The School Bus Routing Problem

Lab sessions

1 Motivation

In many countries, students that live within certain minimum and maximum distances of their school are entitled to free transportation to and from school. This transportation is usually organized by a local company, which uses school buses that drive fixed routes. At each school term, the transportation company determines which routes its buses will follow, and where they should stop so each student has at least one stop that he/she can reach. To this end, a set of potential stops is determined first in such a way that each student lives within a predefined distance from at least one stop. Routes are then determined for the school buses so that all students are picked up at a stop they can reach, while assuring that the capacity of the buses is not exceeded. This problem is called the *School Bus Routing Problem (SBRP)*. In the Flemish region of Belgium, for example, the local transportation company is faced with problem instances where up to 3000 students have to be picked up and brought to seven different schools.

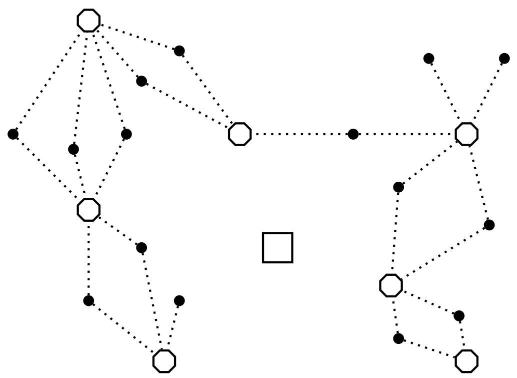
2 General description

In most vehicle routing problems, a set of stops (customers) is given and routes need to be determined such that each stop (customer) is visited once. The SBRP, however, involves a set of potential stops that might or might not be visited. Thus, determining the set of stops to actually visit is a part of the problem formulation. The objective of the SBRP is to simultaneously

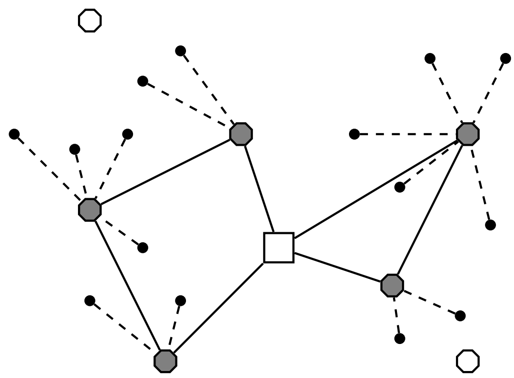
1. Determine the set of stops to visit,
2. Define, for each student, at which stop he/she will be picked up,
3. Determine the routes in which the selected stops will be visited,

such that the total distance travelled by the buses is minimized. Figure 2.1 shows a toy example of a SBRP instance and a solution to it. In this figure, the square represents the school, the octagons represent the bus stops and the black dots represent the students. In the SBRP instance shown in Figure 2.1(a), students are connected (by means of dotted edges) to the bus stops at which they can be picked up. In other words, students are connected to the potential stops that they can be assigned to. In the SBRP solution shown in Figure 2.1(b), each student is connected (by means of dashed edges) to the actual bus stop that he/she is assigned to. Additionally, Figure 2.1(b) also shows the stops that are visited (the gray octagons) and the routes that two buses should follow (solid lines).

The three sub-problems previously mentioned are strongly interrelated. Two of these sub-problems, 1 and 3, have a direct impact on the total distance travelled by the buses. Sub-problem 2, on the other hand, only affects the objective function indirectly since it merely determines whether a combination of routes and



(a) SBRP instance



(b) SBRP solution

Figure 2.1: Toy example of a SBRP instance and a solution to it

selected bus stops is feasible or not. For this reason, it seems intuitively logical to treat sub-problems 1 and 3 on the same decision level.

When the stops have been selected and the routes to visit these stops have been determined, it is necessary to assign students to stops. When a student can be assigned to multiple stops along the same route, the allocation of this student to a particular stop is arbitrary. This is not the case if a student can be assigned to multiple stops in different routes. All students that can be assigned to multiple routes need to be distributed over those routes in such a way that the capacity of the buses is not exceeded. Compared to traditional vehicle routing problems, the option to assign students to different stops offers the possibility to incur potential savings; however, at the same time, it introduces an extra decision level that makes the problem much more difficult to solve.

