University of Amsterdam

Masters Thesis

Geospatial healthcare capacity allocation

Case study of allocating beds to stroke centers in Vietnam

Author:
Corien Westveer
Student number:
13412884

Supervisor:
Prof. Dr. Ir. Dick den Hertog
Examiner:
Prof. Dr. Rob van der Mei
Second Assessor:
Prof. Dr. Joaquim Gromicho

A thesis submitted in partial fulfilment of the requirements for the degree of Master of Science in Computational Science

in collaboration with

Analytics for a Better World
The World Bank

August 2022

Declaration of Authorship

I, Corien Westveer, declare that this thesis, entitled 'Geospatial healthcare capacity allocation' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at the University of Amsterdam.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date: 1 August 2020

UNIVERSITY OF AMSTERDAM

Abstract

Faculty of Science
The World Bank
Analytics for a Better World

Master of Science in Computational Science

Geospatial healthcare capacity allocation

by Corien Westveer

This thesis proposes a Marginal Allocation (MA) method using queueing theory and an Optimization with Constraint Learning (OptiCL) approach using simulation to find the optimal allocation of capacity for healthcare facilities. The main research goal is to find a method to optimally allocate capacity to healthcare facilities given a budget to add capacity. The proposed models have been tested on stroke centers in Vietnam. Two sets of starting assumptions are explored: (i.) when assuming patients have only access to the closest facility, the MA method can be used to give the allocation of capacity for which either the average waiting time of patients or the waiting probability per patient is minimized, and (ii.) when assuming that patients can be forwarded between facilities, OptiCL can be used to give the allocation of capacity for which the average travel time per patient is minimized. A simulation is built to find the average travel time for a given allocation of capacity. Based on simulation results a machine learning model is built to predict the average travel time for a given allocation of capacity, which is then embedded in an optimization model. The marginal allocation algorithm is an exact algorithm with short runtime. The performance of the OptiCL method differs greatly for different predictive models. The OptiCL method with gradient boosting method as predictive model gives the allocation with smallest travel time, this travel time was close to but not smaller than the best allocation in the training dataset. This means that the MA method is suitable to allocate capacity in case patients have only access to the closest facility. The OptiCL method is a promising method for this type of problems, as the travel time of the best allocation is close to the best travel time in the dataset.

Acknowledgements

I would like to thank the following people for their help and guidance during the project.

First, I want to thank Dick den Hertog for his good guidance during the thesis and his inspiring enthusiasm to make the world a better place with help of analytics. I thank Rob van der Mei for his helpful suggestions and feedback during the thesis.

I am grateful to Parvathy Krishnan, Joaquim Gromicho and Britt van Veggel for the valuable contributions to improve my research during our weekly meetings. A special thanks to Donato Maragno for his help with OptiCL.

I would like to express my gratitude to Bas, my friends and family for their support during the thesis.

Contents

Chapter 1

Introduction

1.1 Research background

Strokes are the main cause of death and main cause of years of life lost in Vietnam [1]. Over the course of 20 years, the number of strokes has increased from 214 out of 100,000 persons per year in 1990 to 255 out of 100,000 persons per year in 2010 in Vietnam [2]. According to Cong [3] one of the biggest problems around stroke care in Vietnam, similar to other developing counties, is that the number of stroke victims is enormous and continuously increasing, while the capacity to treat them is limited. Rymer et al. [4] show that earlier treatment of patients after a stroke significantly reduces damage incurred by a patient after a stroke incidence. Therefore, the most effective way to reduce the impact of a stroke is to assure quick access to treatment. This goal can be achieved in multiple ways, predominantly by building new stroke facilities or by adding additional capacity to existing stroke facilities.

In case there is some budget available invest in new stroke facilities, the question arises where to locate these facilities. This could be formulated as a facility location problem, or more specifically as an uncapacitated maximum covering problem by stating that new facilities should be located in such a way that the number of people who can access a facility within a fixed distance is maximized. Two such have been developed to solve the uncapacitated maximum covering problem. Antonissen [5] developed a mathematical model to solve the maximum covering problem and applied it to hospitals in Timor-Leste. Theulen [6] developed a GRASP heuristic to solve large scale maximum covering problems and applied it to stroke centers in Vietnam.

Both of these algorithms are part of the Geospatial Planning and Budgetting Platform (GPBP). This platform provides digital decision support for decision makers at

national and sub-national government level in order to optimize the impact of public infrastructure investments in health facilities and transport networks [7]. GPBP is a collaboration of Analytics for a Better World and The World Bank. The Analytics for a Better World institute contributes to the Sustainable Development Goals of the United Nations through the application of analytics. The World bank is an international organization, which provides financing, policy advice, and technical assistance to governments of developing countries [8]. The mission of the World Bank is to end extreme poverty and to promote shared prosperity [9]. With their collaboration on the GPBP tool they work together in their shared goal to improve healthcare, with a focus on developing countries.

With the creation of the facility location models as part of the GPBP tool, new question arises. If some budget is available to add capacity to healthcare facilities, at what facilities should capacity be added to benefit the patients most? The objective of this is twofold: (i.) find the most suitable model to answer the beforementioned question and (ii.) apply the model to the stroke center optimization problem in Vietnam. The significance of the question cannot be underestimated with an enormous number of stroke victims in this country combined with limited treatment capacity.

1.2 Research question

The research question of this thesis can be formulated as: what is a computationally attractive model to allocate capacity to healthcare facilities in such a way that the benefit for patients is optimized? It is of importance that the model is generic, meaning that it can be applied to a broader range of geospatial healthcare capacity allocation problems. The model will be applied on stroke centers in Vietnam, where the question is: given a fixed budget to add beds, how many beds should be added to each stroke center to benefit the stroke victims most? The goal is to make the resulting algorithm part of the GPBP tool.

1.3 Approach

To answer the research question, we start with getting a better understanding of the problem. Secondly, it is examined what suitable performance indicators are to measure if an allocation is beneficially to patients. Furthermore, it is examined which models in literature could solve this capacity allocation problem. The scope is limited to two types of models: one model where we make the assumption that a person has only access to the

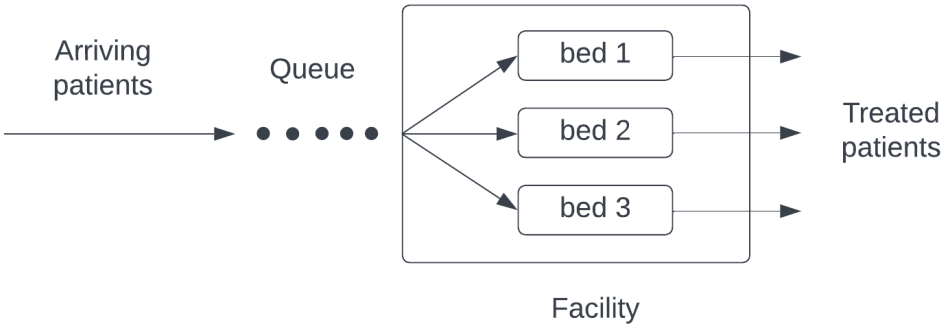

FIGURE 1.1: Qeueuing process at a facility with a capacity of 3 beds.

closest facility and another for allowing forwarding of patients between facilities. Making the mentioned assumption gives the opportunity to use a more simple, computationally less demanding model, whereas the model that allows forwarding between facilities will be closer to reality.

For the first type of model catchment areas will be created around each facility based on travel time. In this way, the process at each facility can be reframed as a queueing system, illustrated in fig. 1.1. This allows to redefine most beneficial to patients in this model as the minimum average waiting time or waiting probability of patients. Average waiting time of patients is defined as the time a patient has to wait on average before receiving treatment. Waiting probability of patients is defined as the probability an arriving patient finds the facility in a status where no capacity is available to immediately receive treatment, i.e., all beds are occupied. The model will be tested on the Vietnam case. First, the relationship between the average probability of waiting and average waiting time will be investigated and then the model will be investigated in more detail by adding 500 extra beds. Next, we will research the runtime, scalability and exactness of the algorithm.

In the second type of model we do not consider catchment areas around stoke centers, but instead we assume that a stroke victim can receive treatment at every stroke center. This assumption is closer to reality, because for example in city areas where multiple facilities are close to each other stroke victims have multiple facilities to choose from. Disadvantage is that it will increase the complexity of the model. As performance indicator for this type of model the average travel time of patients can be used. An illustration of the how the path a patient takes is defined and how this relates to the travel time of a patient is shown in fig. 1.2. Since in this case no explicit mathematical formulas can describe the average travel time of the patients receiving treatment, a simulation will be developed to estimate the average travel time for a given bed allocation. To learn more about the working of the simulation, a couple of scenarios are further investigated

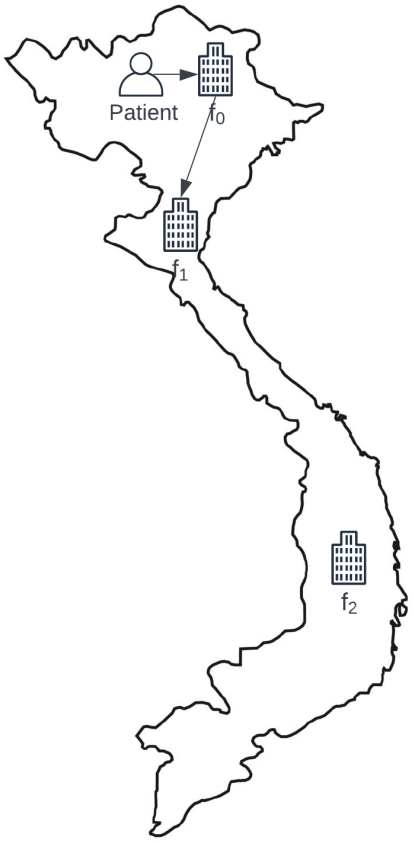

FIGURE 1.2: Example of a path a patient takes until receiving treatment. The patient starts from his/her residence and travels to the nearest facility f_0. Since no capacity is available at facility f_0, the patient is forwarded to the closest facility from this point f_1. Since there is capacity available at facility f_1, the patient receives treatment at this facility and facility f_2 is not visited. The travel time of this patient is the time to travel from his/her residence to f_0 plus the travel time from f_0 to f_1.

with different input parameters to see how they influence the average travel time. Next the results for the Vietnam case are given. Secondly, a mathematical model will be created that minimizes the average travel time of patients in order to find the optimal allocation. Since the average time between stroke incidence and treatment is not an explicit function of the bed allocation, a machine learning model will be trained based on the simulation results to find an expression for this. This machine learning model will be embedded into the mathematical model, also called Optimization with Constraint Learning (OptiCL) and the results for the Vietnam case will be given.

1.4 Outline

This report is structured in the following way. Chapter 2 gives a description of the relevant theory for the used models. Chapter 3 gives the used methodology to solve the research question. Chapter 4 presents the results of the proposed models. In chapter 5 the discussion of the work is given. The thesis ends with a conclusion and future work in chapter 6.

Chapter 2

Theory

This chapter presents an overview of related literature. First, a description of the facility location model is given in Section 2.1, since this thesis is sequential to research on the facility location problem. In the approach to the capacity allocation problem a distinction is made between two types of models, the first model where it is assumed that patients have only access to the closest health facility and the second model where it is assumed that patients can be forwarded between facilities. Making the assumption that patients can only go to the closest stroke center, allows to see the process at each stroke facility as a queueing system. Therefore the relevant queueing theory is described in Section 2.2. The method that will be used to allocate the capacity to the stroke centers in case we see the process at each facility as a queuing system is the MA method, which will be discussed in Section 2.3. For the second type of model, it is assumed that patients can be forwarded between facilities. The goal in this model is to minimize the average travel time of patients. The average travel time of patients for a specific allocation of capacity is estimated using a discrete event simulation, for which the theory is described in Section 2.4. To find the allocation of beds for which the average travel time is minimized we make use of OptiCL, for which the literature is discussed in Section 2.5.

2.1 Facility Location model

One of the previous goals for the GPBP tool was to develop a facility location model to help to decide where new hospitals should be located to gain the greatest increase in the accessibility of hospitals. The model was created for hospitals in Timor-Leste, but has also been applied to find optimal locations for stroke centers in Vietnam. The formulation of the facility location model was based on the Maximal Covering Location Problem proposed by Church and Revelle [10]. The model is defined in the following

way:

$$\text{Maximize} \quad \sum_i v_i y_i \tag{2.1a}$$

$$\text{Subject to} \quad x_j = 1 \qquad \forall j = 1, ..., m \tag{2.1b}$$

$$\sum_{j=m+1}^{M} x_j \leq p \tag{2.1c}$$

$$y_i \leq \sum_{j|d_{ij} \leq S} x_j \qquad \forall i \in I \tag{2.1d}$$

$$y_i \in \{0, 1\} \qquad \forall i \in I \tag{2.1e}$$

$$x_j \in \{0, 1\} \qquad \forall j \in J, \tag{2.1f}$$

where:
- I = the index set of households, or clusters of households, indexed by $i = 1, ..., n$
- J = the index set of all healthcare sites, where indices $j = 1, ..., m$ are corresponding to the already existing healthcare facilities and indices $j = m + 1, ..., M$ are corresponding to potential hospital locations
- v_i = the number of people in (cluster of) household(s) i
- d_{ij} = the travel distance from (cluster of) household(s) i to hospital facility j
- S = the maximum travel distance from a household (or cluster) to a health care facility
- p = the number of additional hospitals located.

The model has two decision variables: $x_j = \begin{cases} 1 & \text{if hospital } j \text{ is opened} \\ 0 & \text{otherwise} \end{cases}$

$y_i = \begin{cases} 1 & \text{if there is an opened hospital within } S \\ & \text{kilometers travel distance away from the (cluster of) household(s) } i \\ 0 & \text{otherwise.} \end{cases}$

The goal of the model, equation 2.1a, is to maximize the number of people that can reach a facility within a maximum travel distance. Constraint 2.1b ensures that the already existing health facilities are built. Constraint 2.1c ensures that at maximum p new hospitals can be built. The next constraint, 2.1d, makes sure that households can only be assigned to hospitals that are built within the travel distance. The last two constraints, 2.1e and 2.1f, make the decision variables x_j and y_i binary variables.

