Real-Time Airport Bus Scheduling Problem

Project for Algorithm: Analysis and Theory

Xiaofeng Gao
Department of Computer Science,
Shanghai Jiao Tong University, Shanghai, China

Abstract. This project introduces a new bus-booking platform. It includes the brief introduction, background and advantages of this platform. Finally, it lists all the requirements and rules for each group. Please read this document carefully and complete the corresponding tasks.
Keywords: Bus-Booking Platform, Order Dispatch

1 Bus Scheduling Problem

The rapid deployment of transportation in recent years speeds up the travel between different cities and increases the travel frequency, which brings more traffic in return. To cater to the increasing travel demands and relieve the traffic pressure, some novel transportation concepts have been proposed and put into the wide application, e.g., online car-hailing. However, some locations with high travel demands are not fully satisfied yet. According to the official statistics from Shanghai Traffic Management Bureau (STMB), the number of the passengers arrived at Shanghai Pudong International Airport is about 18,000 during midnight period (23:00-02:00). However, the local public transport capacity can only provide pick-up service for around 8,200 passengers, including 7,000 passengers by taxi and online car-hailing, and 1,200 passengers by airport shuttle buses (less than 7% of the total arrived passengers), leaving a large number of passengers waiting at the airport for a long duration.

In order to provide a higher-quality service, STMB intends to develop a new service APP named bus-booking platform. Unlike car-hailing platforms, a bus-booking platform can dispatch several orders to one bus together with a specially planned routing cycle, based on real-time requirements from its customers. Accordingly, passengers could be served quickly with much lower price. In all, combining the benefits of bus service and car-hailing, this new APP provides economical service with real-time flexible route planning and order dispatch, which contributes to a better user experience.

2 Tasks and Requirements

In the early stage of this service, it is complicated to compute arbitrary passenger destinations to form a route. Instead, STMB decides to select some hot destinations (denoted by V), such that passengers can select one of them as their destination station. You are required to design the kernel of this bus-booking platform for STMB, with the following sub-problems.

2.1 Task 1

Suppose that there are n passengers and m buses in the airport, where the destination station of passenger i is d_i and the capacity of bus j is K_j ($n \leq \sum_j K_j$). Let the travel cost from station v_s to station v_t be $c_{s,t}$ (the graph is a complete graph). Now you need to design the travel lines for each bus so as to minimize the total travel cost of all the buses. Note that the travel line is a cycle that starts from the airport and also ends at the airport.

1. Please prove that this problem is NP-Hard.
2. Please design an approximation algorithm for this problem.

2.2 Task 2

Suppose that the travel lines of all the m buses have been determined and cannot be changed.

1. (Offline) Now n passengers arrive at the airport and each of them has a destination station. You need to decide to whether to accept this passenger, and allocate this passenger to one bus if he/her has been accepted. Please design a polynomial-time algorithm to maximize the number of the accepted passengers (this problem belongs to P class). Note that the capacity of bus j is still K_j.

2. (Online) Suppose that the passenger arrives one by one. Once a passenger arrive, you need to decide whether to accept this passenger, allocate this passenger to one bus if he/her has been accepted. Note that once you have made the decision you can not change it any more. Similarly, your objective is to maximize the number of the accepted passengers.
   (a) Please prove that there is no algorithm which can always produce the optimal solution for this online problem.
   (b) The greedy algorithm can be used for this problem, which is presented in Algorithm 1. Please prove that this greedy algorithm is an $\frac{1}{2}$-competitive algorithm for this problem. In addition, construct a tight example where this algorithm exactly achieves an $\frac{1}{2}$ competitive ratio. In other words, the analysis of the competitive ratio for this greedy algorithm is tight. (In Algorithm 1, denote by S_j the set of stations that bus j passes through and d_i the destination station of passenger i)
   (c) Can you design another online algorithm with a better competitive ratio?

Algorithm 1: Most Remaining Seats First (MRSF) Algorithm

Input: $m, (S_1, \ldots, S_m), (K_1, \ldots, K_m)$
Output: The passengers in each bus $B_1, \ldots, B_m$
for $j \leftarrow 1 \rightarrow m$ **do**
  $B_j \leftarrow \emptyset$;
  $R_j \leftarrow K_j$.
while *passenger* (i, d_i) *arrives* **do**
  $j^* \leftarrow \arg\max_{j:d_i \in S_j} R_j$;
  if $R_{j^*} \leq 0$ **then**
    reject passenger (i, d_i);
  else
    $B_{j^*} \leftarrow B_{j^*} \cup \{i\}$;
    $R_{j^*} \leftarrow R_{j^*} - 1$;
return $B_1, \ldots, B_m$;

2.3 Task 3

Suppose that all the buses and all the passengers arrives over time (Not only the passengers). Similarly, the travel line of each bus has been determined once it arrives at the airport. Hence, each bus has three attributes, t_j (the arrive time), K_j (the capacity) and S_j (the set of passed stations). Some passengers may want to take the same bus with their friends to their destinations (the same destinations). Suppose that one order has four attributes, t_i (the arrived time), c_i (the number of passengers), d_i (the destination station) and p_i (the priority). Your objective is to maximize the sum of the priority of the accepted orders.

1. Please prove that the offline version of this problem is NP-Hard (The offline version means that all the buses and the orders are given in advance).
2. Please design an online algorithm for this problem.

2.4 Performance Evaluation:

it includes the following requirements:

1. For each question, generate samples with more than 300 orders to illustrate your design. The attachment contains 30 destinations with their latitude and longitude.(The destinations are in Chinese, you can just consider its number in the first column)
2. Test the efficiency of your design by simulations.

3 Report Requirements

You need to submit a report for this project, with the following requirements:

1. Your report should have the title, the author names, IDs, email addresses, the page header, the page numbers, figure for your simulations, tables for discussions and comparisons, with the corresponding figure titles and table titles.
2. Your report is English only, with a clear structure, divided by sections, and may contain organizational architecture like itemizations, definitions, or theorems and proofs.
3. Please include reference section and acknowledgement section. You may also include your feelings, suggestion, and comments in the acknowledgement section.
4. Please define your variables clearly. If needed, a symbol table is strongly recommended to help readers catch your design.
5. Please also include your latex source and simulation codes upon submission.