270.000000
Absolute gap:             0.000000
Relative gap:             0.000000
```

Multi Objective

Minimizing the travel time and maximizing the fines collected

Proven optimal Solution
MIP Solution: 300.300000 (0 iterations, 0 nodes)
Final Solve: 300.300000 (0 iterations)

Best Possible: 300.300000
Absolute gap: 0.000000
Relative gap: 0.000000

```
* Objective: Minimize travel time and maximize profit
obj ..
   z =e= alpha * sum((i,j,k,t), distance(i,j) * x(i,k,t))
         - beta * sum((i,k,t), fine(i) * overstay_prob(i) * x(i,k,t))
         - sum(k, y(k) * officer cost);
Proven optimal solution
MIP Solution:           300.300000    (0 iterations, 0 nodes)
Final Solve:            300.300000    (0 iterations)

Best possible:          300.300000
Absolute gap:             0.000000
Relative gap:             0.000000
```

Result Comparison (IBM ILOG CPLEX)

Single Objective

Maximizing the fines collected

Solution (optimal) with objective : 270

Quality Incumbent solution

MILP objective : 2.7000000000e+02

MILP solution norm |x| : 9.00000e+00 1.00000e+00

```
// Objective Function
maximize sum(i in Nodes, k in Officers, t in Time) f[i] * x[i][k][t]
       - sum(k in Officers) sal * y[k];

subject to{

    //Single visit constraint
    forall(i in Nodes){
        sum(k in Officers, t in Time) x[i][k][t] <= 1;}

    //Time limit constraint
    forall(k in Officers){
        sum(i in Nodes, j in Nodes, t in Time) (d[i][j] * x[i][k][t]) <= max_time * y[k];}

    //Officer availability constraint
    forall(i in Nodes, k in Officers, t in Time){
        x[i][k][t] <= y[k];}

     //Minimum probability threshold
     forall(i in Nodes){
        sum(k in Officers, t in Time) (x[i][k][t] * os[i]) >= mp * sum(k in Officers, t in Time) x[i][k][t];}

}
```

Problem browser | Variables | Breakpoints

Solution with objective 270

Name	Value
Data (12)	
d	[[0 1.2 2.3 1.4 2.1 2.6 1.7 2....
f	[50 40 60 70 30 80 50 90 3...
M	3
max_time	600
mp	0.4
N	10
Nodes	1..10
Officers	1..3
os	[0.6 0.5 0.7 0.4 0.8 0.3 0.9 ...
sal	100
T	10
Time	1..10
Decision variables (2)	
x	[[[1 0 0 0 0 0 0 0 0 0] [0 0 ...
y	[1 0 0]

Multi Objective

Minimizing the travel time and maximizing the fines collected

Solution (optimal) with objective : 300.3

Quality Incumbent solution

MILP objective : 3.0030000000e+02

MILP solution norm |x| : 4.00000e+00 1.00000e+00

Problem browser | Variables | Breakpoints

Solution with objective 300.3

Name	Value
Data (14)	
alpha	1
beta	1
d	[[0 1.2 2.3 1.4 2.1 2.6 1.7 2....
f	[50 40 60 70 30 80 50 90 3...
M	3
max_time	600
mp	0.4
N	10
Nodes	1..10
Officers	1..3
os	[0.6 0.5 0.7 0.4 0.8 0.3 0.9 ...
sal	100
T	10
Time	1..10
Decision variables (2)	
x	[[[0 0 0 0 0 0 0 0 0 0] [0 0 ...
y	[1 1 1]

```
// Objective Function
maximize alpha * sum(i in Nodes, j in Nodes, k in Officers, t in Time)(d[i][j] * x[i][k][t])
         - beta * (sum(i in Nodes, k in Officers, t in Time)(f[i] * os[i] * x[i][k][t])
         - sum(k in Officers) (y[k] * sal));

subject to{

    //Single visit constraint
    forall(i in Nodes){
        sum(k in Officers, t in Time) x[i][k][t] <= 1;}

    //Time limit constraint
    forall(k in Officers){
        sum(i in Nodes, j in Nodes, t in Time) (d[i][j] * x[i][k][t]) <= max_time * y[k];}

    //Officer availability constraint
    forall(i in Nodes, k in Officers, t in Time){
        x[i][k][t] <= y[k];}

     //Minimum probability threshold
     forall(i in Nodes){
        sum(k in Officers, t in Time) (x[i][k][t] * os[i]) >= mp * sum(k in Officers, t in Time) x[i][k][t];}

}
```

Future Scope

1. Incorporating Dynamic Probabilities (Real-Time Updates):

Current Limitation: The current model assumes static probabilities of overstaying for each parking bay, which may not accurately reflect real-time conditions.

Future Improvement:

- Integrate IoT-enabled parking sensors or surveillance data to update overstaying probabilities dynamically.
- Real-time updates allow the model to adapt to evolving conditions, such as peak traffic hours or special events that increase parking demand.

2. Multi-Day Planning with Officer Fatigue Considerations:

Current Limitation: The model optimizes patrol schedules for a single day, assuming officers are always at peak efficiency.

Future Improvement:

- Extend the model to account for multi-day planning, incorporating officer fatigue and rest periods.
- Introduce constraints for maximum working hours per officer across multiple days, ensuring fair workload distribution.
- Consider shifts and overlapping schedules to maintain continuous coverage.

Thank You