The above model is not constraint on capacity, meaning that the capacity of a hospital is infinite. This assumption might not be true for real facilities. Therefore one might

be interested in facility location formulations that are restricted on capacity. Amadi-Javid et al. [11] and Boonmee et al.[12] provided a survey on healthcare facility location models, including facility location model formulations with capacity constraints. One could think that capacitated facility location models are the answer to our capacity allocation and facility location problem. But this type of model cannot provide a good way of allocating capacity. The reason is that in capacitated facility location models the capacity at each facility is restricted, i.e., capacity is not a decision variable in this type of model. In the capacity allocation problem of this thesis it is assumed that there are no boundaries on the capacity that can be added at each facility, but that the capacity should be added to facilities in such a way that patients benefit most.

Since capacitated facility location models are not a good option to use to allocate capacity, other type of models are investigated. The first model proposed in to solve the capacity allocation model is a model that allocates capacity based on queueing formulas. For this model it is assumed that people go to the most closest facility and if no place is available, enter a queue. For such type of model the allocation with the most benefit for people can be defined as the allocation with the smallest average waiting probability or smallest average waiting time. We therefore discuss relevant queueing theory in the next section.

2.2 Queuing theory

Before describing queueing theory, it is first emphasized why it is profitable to allocate capacity based on queueing performance measures instead of allocating capacity proportional to the number of people in the catchment area of a facility, as this would be an obvious way to allocate capacity. This will be done using an example. Let us say there are two facilities f_1 and f_2, with a catchment area of 70,000 and 700,000 people respectively and we assume no forwarding between facilities. Let us assume that the probability of people getting an illness is 0.00005 and that it takes on average two days to treat a person for this illness at the facility. Furthermore, it is assumed that the inter-arrival time and treatment time of patients are independent and identically distributed. It is assumed that the arrival distribution of patients follow a Poisson distribution and that the treatment times of patients follow an exponential distribution. The task is to allocate 110 beds to the facilities. A first idea could be to allocate the beds proportionally to the number of people in the catchment area, leading to the first facility receiving 10 beds and the second facility 100 beds. Another idea is to define the optimal allocation of beds as the allocation of beds where the average probability of waiting is minimal. In that case the optimal allocation of beds would be 14 beds at f_1 and 96 beds at f_2. When

calculating the average probability of waiting using the Erlang C formula, given later in this section, the allocation proportional to the number of people in the catchment area the average probability of waiting would be 0.021. When allocating the beds in such a way that the average probability of waiting is minimal the average probability of waiting is 0.0031. This example shows that with the use of queueing formulas one can reach an allocation with lower average probability of waiting compared to allocating capacity proportional to the number of people in the catchment area. A similar conclusion can be drawn if one would use average waiting time as performance measure instead of average probability of waiting. With showing why it is profitable to use queuing formulas to allocate capacity, this section is continued with a description of queuing theory.

According to Gross et. al. [13], a queueing system can be described as customers that arrive for service, waiting for service and leave the system after being served. Some customers may leave the system without receiving service, for example because they do not want to wait longer than a certain amount of time. In this description, the term customer is used, but this should not be seen as only a human customer, but as anything or anyone that wants to use the serves. In our allocation problem, the customers can be seen as patients and the servers are the number of beds available at a stroke center.

Among others, a queueing system can be characterized by:

1. Arrival pattern of customers
   In common queueing systems the arrival process is stochastic and can be described by a probability distribution of the interarrival times. It is also necessary to know whether customers arrive one-by-one or in batches. Besides that, it is also important to know what the behaviour of customers is. For example, if they decide to not enter the queue if the queue is too long, or keep waiting no matter how long the queue becomes. The last thing to consider is if the arrival pattern changes with time, i.e., if the arrival pattern is stationary (time-independent) or non-stationary.

2. Service pattern
   Similarly to the arrival pattern, the service pattern is described with a probability distribution and the service can be single or batch and stationary or non-stationary. Besides that, the service progress may be dependent on the queue length, for example a server may work faster if a queue is building up.

3. Queuing discipline
   Queueing disciple is the way customers are selected from a queue to be helped. The most common discipline is first come, first served (FCFS).

4. System capacity
   In some queueing systems the length of the waiting line is limited, meaning that

when the waiting line has a certain length, arriving customers will not enter the waiting line.

Besides a description of the queueing system, we usually want to know something about the effectiveness of the system. These performance measures can be measures of the wait time, measures of the way customers accumulate or a measure of the idle time of the server. There are two types of wait time: the time a customer waits for service and the time a customer spends in total in the system. There are also two types of measures of customer accumulation, the number of customers in the queue and the number of customers in the system. Idle time of the server can be measured by either the percentage of time the server is idle or the time there are no customers in the system.

For the GI/G/s system (queuing system with general arrival and service distribution) the probability that an average server is busy ρ can be calculated as $\rho = \lambda/(s\mu)$ with λ the average arrival rate of customers, μ the average service rate of customers and s the number of servers. When $\rho \geq 1$ the queueing line will get bigger and bigger when time goes on. If we are interested in steady state conditions (the state of the system after a long time), it is necessary that $\rho < 1$ for steady state results to exist.

In this thesis we will focus on M/M/s queueing systems, this is a queuing system with exponential interarrival times, exponential service times, multiple servers, a FCFS priority scheme and no restriction on the maximum allowed number of customers in the system. Some relevant performance measures of the M/M/s system are given in this paragraph. Two of such performance measures, the probability that a customer has to wait and the average waiting time are described by Takagi and Walke [14]. Equation 2.2 gives the probability that a customer has to wait and is also called the Erlang-C-formula and equation 2.3 gives the average waiting time

$$P(W_q > 0) = E_c(s,a) = \frac{a^s/((s-1)!(s-a))}{(\sum_{j=0}^{s-1} a^j/j!) + a^s/((s-1)!(s-a))} \tag{2.2}$$

$$E(W_q) = \frac{E_c(s,a)}{(s-a)\mu}, \tag{2.3}$$

with $a = \lambda/\mu$ the traffic intensity and s the number of servers. For both of the above steady-state conditions to exist, it is required that $\rho < 1$.

2.3 Marginal allocation

For the first model proposed to allocate capacity it is assumed that every patient goes to the closest facility. In that case each facility can be seen as a queueing system operating

parallel to each other. The method of marginal allocation (MA) can be used to find the optimal allocation of capacity in number of beds that minimizes the sum of average queueing time at the facilities.

Weber [15] describes the MA as a method to allocate servers amongst facilities in order to minimize the sum of mean queuing times at the facilities. To explain the MA method we consider a situation with N facilities. Each facility can be seen as a GI/G/s queueing system. This means that at each facility a number of servers are available that operate in parallel and serve customers using a FIFO priority scheme. The interarrival time of customers at facility i are independent and identically distributed with mean $1/\lambda_i$. The service time of customers at facility i is independent and identically distributed as well with mean $1/\mu_i$. We define the customers queueing time as the time a customer waits until being served. The mean queueing time of the facility is defined as the expected steady-state (stationary) queueing time at the facility. It is assume each facility has enough servers to ensure finite mean queueing time, i.e., $rho < 1$. The task is to allocate M additional servers at the facilities in such a way that the mean queueing times at the N facilities are minimized. This optimal allocation can be generated by the method of MA.

The MA method allocates each of the additional M servers one by one. This means that the optimal allocation of M servers can be found by first optimally allocating $M-1$ servers and than allocating the Mth server to the facility where it leads to the greatest reduction in mean queueing time. Let us assume that after optimally allocating $M-1$ servers, the number of servers at facility i is m_i (i = 1, ..., N). We call W_m the mean queueing time at facility i. The optimal allocation of the Mth server to facility j is given by:

$$\bar{W}^j_{m_j} - \bar{W}^j_{m_j+1} = \max_{1 \leq j \leq N} \{\bar{W}^i_{m_i} - \bar{W}^i_{m_i+1}\}. \tag{2.4}$$

The MA algorithm is optimal if the mean queuing time at each facility is a non-increasing and convex function of the number of servers at the facility. Weber [15] proves that this is the case for GI/G/s queues. Using the MA algorithm saves a lot of computation time compared to calculating the average waiting time for all possible allocations of servers.

2.4 Discrete Event Simulation

To calculate the average travel time of patients a Discrete Event Simulation will be used. Discrete Event Simulation is a specific type of simulation where the system is modelled as a network of queues and activities and where the state of the system changes at

discrete points in time. The objects in the system are distinct individuals with their own characteristics and the activity durations are sampled for each individual from probability distributions [16].

A typical goal of a discrete event simulation is to find the steady state estimator of a certain quantity. At the start of the simulation one needs to choose a starting state, which may results in an initialization bias. There exist different methodologies in dealing with this initialization bias, including choosing an appropriate starting state or one can start collecting data after the simulation did run for a certain time period. This period is called the *warm-up period* or *initialization phase* [17].

When the goal of the discrete event simulation is to find the steady state mean, one usually wants to compute a confidence interval as well. Two common methods to estimate the confidence interval are the method of batch means (BM) and the method of multiple independent replications (MR). For both methods we assume that the underlying stochastic process $\{X_i\}_{i\geq 1}$ is stationary.

The BM method requires to make one long run and then partitions the observations $X_1, ..., X_n$ in $b > 1$ non-overlapping batches with each m observations. The total number of observations in the run should be $n = bm$. The batch mean $\bar{X}$ of batch $j = 1, ..., b$ is

$$\bar{X}_j = \frac{1}{m} \sum_{i=1}^{m} \bar{X}_{(j-1)m+i}. \tag{2.5}$$

The estimator for the variance of BM is

$$\hat{V}_{BM} = \frac{m}{b-1} \sum_{j=1}^{b} (\bar{X}_j - \bar{X}_{BM})^2, \tag{2.6}$$

with $\bar{X}_{BM} = \sum_{i=1}^{n} X_i/n$. When m is large and the number of batches are fixed, it can be assumed that the batch means are independent and identically distributed. Because of that the confidence interval of μ is

$$\mu \in \bar{X}_{BM} \pm t_{b-1,\alpha/2} \sqrt{\hat{V}_{BM}/n}, \tag{2.7}$$

with $t_d, \alpha/2$ the $1 - \alpha/2$ quantile of the t distribution with d degrees of freedom.

The MR method consists of k independent replications, each consisting of m observations. The replication mean $\bar{Y}$ of replication $r = 1, ..., k$ is

$$\bar{Y}_r = \frac{1}{m} \sum_{i=1}^{m} X_i^{(r)}, \tag{2.8}$$

where $X_i^{(r)}$ is the i^{th} observation of the r^{th} replication. The sample variance is

$$\hat{V}_{MR} = \frac{m}{k-1} \sum_{r=1}^{k} (\bar{Y}_r - \bar{X}_{MR})^2, \tag{2.9}$$

with $\bar{X}_{MR} = \sum_{r=1}^{k} \bar{Y}_r / k$. When m is large, it can be assumed that the replication means are independent and identically distributed random variables, thus the confidence interval for μ is [18]

$$\mu \in \bar{X}_{MR} \pm t_{k-1,\alpha/2} \sqrt{\hat{V}_{MR}/n}. \tag{2.10}$$

2.5 Mixed-Integer Optimization with Constraint Learning

When assuming patients can be forwarded between facilities, the goal is to find the optimal allocation of beds that minimizes the average travel time of patients. There are no explicit formulas available to calculate the average travel time for a specific allocation of beds. Therefore simulations are used to calculate the average travel time. Based on these simulation results we can create a predictive model to predict the average travel time based on an allocation of beds and embed this predictive model in the goal function of an optimization model. The use of predictive models in the constraints (and goal function) of a mixed-integer optimization model (MIO) is also called Mixed-Integer Optimization with Constraint Learning (OptiCL).

In some cases there exist no explicit formulas for the objectives or constraints in a mixed-integer optimization model, but there is data available to learn the constraints or objectives. An example would be the case study Maragno et al. [19] used to showcase OptiCL. This case study gives a model which seeks to optimize the humanitarian food aid, of which an extended version is used at the World Food Programme as a tool for long-term recovery operations. The model of the case study enforces the food baskets to meet the nutritional requirements and be palatable. No explicit formulas exist for palatability, but with data on palatability a machine learning model can be built to find an expression for palatability and this can be embedded in a optimization formulation. Maragno et al. [19] propose a method for MIO with learned constraints. They create an end-to-end pipeline for data-driven decision making in which objectives and constraints are learned from data using machine learning, and the trained models are embedded in an optimization formulation. A schematic overview of the pipeline is given in fig. 2.1. The first step in the pipeline is to create a conceptual model, followed by a data pre-processing step. Next, predictive models are used to built an expression for the learned constraints and objectives and a trust region is defined, resulting in a MIO model with

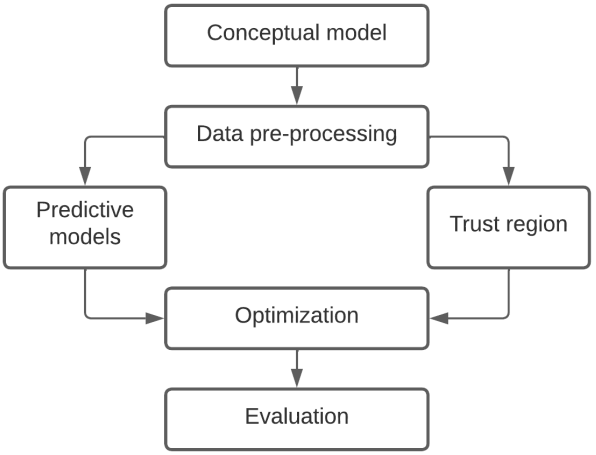

FIGURE 2.1: Constraint learning and optimization pipeline

learned predictive models and trust region constraints. The last step is the evaluation of the optimal solution and the embedded predictive models' performance.

