KnIT Anaemia – Survey Route Optimization

# **Introduction**:

Nationally representative data is collected through surveys. Complex surveying methodologies involving structured sampling techniques are used to collect the data. This is very time consuming, with surveys typically lasting months, incurring large costs. Over here, we develop a tool that can be used by survey supervisors to optimize the surveying strategy in terms of scheduling village visits and choosing sites for base of operations. The tool will aid in choosing an optimal survey schedule for the next day, and can be rerun at the end of each day to update the schedule with latest data.

# Techniques used:

The problem has been treated as a typical Operations Research problem. This problem is a Linear Programming problem(LPP) under the routing spectrum. The optimization engine was implemented using google OR tools. The steps for solving an OR problem involve: Formulation of problem, design of algorithm, implementation.

# Formulation:

There are several factors involved in the formulation of such a problem. The objective is to minimize the total cost incurred. This will, in most cases, be directly proportional to the total time spent for the survey. Hence, the objective of this optimization will be to minimize the time taken to complete the survey. Time is chiefly spent on answering the questionnaire and the inter village travel times for the interviewers. We determine an ideal route that minimizes time spent in travelling and interviewing. Our objective function is

CodeCogsEqn (1)

where, r denotes a vehicle, i and j refer to villages, the binary decision variable Xrij indicates if the vehicle, r, traverses between villages i and j in an optimal solution, Tij - Denotes the time taken to travel from i to j. p is the total no of vehicles and n is the total no of villages.

The constraints are:

1.Every village is visited

CodeCogsEqn (1)

2.The maximum time spent by each group of interviewers outside the central facility (maximum working hour limit).

CodeCogsEqn (1)

Where p is the total no of vehicles, n is the total no of villages and Q is the maximum working time for an interviewer, Dj denotes the time requirement of a particular village.

There are several assumptions involved in the first version implemented,

It has been assumed that the time taken for interview is constant for all households.

The engine gives an optimal route depending on the current times required at each village.

# Design of Algorithm

Mixed linear programming has been used to solve the problem. The engine uses Google OR tools, an opensource, free to use library for commercial and research purposes. The routing library of the Google OR has been used to formulate the problem so as to ensure that the same engine can be easily modified to include other features. <https://developers.google.com/reference/constraint_solver/routing/>.

# Implementation

1. Initial data taken from survey supervisor includes:

* the average time required for the interview
* no of households in all villages
* buffer time for intra household travel - for each village
* time matrix of time between villages

2. Formulation:

* total time = no of households \* (time per households + buffer for intra traversal).
* run the algorithm and display the initial set of routes to be followed - only for day 1.

3. Iterative approach:

* collect the feedback at the end of the day
* modify the demand vector and add time add time constraints to villages if any
* run the algorithm again on the modified data to get the new set of routes
* iterate till all the demands are met

# Executable Description

The formulated problem was divided into 2 versions depending on the various constraints involved in each of them.

1. All the villages could be visited at any time of the day
2. Time constraints were imposed on the villages

The executable created uses village 0 as depot by default.

The executable requires 2 files in the same directory for data: Travel\_times.csv and Demands.csv.

