CME307/MS&E 311 Project Proposal
Santa's Workshop Tour Challenge

Oscar Aguilar
Louis Gautier

April 2023

We intend to work on solving the Santa's Workshop Tour Kaggle challenge. This optimization challenge, posted on Kaggle in 2019, consists in solving a constrained optimization problem. The task is to match customers to days so that they can visit Santa's workshop, given preferences for each day, capacity constraints, and externalities linked to the number of customers visiting the workshop each day. This problem is associated with one large instance available on Kaggle containing the preference lists for 5,000 different families. As this problem is NP-hard, solving it on such a large instance will be challenging and require us to use advanced strategies and heuristics.

1 Problem formulation

The problem posed is to schedule 5,000 different families for tours of Santa's workshop in the 100 days before Christmas. Each day, between 125 and 300 families must visit the workshop, and each family has provided a ranked list of dates that they would prefer to attend. However, Santa's big heart, combined with overcrowding costs and a very complicated North Pole tax code, impose heavy costs when families don't get their preferences and when the workshop gets crowded. Since these costs come out of next year's Christmas budget, it is our duly sworn duty to find a policy that minimizes the total cost. Let's define the mathematical notation and formulation of this optimization problem:

- The policy $\mathcal{S}$ must assign each family i for $i = 1, \ldots, 5000$ with $n_i \in \mathcal{N}$ family members one of the $d = 100, \ldots, 1$ days before Christmas to visit. Let $\mathcal{S}_d$ be the set of families which attend on day d.
- This policy $\mathcal{S}$ is subject to a capacity constraint on the total number of visitors each day: $125 \leq N_d \leq 300$, where $N_d = \sum_{i \in \mathcal{S}_d} n_i$.
- Santa gives consolation gifts to families according to their assigned day relative to their preferences. Let $p_{i,d}$ be the preference that family i gives to day d.

- Let $G = (0, 50, 59, 109, 209, 218, 318, 336, 436, 735, 934, 934, \ldots, 934)$, where G_j is the value of the consolation gift given to each member of a family who gets their j^{th} choice of visit day, for $1 \leq j \leq 100$.

- Thus, the total cost of consolation gifts under a given policy $\mathcal{S}$ is:

$$C_{\mathcal{S}} = \sum_{d=100}^{1} \sum_{i \in \mathcal{S}_d} n_i \cdot G_{p_{i,d}} \tag{1}$$

- Finally, Santa's accountants have developed an empirical equation to represent an accounting penalty which depends on the number of visitors each day. The total accounting cost under policy $\mathcal{S}$ is:

$$A_{\mathcal{S}} = \sum_{d=100}^{1} \frac{(N_d - 125)}{400} N_d^{\left(\frac{1}{2} + \frac{|N_d - N_{d+1}|}{50}\right)} \tag{2}$$

- *Note*: in both cost equations we are counting backwards the days until Christmas. Additionally, for day $d = 100$ we set the initial condition $N_{d+1} = N_{101} = N_{100}$

2 Potential project outline

In this project, we are planning to use the following strategies, which are not definitive as we might have other ideas or refine some of these objectives as we will work on tackling this challenge:

- Formulate this problem as a Mixed Integer Linear Program, that will probably be too large to be solved using a solver like Gurobi in a reasonable amount of time.

- Find lower bounds to the objective values by relaxing some constraints and obtaining smaller and easier-to-solve Mixed Integer Linear Programs.

- Try to solve the problem using an improved model formulation, or a sequence or combination of easier MIPs.

- Try to solve the problem using local search thanks to heuristics or meta-heuristics. This should allow us to explore the search space more efficiently in order to find good solutions.

- Try to solve the problem using a combination of easier MIPs (to find a good starting point) and local search based on heuristics.

We will finally try to implement our solutions in Julia and take advantage of the *JuMP* library.