Conceptual model

Given the decision variable $x \in \mathbb{R}^n$ and the fixed feature vector $w \in \mathbb{R}^p$, the conceptual model can be defined as $M(w)$:

$$\begin{aligned} \min_{x \in \mathbb{R}^n, y \in \mathbb{R}^k} & f(x, w, y) \\ \text{s.t. } & g(x, w, y) \leq 0, \\ & y = \hat{h}_D(x, w), \\ & x \in \chi(w), \end{aligned} \tag{2.11}$$

where $f(., w, .) : \mathbb{R}^{n+k} \mapsto \mathbb{R}$,$g(., w, .) : \mathbb{R}^{n+k} \mapsto \mathbb{R}^m$, and $\hat{h}_D(., w) : \mathbb{R}^n \mapsto \mathbb{R}^k$. Explicit forms of f and g are known, but may depend on the predicted outcome y. $\hat{h}_D(x, w)$ stands for the predictive model, trained on data $D = \{(\overline{x}_i, \overline{w}_i, \overline{y}_i)\}_{i=1}^N$, with observed treatment decisions $\overline{x}_i$, contextual information $\overline{w}_i$, and outcome of interest $\overline{y}_i$ for sample i. χ is the set that defines the trust region, i.e. the solutions for which the embedded predictive models are trusted.

Model $M(w)$ can be used for different constraint learning classes: regression, classification and objective function. When the model is trained for a regression problem, it can be constrained by an upper bound τ, with the constraint $g(y) = y - \tau \leq 0$ or lower bound τ, with the constraint $g(y) = -y + \tau \leq 0$. If the trained model results from a

binary classification algorithm, in which the data is labelled as feasible or infeasible, the prediction is usually a probability $y \in [0, 1]$. In that case we can put a lower bound on the feasibility probability, $y \geq \tau$. In case the objective function has a term that is learned by a machine learning model, an auxiliary variable $t \in \mathbb{R}$ can be added to the objective function along with an epigraph constraint. In case the optimization model involves only a single learned objective and no learned constraints, then the optimization model becomes

$$\begin{aligned} \min_{x \in \mathbb{R}^n, y \in \mathbb{R}, t \in \mathbb{R}} \quad & t \\ \text{s.t.} \quad & g(x, w) \leq 0, \\ & y = \hat{h}(x, w), \\ & y - t \leq 0, \\ & x \in \mathcal{X}(w). \end{aligned} \tag{2.12}$$

Athough the problem is rewritten to prove the generality of the model, it is common in practice to use y in the objective and omit the auxiliary variable t.

Predictive models

To be able to use the machine learning model in a MIO model, the machine learning model needs to be embedded in the MIO model using linear constraints. This is possible for many types of ML models. Maragno et al. [19] provide the embedding in the MIO model for Linear Regression, Support Vector Machines, Decision Trees and ensemble methods, such as Random Forest and Gradient Boosting Machines and multilayer perceptrons (MLP) with a rectified linear unit (ReLU) activation function, a MIO representable class of neural networks.

Trust region

The optimal solution of optimization problems are often found at the extremes of the feasible region. This can be problematic for the trained predictive model, because the accuracy of a machine learning model get worse for samples further away from the samples in data D. In order to solve this problem, a *trust region* is introduced. This trust region prevents the machine learning model from extrapolating. The trust region is constructed by a clustering step to identify the high density regions of the observed input data, and then the trust region is represented as the union of the convex hulls of the individual clusters. The convex hull can be represented by linear constraints, but then the computation time increases with the number of samples in D. Therefore, a

column selection algorithm is used to select a small subset of samples to decrease the computation time.

Chapter 3

Methods

3.1 MA using queueing formulas

3.1.1 Method of MA algorithm using queuing formulas

The MA model is the first model proposed to solve the problem of allocating capacity to healthcare facilities. In this model it is assumed that people have only access to the closest facility. Therefore catchment areas are defined around each facility based upon travel time. As an example, the catchment areas around the stroke facilities in Vietnam are shown in fig. 3.2. This assumption will only hold if the travel times between facilities are large and there is no possibility to go to a different facility than the closest one. This assumption might for example not hold for people that have approximately equal travel distance to multiple facilities or for larger cities, where patients can be forwarded between facilities.

By creating catchment areas, the behaviour at each facility can be seen as a GI/G/s queue that operates in parallel to the other queues and serves patients in order of their arrival. It is assumed that the arrivals of patients at facility i follow a Poisson process with rate λ_i. Furthermore, it is assumed that treatment times are equal for all facilities. Treatment times of patients are also independent and identically distributed with an average of $1/\mu$. When considering behaviour at each facility as a queue, an adjusted version of the MA algorithm described in Section 2.3 can be used to determine the optimal allocation of capacity. In this case we see the capacity at a facility as the number of servers at a facility, which translates to the Vietnam case as the number of beds per stroke center. The optimal allocation of capacity can then be seen as either the allocation of capacity for which the average probability of waiting per patient is minimized or the allocation of capacity for which the average waiting time per patient

is minimized, both under the constraint that the sum of the number of beds allocated cannot exceed a certain budget.

To make the MA algorithm (described in section 2.3) suitable for our needs, some additions are made to the original algorithm:

1. The MA algorithm works only when the number of servers is sufficient, since the mean queuing time tends to go to infinite when not enough servers are available at a facility to treat the patients. However, for the Vietnam case it is not known if the number of beds at each facility is sufficient to ensure that mean queueing time will not go to infinity. Therefore we first start with assigning servers to facilities that do not have enough servers before applying the MA algorithm. The number of servers of a facility is high enough to ensure the waiting time at the facility i is finite in case the utilization rate of facility i $\rho_i = \lambda_i/(s_i\mu) < 1$, with λ_i the arrival rate at facility i, s_i the current number of beds at facility i and μ the treatment rate. Before applying the MA algorithm beds are assigned one by one to the facility based on the highest value of ρ_i until $\rho_i < 1$ for all facilities.

2. Instead of minimizing the sum of the mean queueing times at the facilities, we want to minimize the average waiting time of patients. The average waiting time is calculated as the sum of the mean queueing time at each facility multiplied with the part of the population in the catchment area of the facility.

3. It is assumed that the queueing process at the facilities can be described as $M/M/s$, however as the original MA algorithm described in section 2.2 is proven to be exact for $GI/G/s$ queues, it would be possible to extend the algorithm to other queueing types. For the $M/M/s$ system there is an explicit formula to calculate the average waiting time at a facility, given in eq. (2.3).

4. The original MA algorithm minimizes the sum of mean queueing time, but besides queueing time, it can be an interesting performance measure as well to minimize the average probability of waiting for each patient. The MA algorithm can be used to calculate the probability of waiting per patient in case the probability of waiting at each facility is a non-increasing convex function of the number of servers (beds). Assuming $M/M/s$ queueing type, the probability of waiting at a facility can be calculated by the Erlang C formula, with the number of servers and traffic intensity $a = \lambda/\mu$ as input. Although we suspect the Erlang C formula to be non-increasing convex in the number of servers (and equal traffic intensity) we have not found a proof of this statement in literature. Therefore, it is tested numerically whether the condition of convexity holds for the Erlang C formula E_c, eq. (2.2), on a interval by checking if for every $0 < t < 1$ and all $x, y, x \neq y$ in

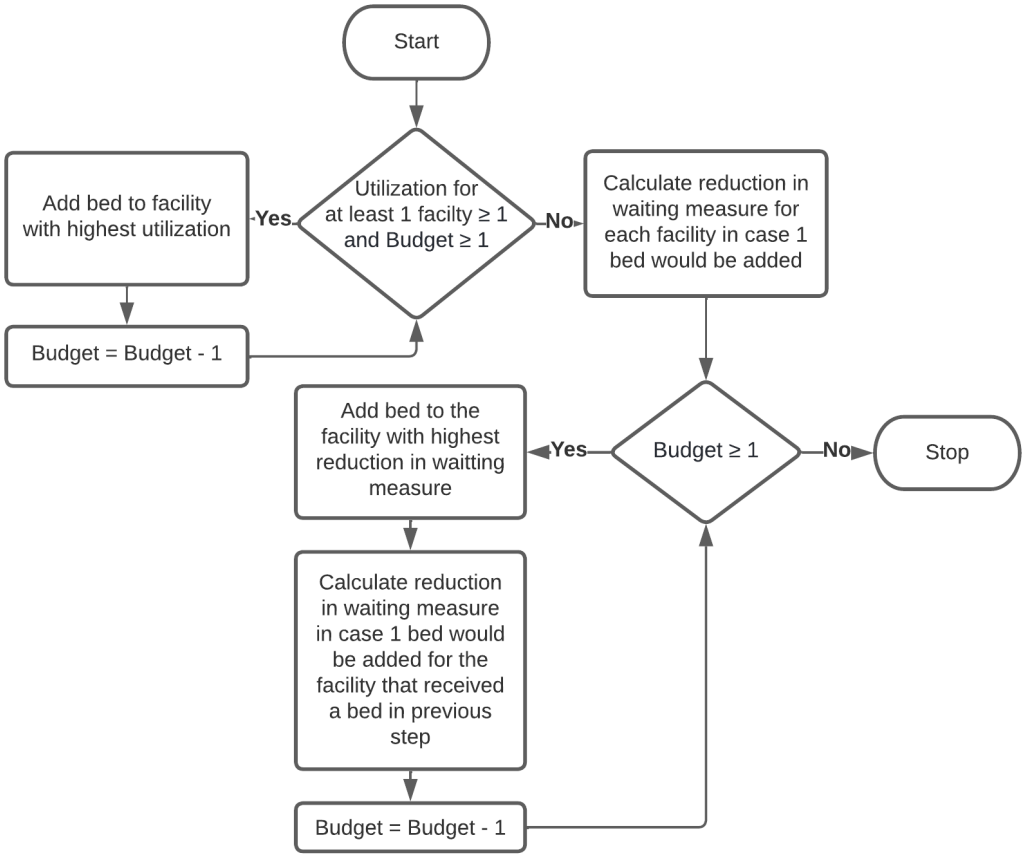

FIGURE 3.1: Flowchart of MA algorithm based on queuing time. Budget is the budget of capacity in number of beds. In case the MA algorithm is used to minimize the average waiting time of patients, the waiting measure in the algorithm is the average waiting time at the facility multiplied with the number of people in the catchment area of the facility. In case the MA algorithm is used to minimize the waiting probability of patients, the waiting measure is the probability of waiting at a facility multiplied with the number of people in the catchment area of the facility.

the interval the condition $E_c(tx + (1-t)y) < tE_c(x) + (1-t)E_c(y)$ holds. It is checked if this condition holds for traffic intensities 0.1 to 99.9 with steps of 0.1 and the servers ranging from the minimum number of servers required to have a utilization smaller than 1 up to 60 extra beds. For all these values the Erlang C formula is convex in the number of beds.

A flowchart of the MA algorithm is given in fig. 3.1. The input parameters for the MA algorithm are the arrivalrate λ_i for each facility i, treatmentrate μ, the number of beds at each facility, the number of people in the catchment area of each facility and the budget in number of beds. The output of the MA algorithm is the optimal allocation of beds.

3.1.2 Application to Vietnam case

The MA algorithm uses λ, μ, the current number of beds, the number of people in catchment area of each facility and a budget of beds as input variables. This section explains how these variables are determined for the Vietnam case study.

Number of people in catchment area

In order to calculate the number of people in the catchment area each person of the population needs to be assigned to the stroke center that can be reached within the shortest travel time. Since no information on travel times is available, travel distance will be used as a proxy to assign the population to the stroke centers. It is to be expected that the catchment areas based on travel distance are similar to the catchment areas based on travel time, because usually travel time increases when travel distance increases. To assign the population to the closest stroke center, data is used where the population is divided in an 1 by 1 km^2 grid. The calculation of the catchment area is done in the following way. First we calculate for every population grid point what the five closest stroke centers are based on haversine distance. Then the distance over the road is calculated between the population point and 5 closest stroke centers for all population points as the haversine distance from the population point to the closest point in the road network plus the distance over the road network plus the haversine distance from the facility to the closest road network point. Corrections are made to the calculated distances. In case the haversine distance is extremely small or in case the distance from the population grid point to the closest road is greater than the haversine distance between the population point and facility or in case no road distance could be calculated we use haversine distance instead of road distance. Next, each population point is assigned to the stroke center with the smallest corrected distance. Parts of the distance calculating code where already developed for earlier projects in the GPBP tool and could be reused for this purpose [7]. Figure 3.2 gives a illustration of the created catchment areas. After assigning every population point to a stroke center, the total number of people in the catchment area of each stroke center can be calculated. An overview of the number of people associated with each stroke center is given in figure 3.3.

Calculation of λ

Since the number of people in the catchment area of each stroke center is large and the probability of having a stroke is small, the number of strokes in a facilities catchment area

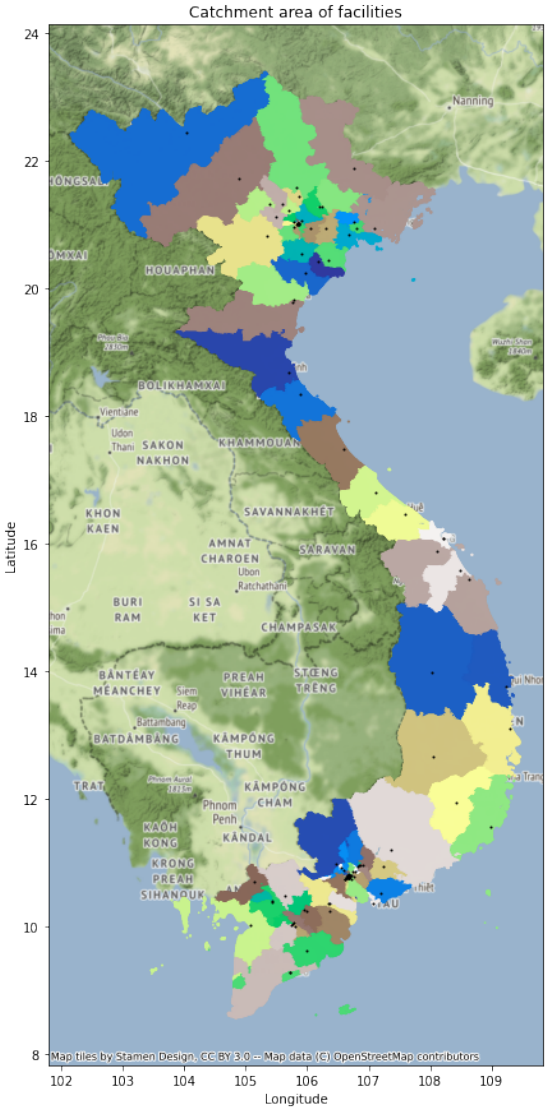

FIGURE 3.2: Catchment areas of stroke facilities (black dots) in Vietnam

will be approximately Poisson distributed with rate $\lambda_i = NrPeople_i * p$, with $NrPeople_i$ the number of people in catchment area of stroke facility i and p the probability that a randomly chosen person has a stroke within a certain time frame. We assume p to be the same for all catchment areas, although this might not hold in reality because of differences in socio-demographics. According to Nguyen et al. [2] the incidence of stroke in Vietnam was 254.78/100,000 person-years in 2010. Assuming the probability of having a stroke is the same nowadays, the daily probability of having a stroke can be calculated as $1 - (1 - 254.78/100,000)^{1/365} \approx 6.99 \times 10^{-6}$. Multiplying this probability with the number of people within the catchment area of the stroke center results in the following values for λ (ordered based on their size): [21.1, 20.1, 18.7, 17.0 ... 0.4, 0.4, 0.4,0.3].

