Aalto-yliopisto
Perustieteiden korkeakoulu

Dynamic Programming Approaches for the Bus Rapid Transit Investment Problem (Presentation of Results)

Bilqays Ayoub

17.06.2024

Supervisor: Philine Schiewe

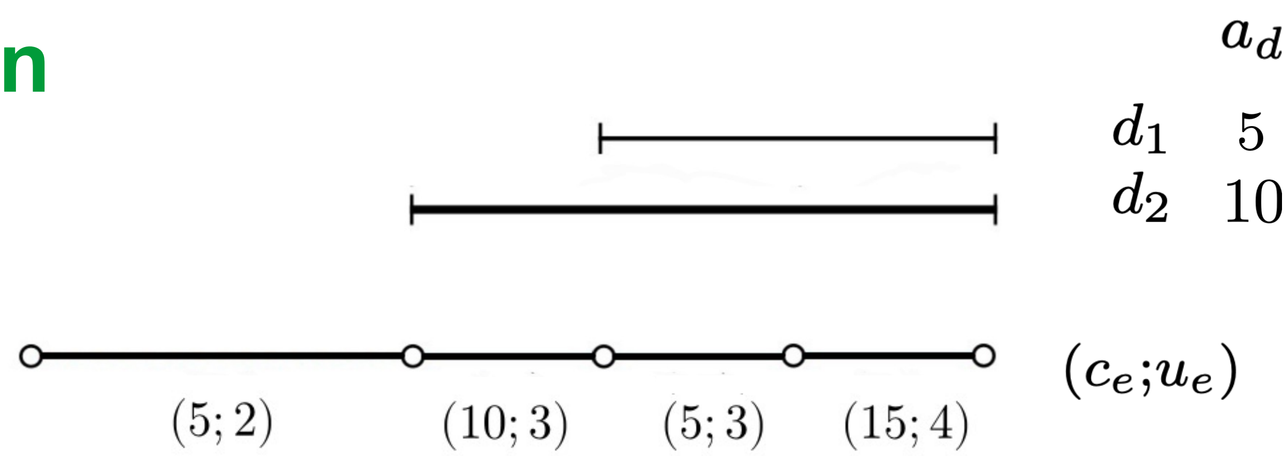
Instructor: Philine Schiewe

Aalto-yliopisto
Perustieteiden korkeakoulu

Background

- Bus Rapid Transit systems offer more reliable transportation with lower investment costs providing fast and reliable bus services
- BRT investment problem optimizes the selected segments for upgrade within budget constraints while maximizing the number of passengers
- Presented in a paper by Hoogervorst [2023].

Given



- A single line, algorithm selects the upgraded segments
- Nodes represent bus stations
- Edges represent segments between stations

- Each segment has cost and infrastructure improvement
- Given OD pairs with passenger demand

IP Formulation

- Objective function $\max \sum_{e \in E} \tilde{u}_e x_e$
- Budget constraints $\text{s.t.} \sum_{e \in E_m} c_e x_e \leq v$
- Variable $x_e \in \{0, 1\}, \ \forall e \in E$

Assumptions

- Linear passenger response

$$\tilde{u}_e := u_e \cdot \sum_{\substack{d \in D: \\ e \in W_d}} \frac{a_d}{\sum_{e' \in W_d} u_{e'}}$$

- Single municipality

$$|M| = 1$$

- Budget is ranged from zero to maximum

- Infrastructure improvements set to $u_e = 1$

Dynamic Programming

- Optimizes the most cost-effective upgrades within budget constraints for BRT segments
- Can be solved in pseudo-polynomial time with efficient solutions
- Pareto-front can be extracted from the dynamic program by identifying optimal solutions

0-1 Knapsack Algorithm

- Knapsack algorithm to optimize the selected segments to be upgraded withitn a given budget:

