Avoid the Overflow of Disaster Victims around the Railway Station
- via Adaptive Management of Passengers’ Routes and Train Schedule -

Chiba University
Hino Yoko, Arai Sachiy0

2015.3.25 RailTokyo 2015
Outline

1. Background
2. Objective
3. Approach
   - Definition
   - Preliminary Experiment
   - Passengers’ Route Assignment
   - Modify Train Schedule
4. Conclusion
## Background

2011.3.11 the Great East Japan Earthquake (Japan)

<table>
<thead>
<tr>
<th>Caused by Nature</th>
<th>Caused by Human</th>
</tr>
</thead>
<tbody>
<tr>
<td>unavailable Major Train Line</td>
<td>resumed partial line</td>
</tr>
<tr>
<td>cannot reach home</td>
<td>rush to the station</td>
</tr>
<tr>
<td><strong>emerge Stranded Commuter cannot avoid</strong></td>
<td>overflow the station’s capacity</td>
</tr>
<tr>
<td></td>
<td>cause additional re-stop</td>
</tr>
<tr>
<td></td>
<td><strong>worsen Stranded Commuter</strong></td>
</tr>
</tbody>
</table>
Objective

Partial Resumption → Capacity Overflow → Additional Re-Stop → Reach Home Early

negative circle

impossible to ensure the safety

impede the recovery
Objective

- **Hardware measures**
  - expand the station ← carry a cost
Objective

Software measures

Change

1. Passengers’ Route
2. Train Schedule

Partial Resumption → Capacity Overflow

Additional Re-Stop → Reach Home Early
Definition

- calculate the number of passengers in the station

$$n_i = H_{i}^{in} + H_{i}^{pass} + H_{i}^{transfer} + H_{i}^{out}$$

**Station** $s_i$

- come in as Origin
- getting out at Destination

**Passed the prior train because of full**

**Waiting to transfer**
Definition

- calculate the number of passengers in the station

Station $s_i$

$$n_i = H_i^{in} + H_i^{pass} + H_i^{transfer} + H_i^{out} \geq \text{capacity}$$

Maximum

Capacity Overflow

overflowed passenger
Preliminary Experiment

- simulate passengers’ translation in normal train service
- estimate the number of passengers in each time
- make the maximum value as capacity

Railway Network

Station
- Train line
Cost of the link : Required time to go through
Preliminary Experiment

- Passengers’ Settings
  - move through the shortest route
  - follow actual data of Shibuya Station’s inflow

Set up approximate curve by least square method

\[
y = -0.0123x^3 + 0.037x^2 + 177.74x + 8822.2
\]
Result: Preliminary Experiment

\[<\text{Station } s_1>\]

Maximum value
\[= 437\]

decide Maximum value as the capacity

\[<\text{Station } s_2>\]

\[c_2 = 381\]

\[<\text{Station } s_3>\]

\[c_3 = 1263\]

\[<\text{Station } s_4>\]

\[c_4 = 1263\]
Result : Preliminary Experiment

\(< \text{Station } s_1 > \) \quad c_1 = 437

Maximum value
= 437

\(< \text{Station } s_2 > \) \quad c_2 = 381

\(< \text{Station } s_3 > \) \quad c_3 = 1263

\(< \text{Station } s_4 > \) \quad c_4 = 1263

decide Maximum value as the capacity
1. Passengers’ Route Assignment

- Objective:
  change the Passengers’ route not to overflow the capacity

- formulate as Constraint Satisfaction Problem
  - Constraint

\[ n_i(t) \leq c_i \]

( \( t : \) time step [min.] )

satisfy the constraints while waiting

- use Backtracking Algorithm to search the route
- adopt Dijkstra method to decide searching order by the shortest route
1. Passengers’ Route Assignment

![Diagram showing route assignment]

The diagram illustrates the assignment of passengers' routes, with the origin at 1 and the destinations at various points. The shortest route, marked as $n_1(t) \leq c_1$, is highlighted.
1. Passengers’ Route Assignment

\[ n_1(t) \leq c_1 \]
1. Passengers’ Route Assignment

- ② secondary shortest route
- Origin
- Backtrack
- $n_1(t) \leq c_1$
- Destination
- Repeat the process until you reach destination
- make possible to disperse passengers’ route
Experiment: Passengers’ Route Assignment

Settings
- equal to Preliminary Experiment
- adopt capacity $c_i$ obtained from Preliminary Experiment
- stop all trains in the first 90[min.]

Proposal:
change the route by backtracking & avoid capacity overflow

Comparison (passenger’s natural behavior):
go on only the shortest route & avoid capacity overflow
wait until they are allowed to enter the station
Result: number of passengers in the station

![Graphs showing passenger numbers at stations s1, s2, s3, and s4 over time.]