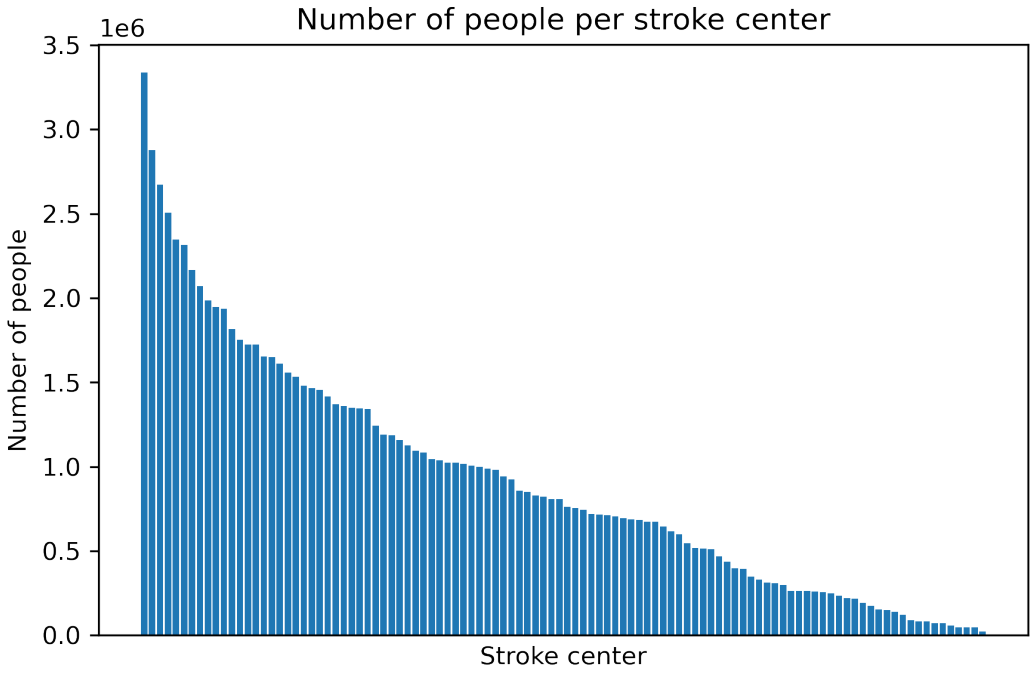

FIGURE 3.3: Number of people in catchment area of stroke center

Calculation of μ

To make an estimation of μ, i.e., the number of people that can be treated on average per bed per day, information is needed about the average length of stay of a patient. There is only information available over the median, quartile 1 and quartile 3 of the length of stay split up in patients that are discharged and deceased. Nguyen et al. [2, p.120] report that the length of stay in case a patient is discharged has a median of 3 and an interquartile range of 1-6, the length of stay in case a patient is deceased has a median of 4 and an interquartile range of 3-6. Furthermore, it is known that 96.9% of the patients are discharged and 3.1% of patients are deceased. An exponential distribution is fitted through the median and interquartile range data given the assumption that the length of stay follows an exponential distribution. This results in the best fitted exponential distribution with corresponding value of average length of stay. Because only one value is needed for μ for both the discharged and deceased patients, it is estimated that their joint distribution has a weighted median of $3 * 0.969 + 4 * 0.031 = 3.031$ and a first quartile of $1 * 0.969 + 3 * 0.031 = 1.062$ and a third interquartile of $6 * 0.969 + 6 * 0.031 = 6$. The best fitted distribution is considered to be the exponential distribution with the minimum of $|distributionmedian - 3.031| + |distributionfirstquartile - 1.062| + |distributionthirdquartile - 6|$. It turns out that the best fitted distribution is a exponential distribution with mean length of stay of 4.33. Therefore, a μ of $1/4.33 \approx 0.231$ will be used in the MA algorithm.

Current number of beds at each stroke center

The process of data gathering for the number of beds at the stroke centers in Vietnam is not yet completed. Therefore some dummy data is generated to test the algorithm. The dummy data is created in such a way that the total number of beds would be sufficient to serve the population, meaning that ρ is smaller than 1 for the whole population. Furthermore we made sure that in general the more people where assigned to the stroke center, a *proportional* number of beds where assigned, but with some random variation. This resulted in the biggest stroke center having a capacity of 146 beds, and the smallest stroke center having a capacity of 2 beds.

Budget

Since we have no information on the budget, we will look how the average waiting probability and the average waiting time decrease for different budgets and we will give more detailed results for an arbitrary budget of 500 beds.

3.1.3 MA based on queueing formulas and travel time

A disadvantage of the MA method is that the method does not take into account the travel time to other facilities in case the closest facility is occupied. This means that in reality it is less bad if no place is available at your closest facility, if the second closest facility is available to you within relative short travel time, compared to when the travel time to the next facility is long. A way of accounting for the extra travel time in the MA model is to multiply the probability a patient has to wait at a facility with the extra travel time to the next facility. In that case the MA method minimizes the extra travel time of patients. Unfortunately this method has some drawbacks. First of all, in this method is only accounted for the extra travel time to the next facility, but it in case also the second closest facility has no place available, the patients has to go to the next facility, but the extra travel time in case the two closest facilities are occupied is not taken into account in this model. In the MA method we assume that patients that cannot be treated immediately, start queueing. This assumption does no longer apply, since we now assume that patients go to the next facility. Therefore it will be more suitable to assume no queue at the facility and that patients that cannot be treated immediately are lost, meaning an M/M/s/s queueing type is assumed. The probability that a arriving patient cannot be treated (blocking probability) can then be calculated using the Erlang B formula. A disadvantage is that patients that were blocked are not accounted for as extra incoming patients in the next facility. The last

problem is that this way of distributing capacity would lead to an underestimation of capacity for a group of facilities close to each other. For example in the Vietnam case strong differences are observed in the rural areas with small number of facilities, with high travel times between the stroke centers and the cities, with multiple facilities close to each other. The MA model based on extra travel time will lead to the facilities in the cities having relative higher blocking probabilities, compared to rural areas. This means that the facilities in the cities will get more incoming patients from nearby facilities and that there is a higher change that patients in the city have to visit more than 2 facilities before they can be treated. For both these effects is not accounted in the model, meaning that the model will make an underestimation of the needed capacity in the city. Therefore, it is concluded that this method would not be suitable to allocate capacity.

3.2 Simulation

The second approach to the capacity allocation problem does no longer assume that people can only be served at the closest facility. Instead it is assumed that people first travel to the closest facility. If the available capacity is sufficient, the person will be treated at the facility, in case the capacity is insufficient, the person will start travelling to the next facility and this process will continue until a facility is found with enough capacity. The goal of this second approach is to calculate the optimal allocation with the smallest average travel time. The first step in this process is to create a simulation to calculate the average travel time for a specific allocation of capacity and the next step is to train a predictive model that can predict the average travel time and embed this model in a optimization model. In this chapter the first step, building the simulation, will be discussed.

A flowchart of the simulation is presented in fig. 3.4. The simulation requires the following parameters: *pop*, a dataframe of the population with the number of people per grid point,$warm\_up$, the duration of the warm up period in number of people for whom we want to know the travel time, $length$, the duration of the simulation (excluding warm up period) in number of people for who we want to know the travel time, $t_{interarrival}$, the average interarrival time, $t_{treatment}$, the average treatment time, *capacity*, an array of the number of servers at each facility, f_{pop}, array with the closest facility in travel time for each grid point in *pop*, tt_{pop}, array with the travel time to the closest facility in travel time for each grid point in *pop*, f_f, dataframe with travel time from each facility to every other facility, NrF, the number of facilities and the stopping criteria *stop*. By default the stopping criteria is the number of patients for which the travel time need to be calculated, namely $length+warm\_up$. Alternatively the stopping criteria can be set

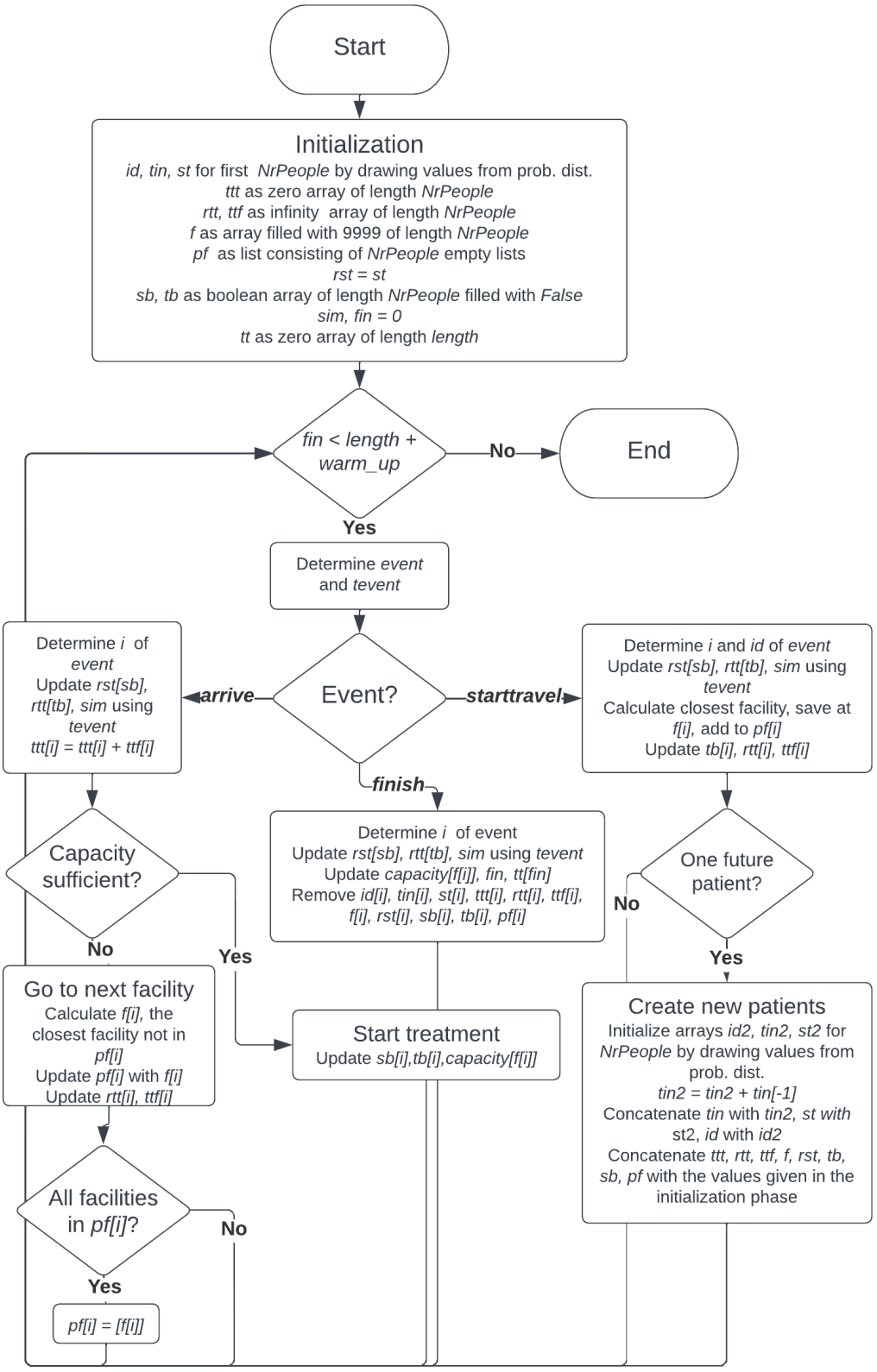

FIGURE 3.4: Flowchart of simulation

to the 95% confidence interval width around the average travel time. In that case the desired confidence interval width should be given as well.

The simulation starts with an initialization phase. First arrays of $NrPeople$ future patients are created, namely: id, array with index of grid point where the patient lives, tin, array with time the patients get a stroke, st, array with the treatment time of patients. The values in tin are drawn from an exponential distribution with the $t_{interarrival}$ as average. The values in st are drawn from an exponential distribution using $t_{treatment}$ as average. There are also arrays initialized for the $NrPeople$ future patients: ttt, total travel times, rtt, remaining travel times to next facility, f, the facility the patients are travelling towards, ttf, the travel times to the next facility, pf previously visited facilities for all patients, rst, remaining treatment times, sb, boolean if the patients are currently in treatment and tb, boolean if the patients are travelling.

The three events in the simulation are $starttravel$, a person that starts travelling towards a facility, $arrive$, a person that arrives at a facility, $finish$, a person for which the treatment at the facility is completed. The simulation will be running until the stopping criteria is met. The flow chart is drawn with the stopping criteria that the travel times of $warm\_up + length$ number of people are gathered. Alternatively, the confidence interval width can be used as the stopping criteria. At each iteration first is determined what the next event is that will happen and the time until this event will happen $tevent$.

If the next event is a person that starts travelling, we first determine the index of the person i that starts travelling and the index of grid point where this person lives id. Next, the simulation time (sim), the arrays with remaining service time (rst) and the remaining travel time (rtt) are updated for all people. Then the facility the person will travel towards is calculated and added to this list of previously visited facilities pf and the time to reach this facility (ttf) also saved at rtt for this person. Followed with setting the boolean if a person is travelling (tb) for person i to $True$. If the length of the list of previously generated people that will start travelling in the future is decreased to 1, then data for $NrPeople$ new persons is generated.