In the context of a matheuristic approach, the SBRP can be decomposed into a master problem and a sub-problem. The master problem is a school bus routing problem with bus stop selection, the objective of which is to minimize the total travelled distance. Once the stops have been selected and the routes have been fixed, a sub-problem remains of allocating students to stops in such a way that the capacity of the buses is not exceeded. This sub-problem is a constraint satisfaction problem in that it does not have an objective function. The existence of a feasible solution to this sub-problem, however, implies that the corresponding solution to the master problem is valid. The sub-problem can be easily formulated as an integer programming problem and solved by means of an exact solver.

3 LAB SESSIONS

For the lab sessions, students will be divided in teams of 3 or 4 members and will be coached by an experienced researcher. Each team will develop an optimization algorithm to solve a different variant of the SBRP. Each group can choose a different optimization approach to tackle the variant selected. One team could, for example, use LocalSolver in combination with a metaheuristic algorithm. Another group might make use of FICO-express embedded in a more elaborate matheuristic framework. The optimization algorithms will be implemented in a web-service platform in order to make them easily available.

3.1 SBRP variants

Each team will be able to choose one among these SBRP variants:

- **Basic SBRP**: the basic version of the problem.

- **SBRP with varying number of students (demands)**: in this version, each student has an additional quantity associated to it. This quantity can represent, for example, the number of students in that location (for instance, two brothers that attend the same school). In a more general context, this variant can be considered to model a routing problem for retail companies that have multiple collection points in a city. These companies, in order to reduce their operating costs, can request their customers to pick up their orders at one of the closest collection points. Companies therefore need to determine the collection points to visit, assign the customers to the points selected, and define the routes in which the selected points should be visited.
- **SBRP with an alternative objective function**: in this version, the goal is not to minimize the total distance travelled but to optimize a different objective function; for example, to minimize the average travel time.
- **SBRP with a time limit:** in this version, the routes cannot take longer than a fixed amount of time (since students should arrive at the school before the classes start).
- **SBRP with heterogeneous fleet of buses**: in this version, there are multiple types of buses (with different characteristics). One could consider, for example, that some students have special requirements that need to be taken into account. Therefore, these students can only be picked up by specific types of buses.
- **SBRP with multiple schools:** in this version, we consider multiple schools and assume that all the buses are parked in a single depot. In this sense, the buses need to pick up the students, drop them off at their school, and go back to the depot. This means that the last location visited by every bus must be a school, and that all the students in a bus must attend the same school.

3.2 Baseline library

The teams will be provided with an initial library that contains code to model/tackle the basic SBRP. This library provides the data structures required to model the different elements of the problem (stops, students, routes, solution, ...) and to perform the operations typically found in optimization algorithms (calculate the objective function, insert and remove stops from a route, ...). This library is coded in C++ and follows an object oriented programming approach (OOP). If you are not familiar with C++ or the OOP approach, we strongly recommend you to spend some time on getting acquainted with them. If a team prefers to, they can write their own code from scratch. However, this will probably slow down the progress and the learning process during the lab sessions.

3.3 Specific tasks

The main goal of the lab sessions is to learn how to implement a matheuristic algorithm and how to make it accessible via a web-service platform. For that, we will aim at accomplishing the following specific tasks:

1. Get familiar with the web-service platform and learn how to make your algorithm available
2. Get acquainted with the baseline library
3. Modify the baseline library in order to handle the SBRP variant selected
4. Implement a metaheuristic framework to solve the master problem described in Section 2 (i.e., school bus routing problem with bus stop selection)
5. Solve the sub-problem described in Section 2 (i.e., assign students to stops) using one of the exact solvers

3.4 Exact solver

Since one of the specific tasks described in Section 3.3 involves using an exact solver, we strongly encourage you to get familiar with, at least, one of them. During the PhD school, there will be one session dedicated to LocalSolver and one session dedicated to the FICO XPress Optimization suite. However, previous experience with one of the solvers (getting acquainted with the API for C++, creating and solving a simple model, ...) will be very beneficial.