Algorithm 1 Dynamic Programming Algorithm for 0-1 Knapsack Problem

```
Input: Instance: list of (w),(v),(C)
Output: Set of items to maximize the total value within the given capacity
n ← length(w)
Initialize K[0..n][0..C] with zeros
for i ← 1 to n do
    for c ← 1 to C do
        if w[i − 1] ≤ c then
            K[i][c] ← max(K[i − 1][c], v[i − 1] + K[i − 1][c − w[i − 1]])
        else
            K[i][c] ← K[i − 1][c]
        end if
    end for
end for
return K[n][C]
```

Pareto Front

- Efficient solutions represent the optimal set of upgrades that can be made within the budget constraints.
- Non-dominated points are the best objective values of efficient solutions.
- Pareto front is the set of non-dominated solutions

Example Instance

- 10 stations
- cost pattern UNIT
- demand pattern EVEN

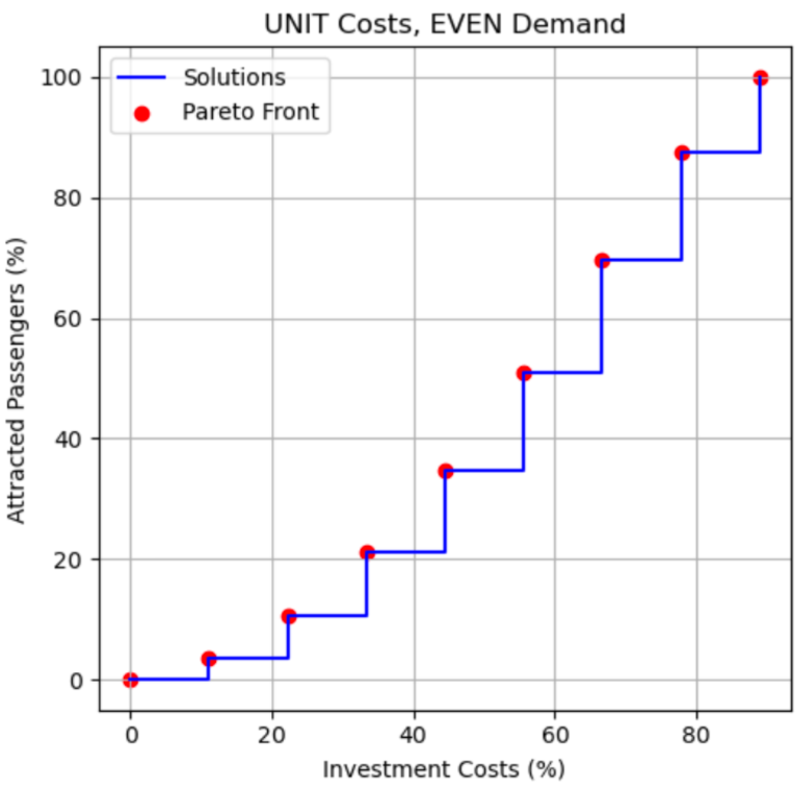


Table 1: Running time

All Points	# Points	Per Point
0.0047	9	0.0005

Running time

- The runtime of the dynamic program for the different data sets on average:

Table 2: Running time and number of Pareto points averaged over artificial instances sharing the same cost pattern

Stops	Cost Pattern	All Points	# Points	Per Point
10	UNIT	0.0037	9.0	0.0004
	MIDDLE	0.0045	18.0	0.0003
	ENDS	0.0044	14.7	0.00037
25	UNIT	0.1095	24.0	0.00457
	MIDDLE	0.2193	48.0	0.00527
	ENDS	0.2117	58.3	0.0041