If the next event is a person arriving at a facility, we start with determining i. Next, sim, rst and rtt of person i are updated with $tevent$. The total travel time ttt of person i is incremented with the travel time to reach this facility. If the capacity at the facility is sufficient, then person i will start treatment at this facility, leading to a decrease in capacity at the corresponding facility of one. The treatment boolean sb is set to $True$ and travel bool tb is set to $False$. In case the capacity at the facility is insufficient, person i will start travelling to the next facility f, which is the closest facility that is not visited before and add this facility is added to the previous visited facilities list. In case all facilities are in the previous visited facilities list, all facilities in the list are

removed, except f. Finally the travel time to the next facility and remaining travel time are updated.

If the next event is a patient has completed its treatment, first i is determined of the person that has completed its treatment. Next, sim, rst and rtt are updated for all people currently in treatment or travelling respectively. The *capacity* of facility f where person i finishes treatment increases by one and number of finished people fin increases with one. The travel time of person that finishes is saved at the travel time array tt. At last, the data of the finished person of is removed from the arrays id, tin, st, ttt, rtt, ttf, f, rst, sb, tb, pf.

Running the simulation results in an array with of travel times of patients. Besides that, the average travel time and corresponding confidence interval, standard deviation of the travel times and a list with the visited facilities per patient is given. The average and confidence interval are calculated by discarding the warm up period. The confidence interval is calculated using the batch means method explained in Section 2.4.

Since we do not know the length of the warm up period before running the simulation, the warm up period needs to be determined first. To determine the warm-up period the simulation can be executed for a number of times. Then the average travel time and confidence interval can be plotted against by the number of people used to calculate these values. The length of the warm up period can be detected visually as the period before the average is more or less a straight line. After that a simulation can be run until the desired confidence level and with discarding the warm up period the average and confidence interval can be calculated using the batch means method explained in Section 2.4.

Vietnam case study simulation parameters

For the Vietnam case study the following parameters are used in the simulation:

pop	A dataset where the population of Vietnam is divided over a grid of 1 by 1 km^2 was used.
length	A run length of 15,000,000 people is used, the analysis of what run length is required to generate valid results is given in Section 4.2.
$t_{interarrival}$	The average interarrival time used is 1.47×10^{-3} day, this is based on the probability of a stroke of 254.78/100000 person-years in 2010 as stated by Nguyen, T [2] and recalculating this to time between strokes in days using the total population of Vietnam of approximately 97 million. Furthermore it is assumed that the interarrival time is exponential distributed.
$t_{treatment}$	The average treatment time is 4.33, as explained in Section 3.1.2. It is assumed that the treatment time is exponential distributed.
capacity	Since there was no information available on the current number of beds at each facility, a fictitious dataset was created and this dataset with current number of beds was added to the generated allocations to obtain the capacity.
f_{pop}, tt_{pop}	To get f_{pop} and tt_{pop}, first the distance over the road from every grid point of the population to every facility is calculated. This distance is converted to travel time by assuming an average speed of 70 km/u. Next, for each population point the facility with the shortest travel time is selected and saved as f_{pop} and the corresponding shortest travel time is saved at tt_{pop}.
f_f	First, the road distance from every facility to every other facility is calculated. Then the distance is converted to travel time with using an average speed of 70 km/u and this is saved at f_f.
NrF	There are 106 stroke centers in Vietnam.

3.3 OptiCL

For this second type of model the goal is to calculate the allocation of capacity resulting in minimal average travel time. This model can be formulated as a mixed-integer

optimization model in the following way:

$$\text{Minimize} \quad t \tag{3.1a}$$

$$\text{Subject to} \quad \sum_{f \in F} x_f \leq B \tag{3.1b}$$

$$t \geq 0 \tag{3.1c}$$

$$t = \hat{h}(x) \tag{3.1d}$$

$$x_f \geq 0, integer \qquad \forall f \in F \tag{3.1e}$$

where: t = average travel time
B = budget of additional capacity
x_f = capacity added to facility $f \in F$
$\hat{h}(x)$ = predictive model trained on added capacity to each facility x.

The goal of this model is to minimize the average travel time (3.1a). Constraint 3.1b ensures that the budget of available additional capacity is not exceeded. Constraint 3.1c restricts the average travel time to non-negative. Constraint 3.1d describes the predictive model for the average travel time. This predictive model will be trained based on a dataset with allocations of capacity to the facilities as features and average travel time as predictive value. These average travel times are calculated using the simulation from Section 3.2. Finally, constraint 3.1e is added to set the decision variable x_f to non-negative and integer.

Vietnam case study OptiCL parameters

There are three steps to determine the best allocation of capacity for Vietnam using OptiCL. First a dataset of allocations is created. Secondly, the simulations for the allocations are executed to determine the average travel time (see Section 3.2). As a final step, an optimization model with constrained learning is built in which the average travel time is learned using a machine learning model. For each of the steps the parameters used are discussed over here.

When running simulations with different allocations of beds to the stroke centers, with the same number of total beds, it turned out that the differences in averages travel time between these simulations are extremely small. This means it is necessary to run these simulations until a very small confidence level is reached, leading to long simulation runtime. This was not feasible since many simulations where required to be executed, therefore a small change was made in the set up of the experiment. The set up of the experiment was changed in the following way. First, the 106 stroke centers are divided in 10 categories, with 10 stroke centers in the first 4 categories and 11 stroke centers in the

last 6 categories. The first category exists of stroke centers with the highest probability that a patient has to queue in case it is assumed that people have only access to the closest stroke center. With every next category the probability of waiting decreases and the last category has the stroke centers in the last category have the smallest probability of waiting. The goal is to allocate the budget of beds in such a way to the categories that the average travel time of people is minimized. When x beds are assigned to a category, this means that every facility in this category receives x beds.

The first step is to generate a dataset of 200 allocations of beds to the categories where the total number of beds of each allocation is around the budget of 1000 beds and where the sampled allocations are space filling. To achieve this goal, each category is given a range of possible added capacity. The upper limit of the range is based on the total number of beds added to each category according to the MA algorithm plus a margin of 200 beds divided by the number of facilities in the category. The lower limit is set to zero. Table 3.1 gives an overview of the upper and lower limit of each category.

TABLE 3.1: Upper and lower limit of extra capacity for each category.

Category	**Facilities in category**	**Range**
1	60, 12, 76, 37, 90, 32, 27, 95, 24, 23	0, 40
2	16, 101, 45, 4, 35, 30, 54, 3, 14, 72	0, 39
3	18, 87, 96, 38, 70, 84, 64, 78, 6, 73	0, 33
4	102, 94, 68, 79, 55, 91, 66, 80, 69, 93	0, 29
5	44, 62, 40, 10, 75, 42, 74, 33, 9, 21, 5	0, 26
6	20, 56, 65, 0, 2, 50, 7, 41, 81, 77, 82	0, 26
7	100, 86, 13, 1, 57, 71, 59, 26, 51, 104, 61	0, 26
8	31, 22, 43, 53, 8, 67, 11, 63, 97, 92, 19	0, 26
9	52, 99, 25, 34, 39, 46, 49, 105, 85, 36, 89	0, 22
10	103, 88, 17, 83, 48, 28, 47, 15, 29, 98, 58	0, 19

To generate an allocation a random uniform number is picked within the range of each category. When the total number of beds added to the facilities is between 900 and 1100, the allocation is accepted and added to the allocation dataset. These steps are repeated until a dataset of 200 allocations is generated. The above process is repeated for 20,000 times and then the best allocation dataset is selected as the dataset with the highest L2 distance between the allocations.

The mathematical model for the categorical setup of the Vietnam case can be defined as follows:

$$\text{Minimize} \quad t \tag{3.2a}$$

$$\text{Subject to} \quad \sum_{c \in C} f_c x_c \leq B \tag{3.2b}$$

$$t \geq 0 \tag{3.2c}$$

$$t = \hat{h}(x) \tag{3.2d}$$

$$x_c \geq 0, integer \qquad \forall c \in C \tag{3.2e}$$

where: t = average travel time
B = budget of additional number of beds
f_c = number of stroke centers in category $c \in C$
x_c = number of beds added to each stroke center in category $c \in C$
$\hat{h}(x)$ = predictive model to predict the average travel time trained on simulation data of added beds to every stroke center x.

For the predictive model we will use six types of models: linear model, gradient boosting machines (gbm), decision tree (cart) and random forest (rf) and neural network (mlp), implemented as a multilayer perceptron with rectified linear activation function. To determine the hyperparameters, a grid search was done using the following values.

TABLE 3.2: Hyperparameters for grid search predictive models

Model	**Hyperparameters**
mlp	hidden layer sizes: [(10,), (20,), (50,), (100,)]
linear	alpha: [0.1, 1, 10, 100, 1000], l1 ratio: [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9]
gbm	learning rate: [0.01, 0.025, 0.05, 0.075, 0.1, 0.15, 0.2], maximum depth: [2, 3, 4, 5], number of estimators: 20
cart	maximum depth: [3, 4, 5, 6, 7, 8, 9, 10], minimum samples leaf: [0.02, 0.04, 0.06], maximum features: [0.4, 0.6, 0.8, 1.0]
rf	number of estimators: [10, 25], maximum features: ['auto'], maximum depth: [2, 3, 4]
svm	Regularization parameter: [0.1, 1, 10, 100]

3.4 MA based on travel time

As an alternative to the OptiCL model, a heuristic is proposed to calculate the optimal allocation of capacity given a certain budget of capacity. The basic idea of this heuristic is to add capacity to the facilities one-by-one, with adding the bed to the facility that results in the greatest decrease in average travel time.

The first step in the heuristic is to calculate the average travel time of the current capacity using the simulation of Section 3.2. Next, the average travel time and the decrease in travel time is calculated when one extra bed is added to the facility, and this is repeated for all facilities. Then one bed is added to the facility with the greatest decrease in average travel time. The next steps will be repeated until the budget of capacity is exhausted. The average travel time is calculated with one extra bed added for each of the facilities the corresponding decrease in average travel time is determined. Because the decrease in travel time is smaller for adding the $x + 1$ extra bed to a facility compared to adding the xth extra bed, computational resources can be used more efficient by not calculating the travel time with one extra bed for all facilities. The (decrease in) travel time does not need to be calculated for a facility if the decrease in average travel time of adding one extra bed to one of the other already calculated facilities is greater than the decrease in average travel time for adding one extra bed in a previous iteration. A bed will be added to a facility were an additional bed leads to the greatest decrease in average travel time. This part will be repeated until the budget of capacity is fully used.

The advantage of this heuristic is that the number of simulations is at maximum a multiplication of the number of facilities and budget. One of the disadvantages of this method is that it is not guaranteed that the resulting allocation is optimal. Another disadvantage is that the required simulation run length can become excessively long, because the differences in travel time between adding one extra unit of capacity at facility a versus adding one extra unit of capacity at facility b can be extremely small, meaning that a very small confidence level is required to obtain valuable results. For example, for the Vietnam case one can image that the difference of adding one bed to stroke center a versus adding one bed to facility b on a total number of beds of approximately 4,000 is very small, leading to extremely long simulation run length to obtain an average travel time with small enough confidence interval. For this reason it was not possible to use this heuristic to find a good allocation of beds for the Vietnam case study.

Chapter 4

Experiments and results

This chapter presents the results of the models. In subsequential order the results are presented for the MA algorithm, the simulation and for allocating capacity using the OptiCL.

4.1 MA

This section discusses the results of the MA algorithm on the Vietnam case. Since there is no information available on the budget for adding beds the first step is to provide insight in the relationship between the budget of beds and the waiting time and waiting probability. Secondly, more insight is provided to which stroke centers the beds are added, given a budget of 500 beds. Finally, the runtime, exactness and scalability of the MA algorithm are given.

4.1.1 Relationship between capacity and waiting time and waiting probability

For the Vietnam case the relationship between adding beds and the probability of waiting is presented in figure 4.1. The extra beds are distributed over the stroke centers using the MA algorithm to minimize probability of waiting (blue line) and using the MA algorithm to minimize the average waiting time (orange line). As expected, the probability of waiting is a non-increasing convex function of the number of beds in case MA is used to minimize the probability of waiting. This means that adding the first extra bed will have the biggest impact on the probability of waiting and that with every extra bed added, the reduction in waiting probability will be smaller. When using MA to minimize the average waiting time, the decrease in probability of waiting is smaller, but very close

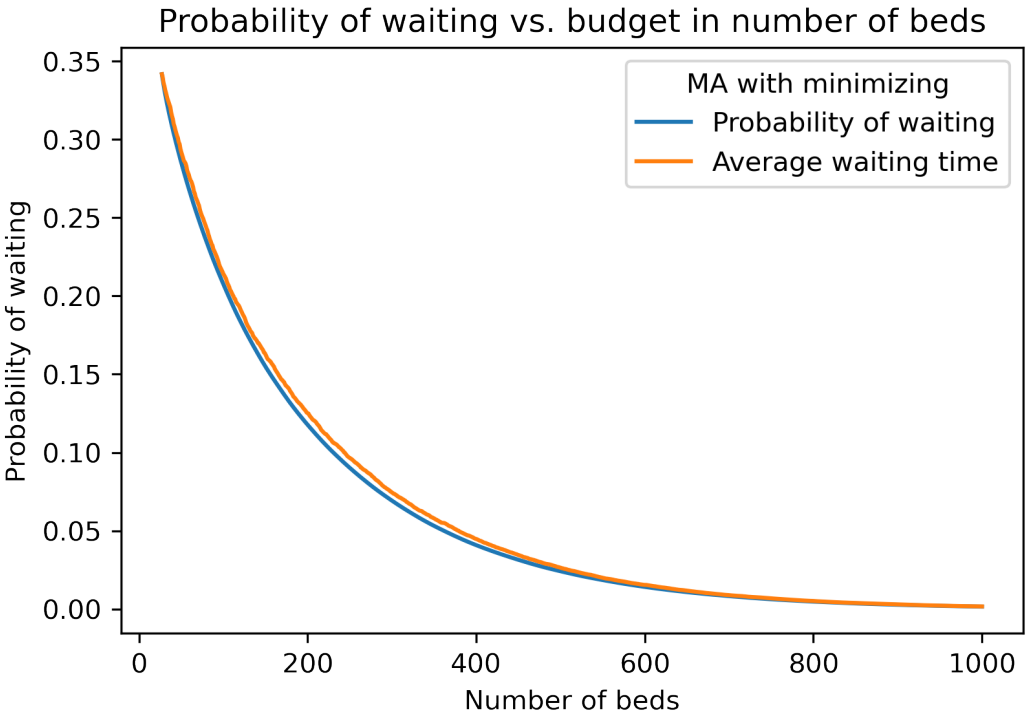

FIGURE 4.1: Relationship between probability of waiting and the budget in number of beds for the optimal allocation of beds when allocating beds using MA with minimizing the probability of waiting and MA with minimizing the average waiting time.

