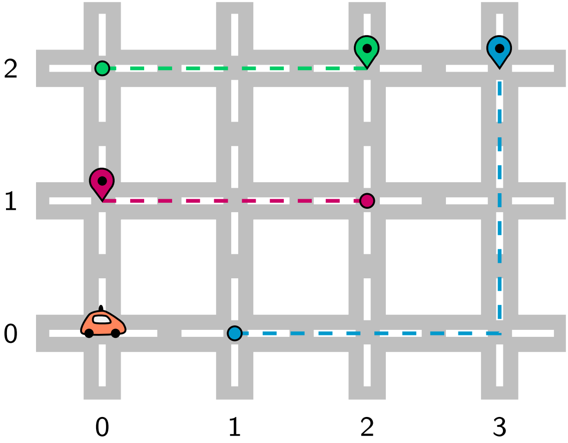
Google Hash Code

Self-driving rides

Hash Code 2018, Online Qualification Round

Problem Statement

Problem representation



Problem Statement

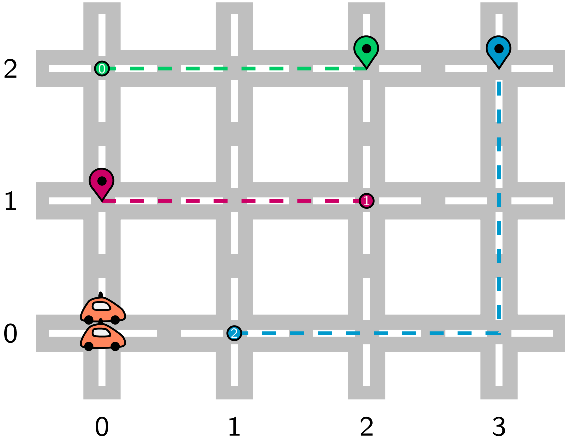
Problem representation

- R, C number of rows and columns in the grid
- F size of the fleet (number of vehicles)
- N number of rides
  - $\forall r \in [1, N], s_r, f_r$: starting and ending points of the ride
  - $\forall r \in [1, N], e_r, l_r$: earliest start time and latest end time of the ride
- B bonus for rides that start on time
- T time horizon
- Score for a ride: distance of the ride plus a potential bonus if it starts on time

  Objective: Maximize the score for all completed rides

Example

Example



$T = 15$
$B = 2$

$e_0 = 2$
$l_0 = 14$

$e_1 = 4$
$l_1 = 14$

$e_2 = 0$
$l_2 = 14$

Example

Example

- Grid with 3 rows and 4 columns
- 2 vehicles
- 3 rides
  - $s_0 = (0, 2), f_0 = (2, 2), e_0 = 2, l_0 = 14$
  - $s_1 = (2, 1), f_1 = (0, 1), e_1 = 4, l_1 = 14$
  - $s_2 = (1, 0), f_2 = (3, 2), e_2 = 0, l_2 = 14$
- Bonus: 2
- Time horizon: 15 time steps

Problem Statement

Variables?

- The rides assigned to the vehicles
  - $\forall v \in [0, F-1], L_v$: the list of rides assigned to vehicle v

Greedy Algorithm

Principle

- At each step, a choice is made that seems the best at that moment
- It constructs a solution step by step
  - Without revisiting decisions
  - By making the best choice at each step
  - Hoping to achieve an optimal global result
- Greedy Approach
  - No guarantee of optimality for some problems (greedy heuristic)
  - Low-cost (compared to exhaustive enumeration)
  - Intuitive choice

Greedy Algorithm

Example

- 2 vehicles, 3 rides
  - $s_2 = (1,0), f_2 = (3,2), e_2 = 0, l_2 = 14, d_2 = 4$
  - $s_0 = (0,2), f_0 = (2,2), e_0 = 2, l_0 = 14, d_0 = 2$
  - $s_1 = (2,1), f_1 = (0,1), e_1 = 4, l_1 = 14, d_1 = 2$

Objective: Maximize the score for all completed rides

- Sort the rides by decreasing distance
- Go through the rides and try to assign each one to a vehicle to maximize the score (distance + bonus)

Greedy Algorithm

Example

- 2 vehicles, 3 rides
  - $s_2 = (1,0), f_2 = (3,2), e_2 = 0, l_2 = 14, d_2 = 4$
  - $s_0 = (0,2), f_0 = (2,2), e_0 = 2, l_0 = 14, d_0 = 2$
  - $s_1 = (2,1), f_1 = (0,1), e_1 = 4, l_1 = 14, d_1 = 2$

- $L_0 = [2,1]$ $\quad t_0 = 9, p_0 = (0,1)$
- $L_1 = [0]$ $\quad t_1 = 4, p_1 = (2,2)$

- score $= 10$

Greedy Algorithm

Example

- 2 vehicles, 3 rides
  - $s_2 = (1,0), f_2 = (3,2), e_2 = 0, l_2 = 14, d_2 = 4$
  - $s_0 = (0,2), f_0 = (2,2), e_0 = 2, l_0 = 14, d_0 = 2$
  - $s_1 = (2,1), f_1 = (0,1), e_1 = 4, l_1 = 14, d_1 = 2$

Improvements

You can change the strategy

1. Sort rides by decreasing distance
2. Sort rides by bonus potential
3. Use a combination of both strategies