- control overflow than passengers’ natural behavior
Result: overflowed passengers upon delay time

- decrease overflow than passengers’ natural behavior in any delay time

**Comparison**
- lower overflowed passengers by 21.04%

**Proposal**
- appear difference after 40 minute
2. Modify Train Schedule

- Objective:
  change the **Train schedule** to reduce the overflowed passengers
- formulate as **Combinatorial Optimization Problem**
- solve by means of **Genetic Algorithm**

- Objective function

\[
\text{min. fitness} = \sum_{i=1}^{4} (n_i(t) - c_i)
\]

unless \((n_i(t) - c_i) \leq 0\)

- minimize the overflowed passengers

\[
\]
2. Modify Train Schedule

- define individual
  - one of the elements of Train Schedule
  - length of stoppage time of each train at each Station

- Stoppage time

<table>
<thead>
<tr>
<th></th>
<th>Station $s_1$</th>
<th>Station $s_2$</th>
<th>Station $s_3$</th>
<th>Station $s_4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train $v_1$</td>
<td>$x_1$</td>
<td>$x_2$</td>
<td>$x_3$</td>
<td>$x_4$</td>
</tr>
<tr>
<td>Train $v_2$</td>
<td>$x_5$</td>
<td>$x_6$</td>
<td>$x_7$</td>
<td>$x_8$</td>
</tr>
<tr>
<td>Train $v_3$</td>
<td></td>
<td></td>
<td></td>
<td>$x_{10}$</td>
</tr>
<tr>
<td>Train $v_4$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- search the optimal combination of variables
2. Modify Train Schedule

Algorithm

- Initial schedule
- simulate passengers’ translation
- calculate fitness
- selection, crossover, mutation
- satisfy end condition
- YES
- NO
- simulate passengers’ translation
- calculate fitness
- End
2. Modify Train Schedule

- **Proposal**
  Passengers’ Backtracking Route
- **Comparison (natural behavior)**
  wait on the Shortest Route

- Algorithm
  - Initial schedule
  - simulate passengers’ translation
  - calculate fitness
  - selection, crossover, mutation
  - satisfy end condition
  - create Train Schedule with Individual which minimize the fitness

① Proposal
Passengers’ Backtracking Route
② Comparison (natural behavior)
wait on the Shortest Route
Result: sum of overflowed passengers

Change Train Schedule with Proposal Route

Decreased by 8.83%

Change Train Schedule with Comparison Route

Decreased by 24.38%

The both overflowed passengers decreased
- have still lower by 20.94% than comparison after modifying the Train Schedule
Conclusion

- focused on the software measure to avoid capacity overflow
- proposed to change Passengers’ Route not to overflow
  - solve as Constraint Satisfaction Problem
  - control and decrease the overflow by 21.04%
- proposed to change Train Schedule which reduce overflow
  - solve by means of Genetic Algorithm
  - decreased 8.83% of the overflow after modifying the Train Schedule
- decreased 29.04% overflowed passengers by changing both Passengers’ route & Train schedule
- changing only software are also effective to prevent overflow
Thank you for your kind attention.
Result: total trip time upon delay time

- Shorten total trip time than passengers’ natural behavior in any delay time

- 11.83% shorten in delay time
### Time Complexity

- **Station number is** \( n \)

\[
\text{Number of judging} \times (n - 1) \times n^2
\]

- **Worst-case execution time**

\[
O(8(n - 1)n^2) = O(n^3)
\]
Station number is \( n \)

\[ n \times (n-1) \times n^2 \]

※ if the train line (degree) increase, the route increase & directly affect the number of judging
**time complexity**

- Station number is $n$

<table>
<thead>
<tr>
<th>$n$</th>
<th>degree</th>
<th>Number of judge</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>9</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>31</td>
</tr>
<tr>
<td>6</td>
<td>5</td>
<td>129</td>
</tr>
<tr>
<td>7</td>
<td>6</td>
<td>651</td>
</tr>
<tr>
<td>8</td>
<td>7</td>
<td>3913</td>
</tr>
<tr>
<td>9</td>
<td>8</td>
<td>27399</td>
</tr>
<tr>
<td>10</td>
<td>9</td>
<td>219201</td>
</tr>
<tr>
<td>:</td>
<td>:</td>
<td>:</td>
</tr>
<tr>
<td>$n$</td>
<td>$n-1$</td>
<td></td>
</tr>
</tbody>
</table>

Estimate worst judging time

With the complete graph

- Only 15 train lines in Shinjuku station

Combinatorial Explosion