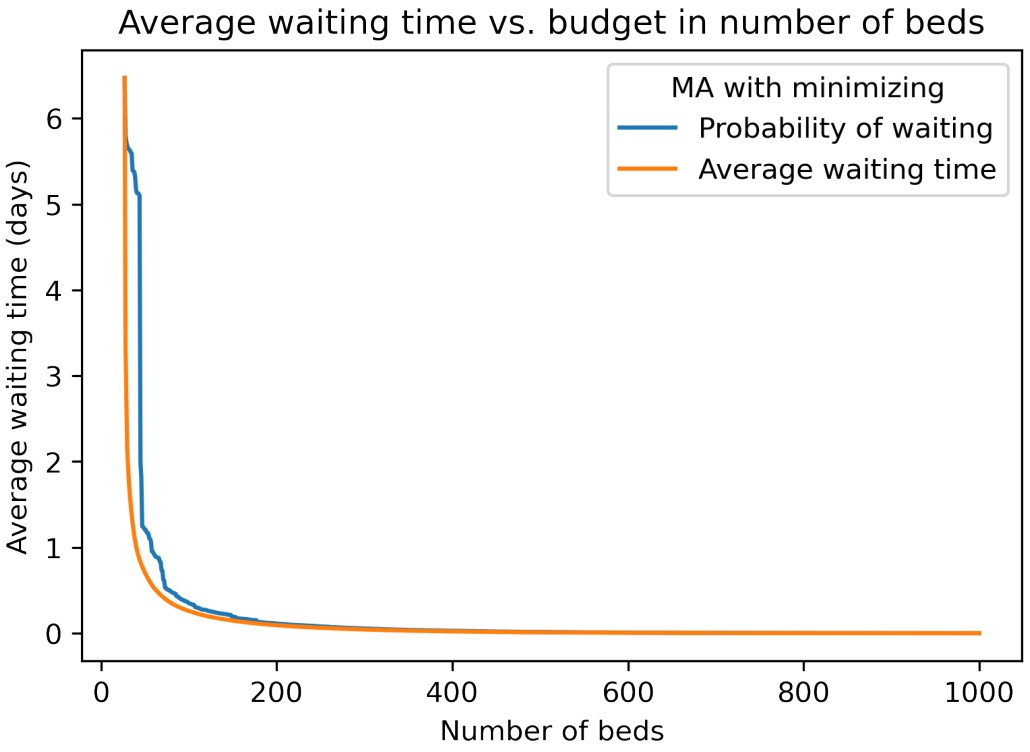

FIGURE 4.2: Relationship between average waiting time and the budget in number of beds for the optimal allocation of beds when allocating beds using MA with minimizing the probability of waiting and MA with minimizing the average waiting time.

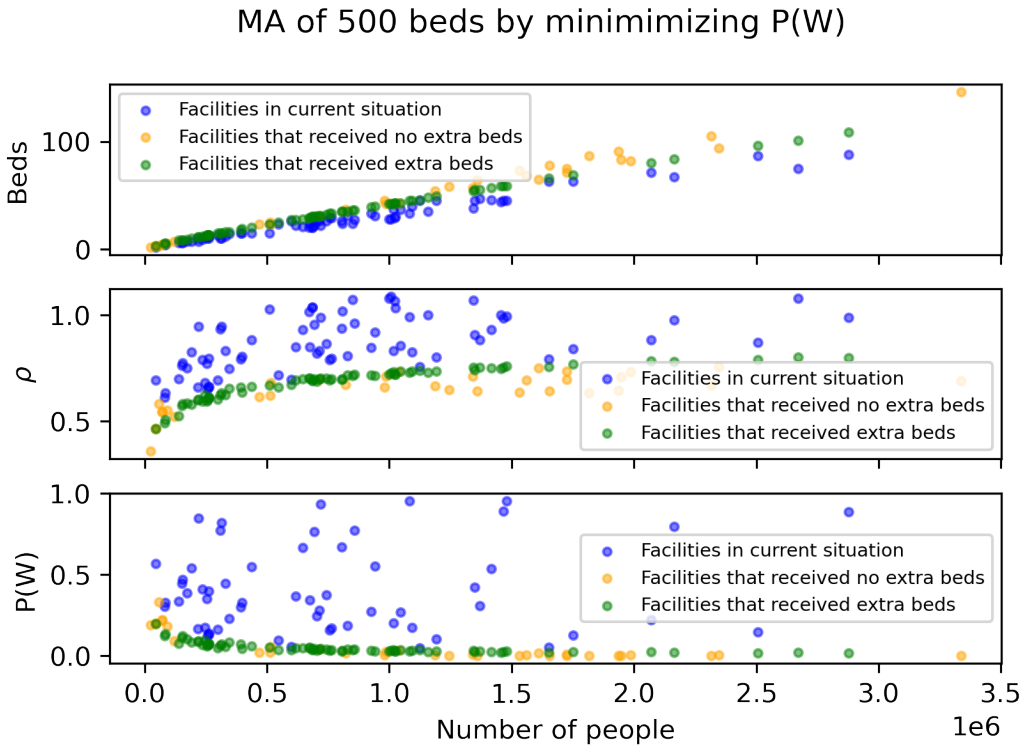

FIGURE 4.3: MA of 500 beds with minimizing probability of waiting P(W). Note that for some stroke facilities the probability of waiting for the current number of beds could not be calculated because $\rho \geq 1$.

compared to the decrease in probability of waiting when MA was used to minimize the probability of waiting. Note that for the first 26 extra beds the average waiting probability could not be calculated, because for some of the stroke facilities $\rho > 1$.

Similar results are found for the relationship between optimal allocating extra beds and the average waiting time, see figure 4.2. In case MA is used to find the optimal allocation that minimizes the average waiting time, as expected the average waiting time is a non-increasing convex function of the number of beds. For the first 26 extra beds the average waiting time could not be calculated, due to $\rho > 1$ for some of the stroke centers. When using MA to minimize the probability of waiting, the decrease in average waiting time was less compared to MA with minimizing average waiting time for the first 200 added beds, after that, the average waiting time was almost equal for minimizing the average waiting time or probability of waiting in the MA algorithm.

4.1.2 Results for adding 500 beds

In order to provide additional insight in the working of the MA algorithm, the current and added number of beds, ρ and the probability of waiting or average waiting time for each stroke center for applying the MA algorithm with a budget of 500 beds. The

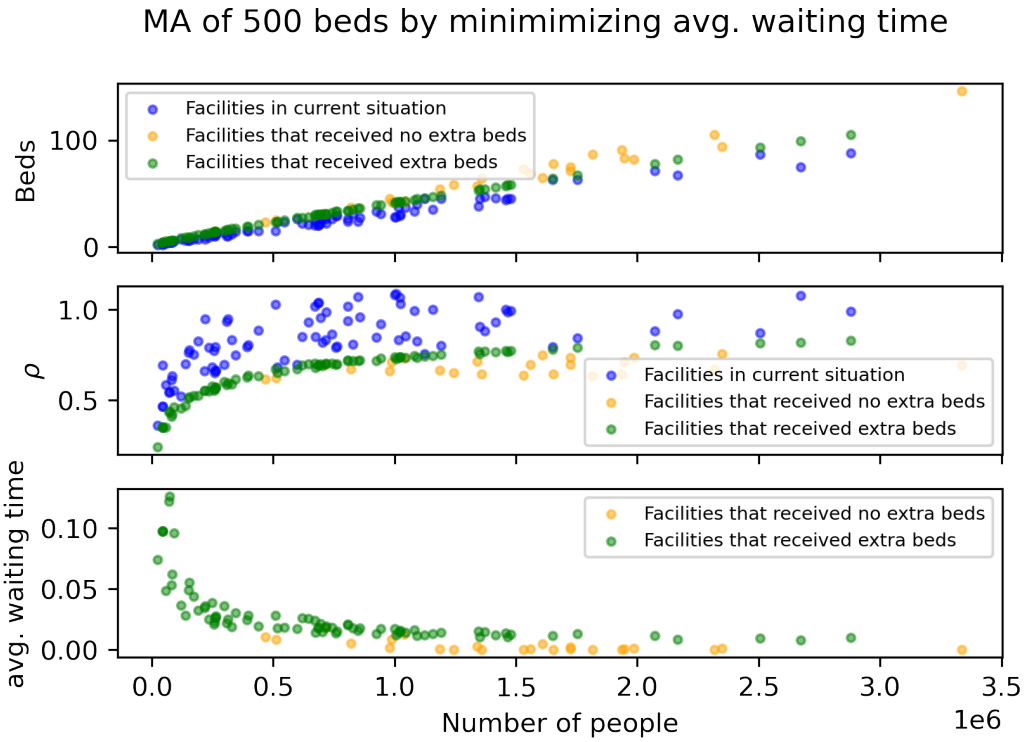

FIGURE 4.4: MA of 500 beds by minimizing average waiting time. For the waiting time graph the waiting time for stroke centers with only the current number of beds was omitted, since these values where extremely high (up to 15 days) and in case $\rho \geq 1$ and time $\longrightarrow \infty$ the waiting time $\longrightarrow \infty$.

results for MA with minimizing the probability of waiting P(W) can be viewed in figure 4.3 and the results for MA with minimizing the average waiting time are given in 4.4.

Based on figure 4.3 it can be concluded that there is approximately a linear relationship between the number of people in the catchment area and the stroke centers that got beds allocated. Stroke centers that had an already relatively high number of beds compared to the number of people in the catchment area did not get additional beds. Besides that, it can be observed that for stroke centers that received beds ρ increases with the number of people in the catchment area of the stroke center. The probability of waiting for stroke centers that did receive extra beds decreases when the number of people in the catchment area increases. It might be surprising to see that the probability of waiting is higher for stroke centers with smaller number of people while ρ is small for stroke centers with smaller number of people. This can be explained by looking at the inputs of the probability of waiting formula, namely the traffic intensity (closely related to ρ) and the number of beds. The stroke centers with a small number of people in the catchment area have a smaller number of beds, which has an increasing effect on the probability of waiting, while lower values of traffic intensity will have a decreasing effect on the probability of waiting. It turns out that the effect of having less beds is greater,

resulting in higher probability of waiting for a small number of people in the catchment area. Similar results are found for MA with minimizing average waiting time. It is numerically verified that these results hold not only for a budget of 500 beds, but also for a wider range of budgets.

Besides the differences in the number of beds, ρ and average waiting time or P(W), we looked at the differences in allocation of beds between minimizing on average waiting time and probability of waiting. The allocation of beds are close to each other, with a maximum difference of 4 beds per stroke center and an average difference of 0.64 beds per stroke center.

4.1.3 Runtime, scalability and exactness of MA algorithm

Next, the runtime, whether the algorithm is exact and if it is scalable is examined to see how usefull the algorithm is in practice. The runtime of the algorithm is short, with an average of 170 ms and standard deviation of 1.58 ms for adding 500 beds to the Vietnam case using the average waiting time and 177 ms on average with a standard deviation of 2.65 for using the probability of waiting. Both runtime measures where done using an AMD Ryzen 5 4500U 2.38 GHz CPU, 16 GB RAM. The run time of the algorithm scales linearly in the number of facilities and budget of additional capacity. The result of the algorithm is proven to be exact by Weber [15]. This means that the MA algorithm is a suitable algorithm in case the assumption of no forwarding between facilities holds. When this assumption does not hold, we need to simulate the behaviour of patients forwarding between facilities and then we can use OptiCL to find the optimal allocation of beds, for which the results are given in the next section.

4.2 Simulation

In this section the influence of the input parameters in the simulation on the average travel time will be examined and the simulation results for the Vietnam Case are presented.

4.2.1 Influence simulation input parameters on average travel time

To provide better insight in the simulation results dependency on input parameters, the influence of the average treatment time and the population size on the simulation output is examined. The simulation output measures examined are the average travel time, standard deviation, the percentage of patients treated at the first, second or at

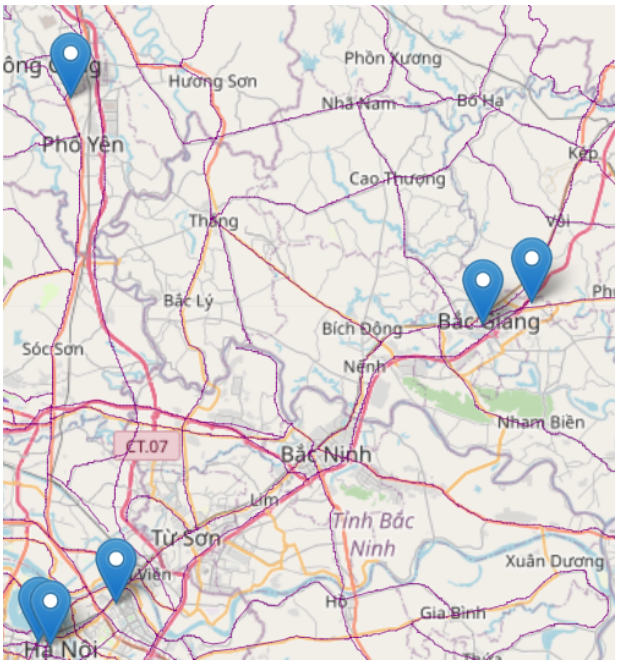

FIGURE 4.5: Map of testdataset. The markers show the location of the 6 stroke facilities in the testset.

least the third facility, the probability for each facility to being forwarded to the next facility and the simulation run length needed to obtain the desired confidence interval. The results are obtained on a small subset of the Vietnam case, shown in fig. 4.5. The two facilities located in the high densely populated city of Ha Noi (left under) are facility f_0 and f_1, which have 18 and 28 beds, respectively. The stroke center close to Ha Noi is facility f_2 with 47 beds. The stroke center in the left upper corner f_3 has 10 beds. The stroke facility most on the right is facility f_5 with 5 beds and the facility close to to f_5 is f_4 with 58 beds. The number of beds at facilities are made up, as in the current status of the project there is no information available on the number of beds. The facility that is closest for the highest number of people is facility f_2 with it being the closest stroke center for 1.58×10^6 people. The facility that is closest for the smallest number of people is f_0 with it being the closest facility for 2.50×10^5 people. For the other simulation input parameters, the same values are used as in the Vietnam test case described in Section 3.2.