Results

10 stops

25 stops

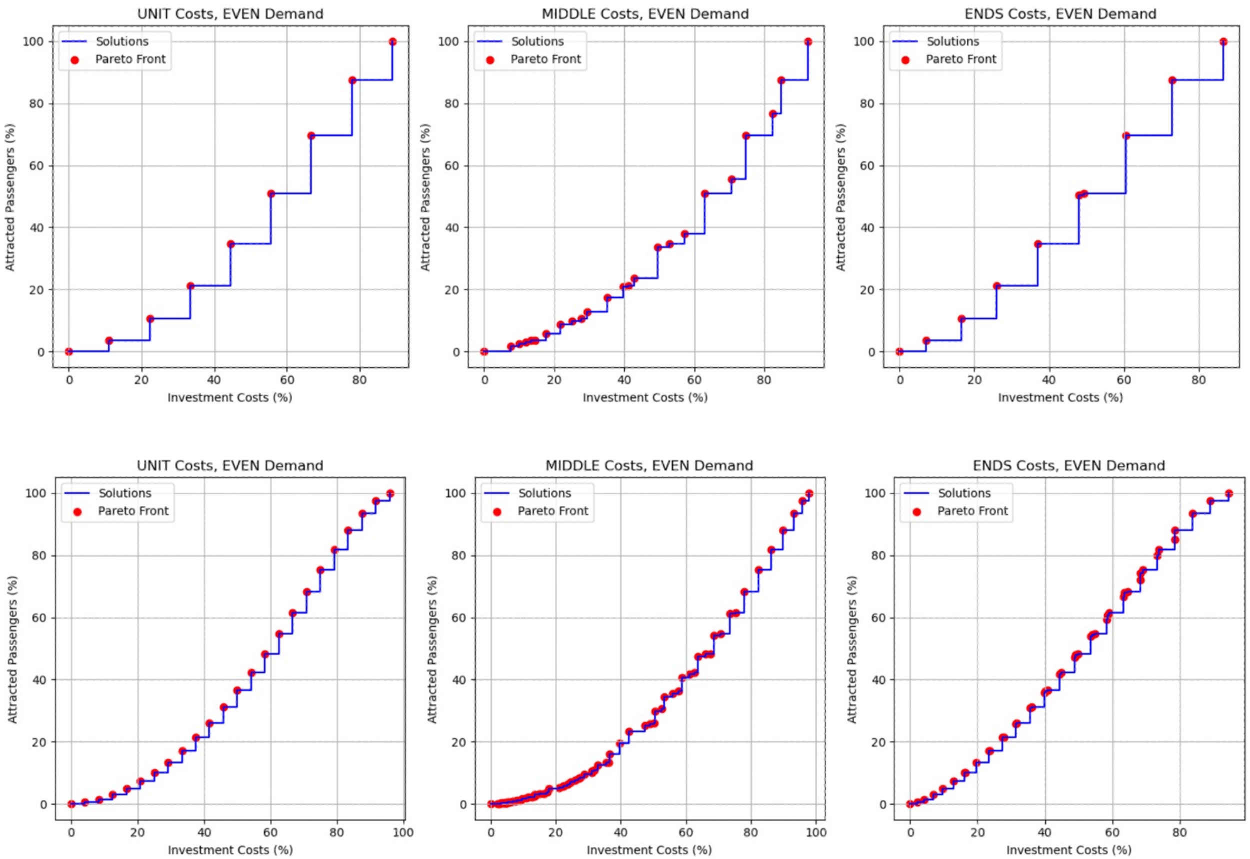


Figure 1: Non- dominated points of BRT representing Linear passenger response for artificial instances across all cost patters while passenger demand is even.

References

- Rowan Hoogervorst, Evelien van der Hurk, Philine Schiewe, Anita Schöbel, and Reena Urban, ”The Bus Rapid Transit Investment Problem”, 2023, arXiv preprint arXiv:2308.16104

- Hoogervorst, R., Hurk, E. V. D., Schiewe, P., Schöbel, A., & Urban, R. (2022). The Edge Investment Problem: Upgrading Transit Line Segments with Multiple Investing Parties. In Proceedings of 22nd Symposium on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS 2022)