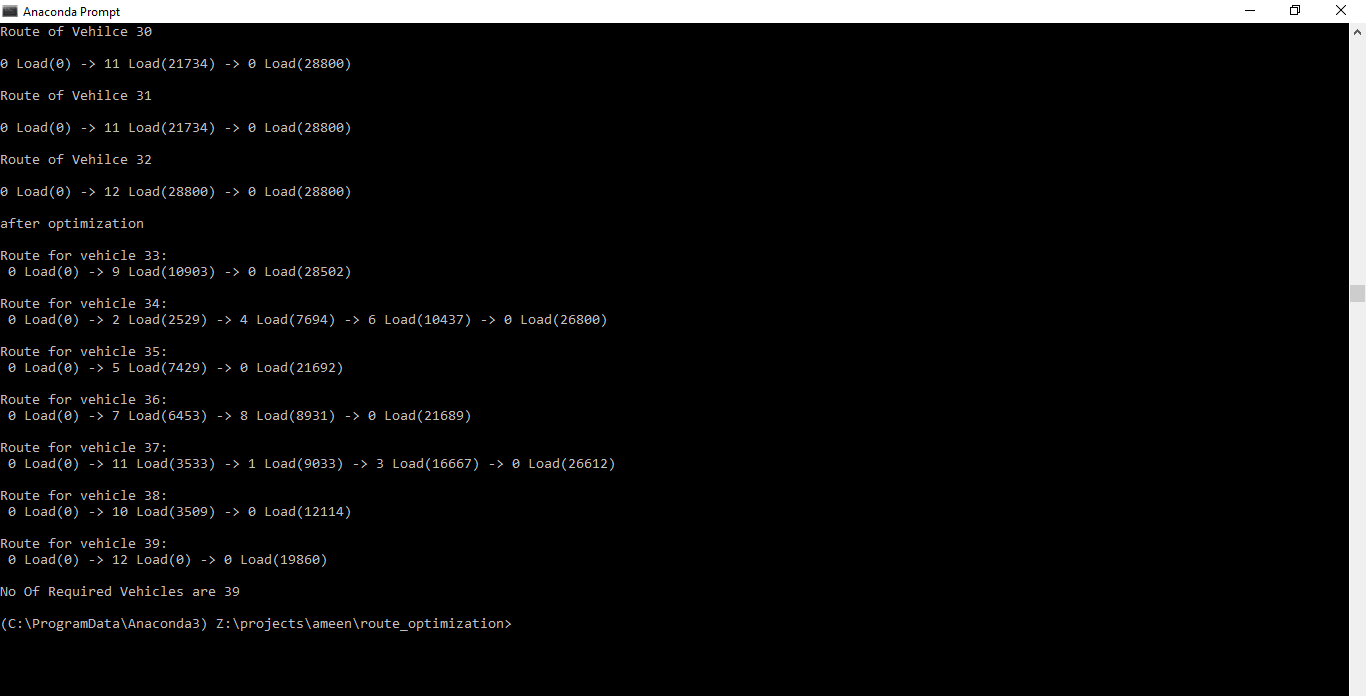
Demands.csv has the time requirement of every village encoded in seconds and the column that has the requirement has to specified in the program. The executable takes by default the column 5 to contain the demands.

Travel\_times.csv contains a matrix in the csv format that represent the travel times required to travel from one village to another. Its defined as d[i][j] = time taken to travel from village i to village j.

There could also be a case of having specific time constraints for a particular village.

In such cases, the time-constraints should also be explicitly specified in a separate file in the same directory named Time\_constraints.csv with first column containing the constraint time intervals of villages in column 2 of csv in the same order as of Demands.csv.

The output of the executable will be something will be of the form



In addition, an API for the software is also being developed that will allow it to be plugged into any front end (website, mobile app etc.) and offer more flexibility in terms of input and constraints.

## Assumptions Involved

The current executable has several assumptions involved.

1. The interviewers leave the central facility in groups of same size and the time demand of each village is expressed in terms of the time required by this group.
2. The central facility has a time\_demand of 0. The demand of central facility is met outside the optimization by vehicles that return early or on the last day.
3. Intra household traversal time is a constant for a given village, given by the supervisor at the beginning of the survey

## Known Issues that may impede a possible optimum solution

1. Inherent variability in time taken per interviewer

2. Accuracy of distance and time estimates provided by google

3. Variability induced in distance and time estimates due to natural phenomena

\* This may cause a permanent change in schedules, rendering the solution implemented as non-optimal.

4. Accuracy of GPS coordinates used in geocoding the villages

# Validation

The performance was validated using DFS data set and was giving an output much better than the real time taken to complete the entire survey. The no if interviewers involved in the survey was 10. The survey went on to take 31 days for its completion.

The engine suggested routes can theoretically complete the survey in less than half this time.