The influence of population growth and decline on the simulation output is tested first. Five scenarios are tested: the base scenario with the normal population size of approximately 4 million people, a population growth of 3% and 6% and population decline of 3% and 6%. It is assumed that the percentage increase or decrease of the population is evenly spread across the map. First, the warm up period is analysed, see fig. 4.6. From this figure, it can be concluded that a warm up period of 7500 is enough to remove the initialization bias. Based on this figure it can also be concluded that a confidence level

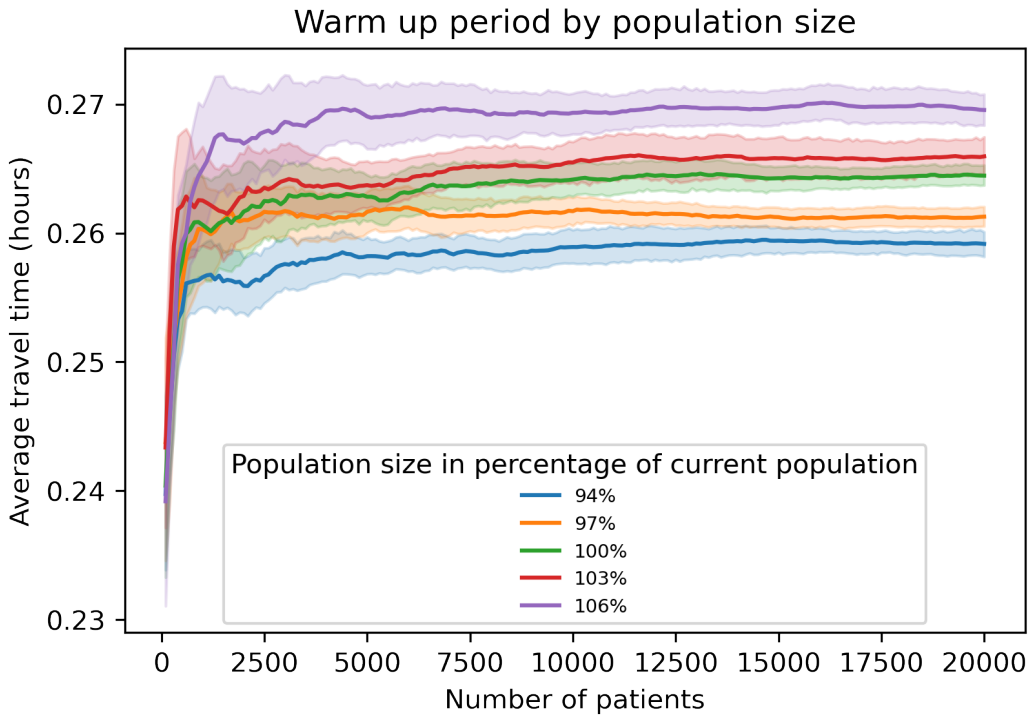

FIGURE 4.6: Warm up period for different population sizes

of 0.001 hours is a suitable choice for the desired confidence level. From each scenario a simulation was executed until the desired confidence level is reached and with the determined warm up period. The results are given in table 4.1.

Based on this table it can be concluded that average travel time increases with population size, with the highest growth scenario having 0.012 higher travel time compared to the scenario with the most population decline. The standard deviation increases with the population size. The percentage of people that are treated in the first decreases and the number of people visiting 2 or at least 3 facilities before receiving treatment increases. An increase can be seen in the probability of being forwarded at each facility when the population increases. The length of the simulation of the of the scenarios with 94, 97 and 100% of the population size are similar, but simulation length increases considerable for the scenarios with population growth, with the highest growth scenario having almost 6 times longer simulation length than current population scenario.

A similar analysis is done for the treatment time. Five scenarios were considered, with average treatment time of 4.03, 4.18, 4.33 (original treatment time), 4.48, 4.63. When analyzing the warm up period in fig. 4.7 it can be concluded that a warm up period of 9000 is enough to remove the initialization bias, and a desired confidence level of 0.001 being enough to find the average travel times with enough security. Running the simulations until the required confidence level and with the analyzed warm up period

TABLE 4.1: Average travel time of patients in hours (Avg. travel time), standard deviation in hours (Sd), percentage of patients receiving treatment at first facility they visited (treated facility 1), percentage of patients receiving treatment at second facility they visited (treated facility 2), percentage of patients receiving treatment at more than the 2^{nd} facility they visited (treated facility 3+), probability that patients are forwarded to the next facility for each facility f_0 to f_5 (Prob forward $f1$ - $f5$) and simulation run length in number of finished people (length simulation) for different population sizes in percentage of original population (94-106). All simulation results are obtained by running the simulations until a confidence level of 0.001 hour was reached and with a warm up period of 7500.

	94	**97**	**100**	**103**	**106**
Avg. travel time	0.259	0.261	0.264	0.267	0.271
Sd	0.195	0.197	0.200	0.207	0.225
treated facility 1	89.8%	88.7%	87.5%	86.3%	85.0%
treated facility 2	9.7%	10.6%	11.5%	12.3%	13.2%
treated facility 3+	0.5%	0.7%	0.9%	1.3%	1.8%
Prob forward f_0	3.6×10^{-3}	7.9×10^{-3}	1.2×10^{-2}	2.8×10^{-2}	4.7×10^{-2}
Prob forward f_1	0.02	0.03	0.04	0.07	0.09
Prob forward f_2	0.11	0.13	0.15	0.17	0.20
Prob forward f_3	0.23	0.25	0.26	0.28	0.29
Prob forward f_4	1.5×10^{-4}	2.7×10^{-4}	3.8×10^{-4}	1.2×10^{-3}	2.4×10^{-3}
Prob forward f_5	0.48	0.49	0.51	0.52	0.53
Length simulation	1.18×10^6	9.98×10^5	9.68×10^5	2.57×10^6	5.42×10^6

TABLE 4.2: Avg. travel time of patients in hours (Average travel time), standard deviation in hours (Sd), percentage of patients receiving treatment at first facility they visited (treated facility 1), percentage of patients receiving treatment at second facility they visited (treated facility 1), percentage of patients receiving treatment at more than the 2^{nd} facility they visited (treated facility 3+), probability that patients are forwarded to the next facility for each facility f_0 to f_5 (Prob forward $f1$ - $f5$) and simulation run length in number of finished people (length simulation) for treatment times in days (4.03-4.63). All simulation results are obtained by running the simulations until a confidence level of 0.001 hour was reached and with a warm up period of 9000.

	4.03	**4.18**	**4.33**	**4.48**	**4.63**
Avg. travel time	0.258	0.261	0.264	0.267	0.273
Sd	0.19	0.20	0.20	0.21	0.26
treated facility 1	90.1%	88.8%	87.5%	86.0%	84.6%
treated facility 2	9.5%	10.5%	11.5%	12.6%	13.4%
treated facility 3+	0.4%	0.7%	1.0%	1.4%	2.0%
Prob forward f_0	1.6×10^{-3}	7.1×10^{-3}	1.4×10^{-2}	3.0×10^{-2}	5.5×10^{-2}
Prob forward f_1	0.01	0.03	0.05	0.07	0.10
Prob forward f_2	0.11	0.13	0.15	0.18	0.20
Prob forward f_3	0.23	0.25	0.26	0.28	0.30
Prob forward f_4	1.3×10^{-4}	2.9×10^{-4}	6.9×10^{-4}	1.5×10^{-3}	3.1×10^{-3}
Prob forward f_5	0.48	0.49	0.51	0.52	0.54
Length simulation	9.49×10^5	1.66×10^6	1.24×10^6	1.84×10^6	2.00×10^7

leads to the results in table 4.2. Based on this table it can be concluded that average travel time and the standard deviation increase with increasing treatment time, the percentage of patients treated at the first visited facility decrease and percentage of

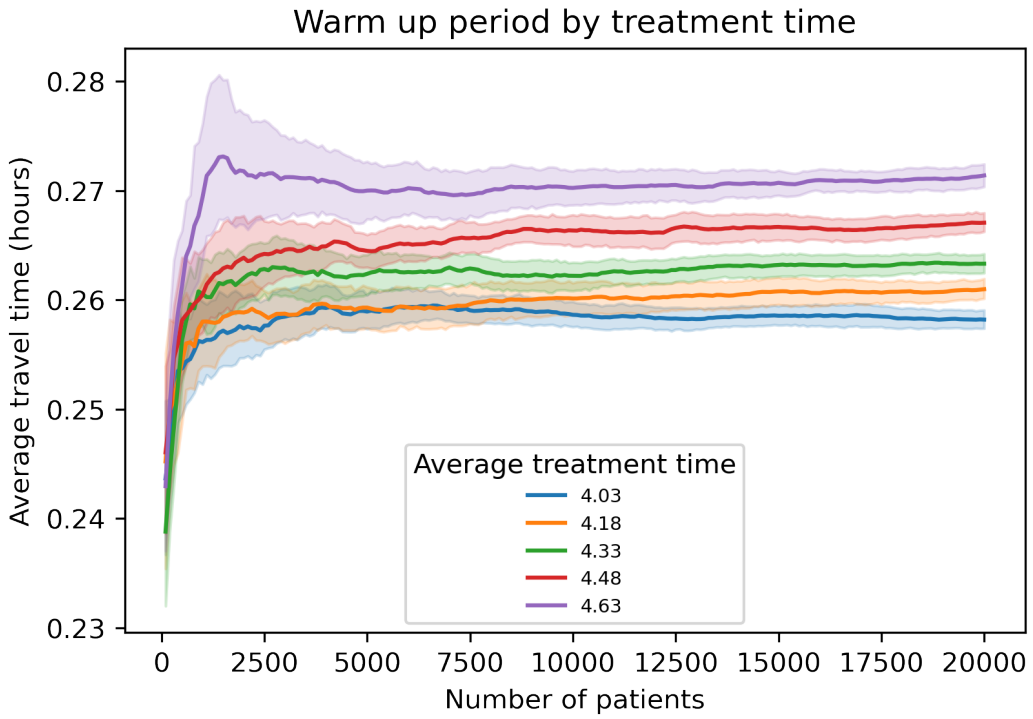

FIGURE 4.7: Warm up period for different treatment times

patients needing to visit two, three or more facilities increase. The probability to being forwarded to another facility increases for all facilities with the average treatment time increasing. A minor exception to this rule is found at facility f_3, but this is most likely the case because the probability of being forwarded is extremely small and therefore more susceptible to changes in the random generated values in the simulation. The simulation length increases with the increase in treatment time, with the scenario with the highest treatment time having a 21 times as high simulation length compared to the scenario with the smallest treatment time.

4.2.2 Vietnam case study

For the Vietnam case study 200 distinct allocations were created for which the average travel time needs to be computed. In this approach it is assumed that people go to the closest stroke center, start treatment in case a bed is available and otherwise travel to the next closest stroke center.

The first step to calculate the average travel time is to determine the warm up period and the run length of the simulation. The warm up period of the simulation is determined by taking 8 out of the 200 allocations and running each simulation 20 times until the resulting travel times of 500,000 patients where gathered. Based on these results the average travel time and 95% confidence interval where calculated for 1 until 500,000

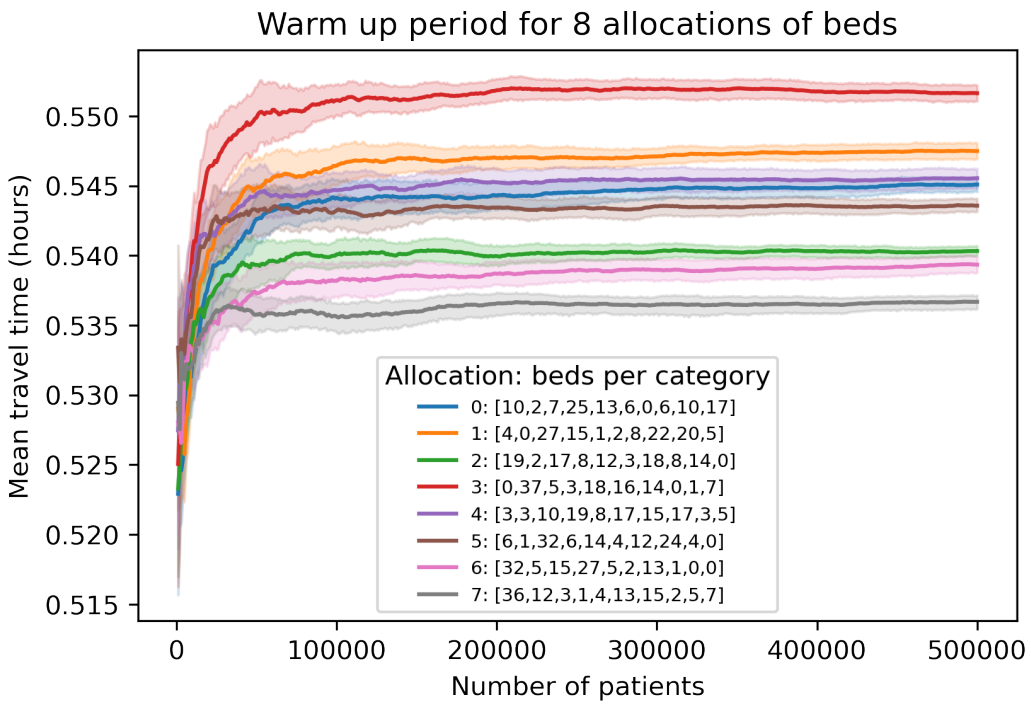

FIGURE 4.8: Warm up period for 8 allocations of beds. The legend includes the beds per category in order from category 1 up to category 10. Mean travel times and confidence intervals are based on 20 repetitions.

