Leonora Frangu, Fabio James Greenwood, Giada Palma, Benedetta Pasqualetto

**Discrete Optimisation Project Documentation**

**Planning and multimodal transportation**

**Index**

[Info 1](#_Toc116117727)

[Documentation Notes 1](#_Toc116117728)

[Assumptions 1](#_Toc116117729)

[Indexes 2](#_Toc116117730)

[Constants 2](#_Toc116117731)

[Independent Variables 2](#_Toc116117732)

[Semi-Dependant Variables 3](#_Toc116117733)

[Dependent Variables 3](#_Toc116117734)

[Constraints 3](#_Toc116117735)

[Basic Conservation of Flow 3](#_Toc116117736)

[Task Timing 3](#_Toc116117737)

[Bus Travel Constraints 4](#_Toc116117738)

[Bike Capacity Constraints 4](#_Toc116117739)

[Unfinished Tasks Penalties 5](#_Toc116117740)

# Requirements

|  |  |  |
| --- | --- | --- |
| #1  OBJECTIVE FUNCTIONS | Determine the tours that all people have to do in order to perform their tasks, by considering the different modes of transportation available and all the constraints, in order to minimize the total travelled time by also taking into account the overall fitness score. |  |
| #2  PEOPLE | It is a new day in Verona, and a set N := {1, . . . ,N} of people is ready to perform all their errands as efficient as they can be.  Each person n ∈ N, who starts from a known point Hn (e.g., their home) with a given amount of money Wn, has a certain set Tn of tasks to do. | All Done |
| #3  TASKS | Each task t may cost ct €, has to start during the related time window [at , bt ], and generally lasts τt minutes.  Some special tasks t∗ ∈ T∗ n ⊆ Tn, either t∗ has to start in the related time window, or it can be started anytime. However, in this latter case, the duration of t∗ will be increased by a certain amount of minutes.  Each task t occurs in a different place p. | All done |
| #4  PLACES | Let P := {1, . . . , P} be the set of all places of the tasks of all people in N (e.g., shops, squares, bars, facilities, etc.).  Each place p ∈ P is described by a latitude value latp and a longitude value lonp. | Not implemented |
| #4.1  MAX NUMBER OF PEOPLE | Also, a place p cannot be visited by more than Np people at the same time. | Not implemented |
| #5  MODES OF TRANSPORTATION | To reach the places associated to their tasks Tn, each person n is allowed to either walk, or to ride a bike offered by the bike-sharing company in Verona, or also to take a bus, according to the distribution of the bike-sharing stations and the bus lines in the city. | NA |
| #6  TRANSPORTATION MODES CHANGES | Let M:= {Walking , Cycling, Bus} be the set of modes of transportation.  Each person n ∈ N can change their mode of transportation at most Kn times, and, in the end, n must return to their starting point Hn. | Red not implemented |
| #6.1  PENALTY | A penalty Qt must be paid for a task t ∈ Tn not performed. | None Implemented |
| #7  BIKES AND BIKE STATIONS | Let B := {1, . . . ,B} be the set of bike stations.  For each bike station b ∈ B, the number αb of bikes available at the beginning of the day is known, as well as the number βb of free spots where to leave a bike (of course, if all spots are available, the maximum capacity of a bike station b is Nb := αb + βb).  Similarly, to any place each bike station b is associated with a latitude value latb and a longitude value lonb. | Capacity not checked |
| #7.1  BIKE COSTS | Riding a bike costs Cbike €/min. | Not implemented |
| #8  BUSES AND LINES | Let L := {1, . . . , L} of lines is available in the city.  Each line ℓ is composed by a set of stops Sℓ := {Dℓ, . . . , Sℓ}, and starts from a certain depot Dℓ at time δ. | All Implemented |
| #8.1  RIDES FREQUENCY | Then, there is a ride every δ minutes. | All Implemented |
| #8.2  MAX NUMBER OF PEOPLE | The maximum number of people who can be on a bus at the same time is Nℓ. | Not Implemented |
| #9  STOPS AND  LINES CONSTRAINTS | Each stop s ∈ Sℓ is associated with a latitude value lats and a longitude value lons .  No two lines Sℓ1 and Sℓ2 have two identical consecutive stops.  Anyway, some lines Sℓ1 and Sℓ2 may intersect at some stops (i.e., Sℓ1 ∩ Sℓ2 ̸= ∅). | NA |
| #10  BUS SELECTION AND CHANGE | Every time that a person n chooses the bus to change their mode of transportation, they pay Cbus €.  If n just chooses to change the bus line in a stop, without changing the mode of transportation, they do not have to pay another ride. | Half Implemented |
| #11  NETWORK | Let H be the set of all starting points Hn, for every n ∈ N.  Let S be the set of all bus stops, for every line ℓ ∈ L.  Let G := (V,A) the directed weighted multi-graph such that V := H ∪ P ∪ B ∪ S is the set of all points in the city.  Let A := {(i , j ,m) | i , j ∈ V,m ∈ M} be the set of all arcs connecting point i to point j by using mode of transportation m. | Unsure about how to impliement |
| #12  TRAVELLING TIME | Each arc is associated with a travelling time tm,i,j | Half Implemented |
| #13  FITNESS COEFFICIENT | Fitness coefficient fm,i,j ∈ [−1, 1], which represents a sort of “health gain” or “health loss”, according to the mode of transportation chosen. | Not Implemented |

# Info

For complex constraints a line explaining what the constraint is controlling should be placed above. *[An additional detailed note about how exactly if works can be placed above the constraint in square brackets and italics if needed, keep this to a minimum]*

Please just steal maths from the file “DODM\_07\_ILP-VehicleRoutingProblems.pdf” and leave comments about what you’re stuck on [Review Tab > New Comment]

# Documentation Notes

* The commodities that a person must track are:
  + money, number of times changed transportation method, spending costs, current mode of transportation, health,
  + We will have to nick (steal) constraints from commodity flow maybe? I would just like to sum out where they go i.e. [sum(places been \* cost of places) < total money] but I’m not sure it’s that simple
* The assumption about a person can’t return to a node they have previously travelled too must be confirmed
* The capacity of nodes and modes of transports is not yet considered by the model

# Assumptions

* A person will not return to a node they have personally travelled to (this needs to be confirmed)
* A person will only complete a single task at each node
* A bus spends no time at a bus stop
* A bus can be caught anytime within a x second window before it arrives at a point. This time is called the bus relaxation (I assume 60 seconds is fine, however we can tune this)
* The travel times of two people riding the same bus are modelled separately and may lag between each other according to the bus relaxation parameter
* The indexing of modes of transport is currently assumed to be {1 = walking, 2 = Cycling, 3 = Bus}. I will be worth asking the prof if we are ok to always assume this is the case. Otherwise, there will have to be some hardcoding in the model to catch instances in the inputs where this is not the case.
* The bike station capacity is checked at x min intervals (default 30 mins), there will have to be a consultation about how this linear relaxation could be best handled
* Every person must leave their home

# Objective functions

Determine the tours that all people have to do in order to perform their tasks, by considering the different modes of transportation available and all the constraints, in order to minimize the total travelled time by also taking into account the overall fitness score.

# Indexes

# Sets

# Special Subsets

# Constants

# Independent Variables

# Semi-Dependant Variables

***These are the variables that are technically dependant variables but are modelled as constrained independent variables***

# Dependent Variables

***This section is saved for any variable which are fully dependant on other variables, this is generally used if various variables need to be consolidated into a single figure i.e., the sum of costs. Please note that when using these variables, the developer needs to be careful not to make any illegal calculations this new variable***

# Constraints

## Basic Conservation of Flow

Flow is directional, multi-medium, multi-flow (person)

*[the entries/exits from a node j, needs to be larger than one if there is a task at the node or if it is the home node of the person (h) (Boolean values)]*

Currently each node can only be visited once to not interfere with constraints around timing

The person must enter and exit their home

## Task Timing

This controls the time (w) which the task at j starts

*This doesn’t apply where a person is returning to their home node|*

This controls that tasks happen within their designated time windows (only applies if the tasks happens and the task extension penalty for falling out of the allotted time isn’t applied)

Task extensions/delays only apply to tasks that are undertaken

Task extensions/delays only apply to tasks that are special

A person will only complete a single task at each node

## Bus Travel Constraints

These two constants state that person must finish their task and waiting period at node i, x seconds (controlled by the bus relaxation constant) before the exact bus they want to catch arrives at

A person can only leave a node once

The person n can only get bus (i, d, r) the relevant Bus\_idrn must be positive

Every time a person gets on a bus from another mode of transport, they must purchase a bus fare

## Bike Capacity Constraints

The bike station capacity is checked at x min intervals (default 30 mins), there will have to be a consultation about how this linear relaxation could be best handled

At the end of the period, there must be a non-negative quantity of bikes and free spaces at each bike station

## Unfinished Tasks Penalties

A penalty Qt must be paid for a task t ∈ Tn not performed

## Starting total amount of money (Knapsack)

A given amount of money Wn

## Maximum people in a place

A place p cannot be visited by more than Np people at the same time

## Transportation mode change

Each person n ∈ N can change their mode of transportation at most Kn times

## Maximum people in a bus

A maximum number of people who can be on a bus at the same time is Nℓ

## Fitness coefficient

Which represents a sort of “health gain” or “health loss”, according to the mode of transportation chosen, where f= fitness score