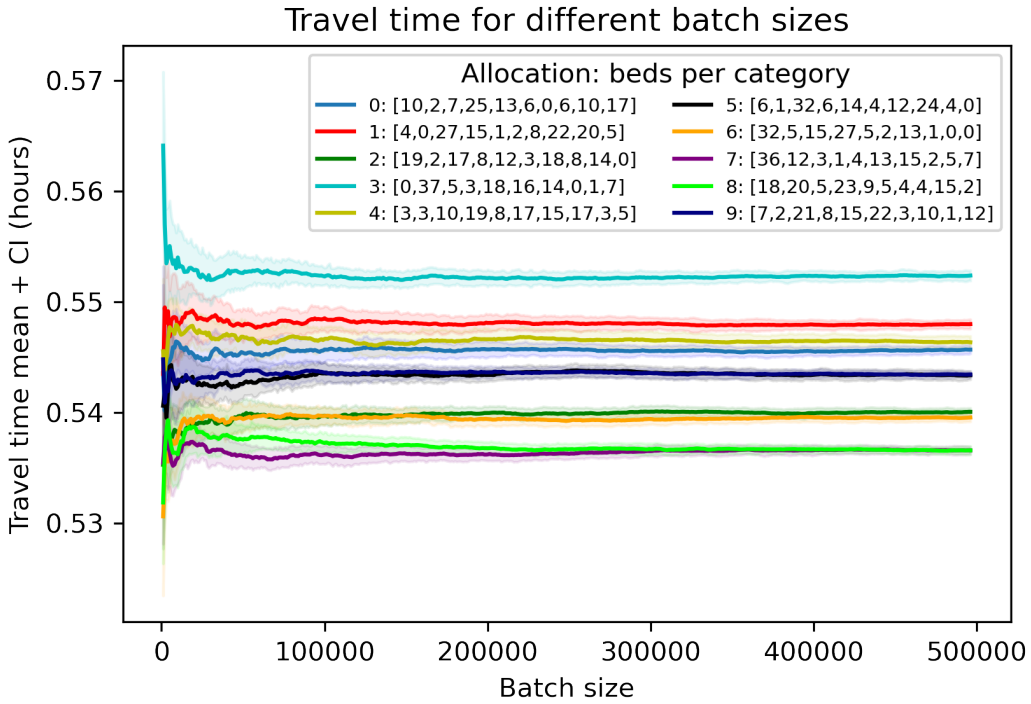

FIGURE 4.9: Average travel time and 95% confidence interval of 10 allocations of beds and different batch sizes. Average travel time and confidence interval are calculated using the batch means method with a 30 number of batches. The warm up period of 100,000 number of people was discarded for the batch means calculation.

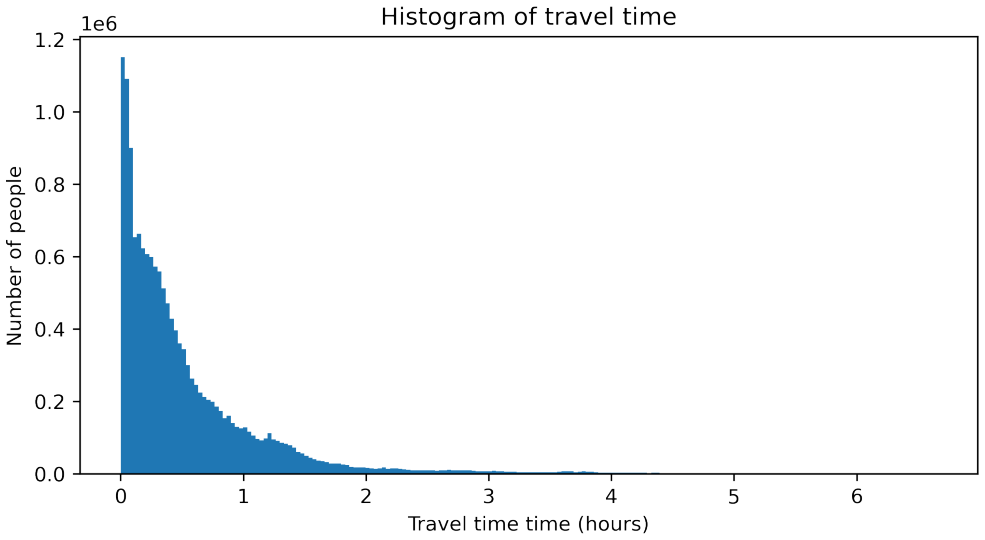

FIGURE 4.10: Histogram of travel time for an allocation from the allocation list.

number of people. The results are given in figure 4.8. Based on this graph we can see that average travel time for small number of people are clearly smaller compared to the average travel time when the number of people increases. This is because the stroke centers are initialized with zero patients, meaning that at the beginning no patient has to travel to the next stroke center. We can see in the graphs that the warm up period has approximately a length of around 100,000 number of patients. We assume this warm up length is representative for the other allocations as well.

After determining the warm up time it is required to determine the run length of the simulation. The average travel time and confidence interval is calculated using the batch means method. With this method one long simulation is used, the warm up period is discarded and the remaining travel times are divided in batches with each the same batch size. For the Vietnam case 30 batches are used, since Schmeiser (1982) [20] advises to take the number of batches not less than 10 and not greater than 30. To determine a suitable batch size a graph was created for 10 of the 200 allocations of beds and the average travel time and 95% confidence interval for different batch sizes are given, see figure 4.9. In this graph can be observed that the difference in the average travel time between these allocations is at maximum 0.016. This means that the desired confidence interval should be small, we would like the size of our confidence interval to be at maximum 1/20 of the maximum difference in travel time. Due to time limitations it is not desirable to have longer simulation run length than 15 million number of people. When using a simulation run length of 15 million people, implying a batch size of 496.666, the size of the confidence intervals of the 8 allocations range from 0.00054 to 0.0011 with an average of 0.00079. The desired confidence interval was 0.0008, hence it is concluded

that on average using a simulation run length of 15 million will result in small enough confidence intervals around the average travel times.

Using a run length of 15 million the average travel times and confidence intervals of the 200 allocations are calculated. It took approximately 258 minutes per allocation to calculate the average travel time using a AMD Ryzen 5 4500U 2.38 GHz CPU and 16 GB RAM. The best of 200 allocations was the allocation [category 1, category 2, ..., category 10] = [17,15,8,3,10,16,14,6,0,13] with average travel time 0.53598 and 95% CI ranging from 0,53563 to 0,53634. The worst allocation was [category 1, category 2, ..., category 10] = [1,5,2,23,13,14,0,18,19,6] with an average travel time of 0.55727 and 95% confidence interval from 0.55680. This is in line with our expectation, because we would expect allocations with higher number of beds at the first few categories would perform better. The best allocation with a maximum budget of 1,000 beds in the allocation dataset is [category 1, category 2, ..., category 10] = [21,10,6,4,12,7,9,9,6,3] with an average travel time of 0.53683 and 95% CI ranging from 0.53645 to 0.53722. On average is the confidence interval half with of the allocations 0.00042. The standard deviation of the allocations is on average 0.63. Figure 4.10 gives an overview of the individual travel times in one of the allocations. From this graph can be conclude that the travel times of patients are in general small, with only a small proportion of patients having higher travel times than 2 hours. The distribution of travel times in fig. 4.10 are representative for the 200 allocations.

4.3 OptiCL

In this section the results are given of the OptiCL approach to the Vietnam case study. First a prediction model is trained using the OptiCL Python package [21] to predict the average travel time for a specific allocation of beds. Six different algorithms are used to find a good prediction for the average travel time: a neural network (mlp), linear model, gradient boosting machine (gbm), decision tree (cart), random forest (rf) and support vector machine (svm).

TABLE 4.3: Mean Squared Error (MSE) and R-Squared (R^2) performance measures for predictive models on test dataset.

Alg	**MSE**	**R^2**
mlp	1.81E-02	-945.22
linear	1.28E-05	0.33
gbm	4.16E-06	0.78
cart	7.53E-06	0.61
rf	5.07E-06	0.74
svm	1.50E-03	-77.09

The results for training these models are given in table 4.3. Based on these metrics, gbm predicted the average travel time best. Mlp and svm performed very badly compared to the other models, with negative R^2 scores. Next, the predictive models are embedded in the optimization formulation to generate the optimal allocation of beds.

TABLE 4.4: Results for allocating beds using OptiCL. Results include the allocation of beds ordered from category 1 to 10 (Alloc.), total number of allocated beds (Beds), average travel time based on prediction model in hours (Tt OptiCL) and average travel time and confidence interval in hours based on the simulation (Tt + 95% CI sim.) for all different predictive models (Alg.)

Alg.	**Alloc.**	**Beds**	**Tt OptiCL**	**Tt + 95% CI sim.**
mlp	[4; 5; 26; 19; 5; 7; 4; 17; 3; 5]	991	0.0003	0.5433 (0.5429, 0.5436)
linear	[38; 6; 8; 6; 9; 6; 8; 3; 3; 9]	998	0.5388	0.5375 (0.5372, 0.5378)
gbm	[21; 21; 14; 2; 4; 4; 3; 5; 9; 13]	998	0.5376	0.5369 (0.5366, 0.5373)
cart	[22; 7; 7; 8; 5; 5; 7; 8; 19; 6]	990	0.5366	0.5380 (0.5374, 0.5386)
rf	[23; 9; 4; 8; 14; 11; 1; 16; 4; 4]	990	0.5380	0.5378 (0.5374, 0.5382)
svm	[25; 5; 12; 15; 2; 14; 3; 0; 5; 8]	922	0.4579	0.5400 (0.5395, 0.5404)

The optimal allocation of beds and their corresponding average travel time while using the predictive models are given in table 4.4. Based on this table it can be concluded that the OptiCL model with gbm as the predictor of average travel time resulted in the best allocation, since it has the lowest average travel time according to the simulation. This average travel time was not smaller than the smallest average travel time with a maximum budget of 1000 beds in the dataset used to train the predictive models. A possible reason for this observation could be that a dataset of only 200 allocations was too small generate models that where good enough in predicting average travel time for an allocation of beds. Another explanation could be that the dataset was generated in a space filling way around the optimal solution of the MA algorithm, meaning that potentially their where already good allocations in the dataset. The svm and mlp model resulted in bad allocations with higher travel time. The number of beds allocated using OptiCL with svm was considerably less than the total budget.

Chapter 5

Discussion

In this section the results from the MA algorithm, simulation and OptiCL are discussed.

5.1 Marginal allocation using queueing theory

The MA algorithm provides exact results, runs fast and scales well to bigger size problems, which makes it a suitable choice for capacity allocation problems were each facility can be seen as a separate queue.

When optimizing the allocation of beds using MA with minimizing the average waiting time for different budgets of beds, the decrease in average waiting time becomes smaller with every added bed. Similar, when optimizing the allocation of beds with MA with minimizing probability of waiting, the decrease in probability of waiting becomes smaller with every added bed. It is also observed that the difference in the resulting average waiting time and waiting probability using MA when minimizing the average waiting time or probability of waiting is small. Furthermore, it can be concluded that in a test case of adding 500 beds using the MA algorithm to minimize the average waiting time that there exists a approximately linear relationship for the stroke centers that received beds between the number of total beds and number of people in the catchment area of a stroke center. Besides that, ρ was small for stroke centers with small number of people compared to stroke centers with higher number of people in the catchment area. This indicates that the utilization of stroke centers with lower number of people is relatively low. Finally, it can be concluded that the average waiting time is relative high for stroke centers with smaller number of people compared to stroke centers with larger number of people in the catchment area. This is acceptable, since longer average waiting times have an impact on fewer people for stroke centers with smaller number of people in the

catchment area. In case the population is very unequal distributed across facilities, the differences in average waiting time per facility can become significant, which might be undesirable in practice. The above found results also apply for the MA algorithm that minimizes the probability of waiting per patient.

The main disadvantage of this method is the assumption that people can only go to the most closest stroke center. Especially in case facilities are very close to each other or when a patient has multiple facilities that are equally close this assumption does not hold. Besides that, the current implementation of the algorithm is limited to M/M/s type of queues. Furthermore, the number of patients arriving at the facilities is based on the number of people for whom this facility was closest combined with a probability per person to get sick. In reality the probability of getting sick differs from person to person. To keep the model simple it is decided not to account for socio-demographic factors for the probability of getting sick.

5.2 Simulation

When examining the influence of the parameters in the model on the average travel time, it was found that with increasing population size all the following measures increase: the average travel time, standard deviation, length of the simulation, probability that three or more facilities are visited before receiving treatment and the probability of being forwarded to another facility for all facilities increases. The only measures that show a decrease are the probability of patients being treated at the first or second visited facility. With increasing treatment time similar results are found, the average travel time, standard deviation, length of the simulation, probability that three or more facilities are visited before receiving treatment and the probability of being forwarded to another facility increases, while the probability of patients being treated at the first or second visited facility decreases.

For the simulation results for the Vietnam case, it can be concluded that the differences in average travel time between allocations are extremely small. This is because the number of total beds is relatively high with respect to the number of people going to the stoke centers with a stroke, which was done deliberately, as the simulation runtime increases when the capacity is smaller compared to the number of stroke victims.

Furthermore, the simulation time per simulation is quite long, because the differences in travel time were small, meaning that long simulation run length was required to reach a small enough confidence interval around the average travel time. This long simulation

runtime limited the number of allocations for which the average travel time could be calculated.

The current simulation is limited to the behaviour that patients go to the closest facility, the next closest facility if full, etc. In reality the behaviourall pattern of patients might be different, for example when patients or healthcare providers have access to information on which facility has capacity to treat the patient, the patient can immediately go to the closest facility with capacity available. Besides that, is is assumed that the treatment time distribution follows an exponential distribution, which is a common service distribution, but it is not guaranteed that treatment times in practice follow an exponential distribution. Similar to the MA algorithm, socio-demographic factors are not taken into account in the probability of getting ill. Finally, to calculate the travel time between facilities an assumption was made that people travel an average of 70 km/u. This assumption deviates from the real world, because the speed a person can drive is, among other things, dependent on the types of roads, speed limit, congestions and type of transport.

5.3 OptiCL

When using the OptiCL method to calculate the optimal allocation of beds for the Vietnam case, big differences are observed in performance between the predictive models. The mlp and svm method perform badly compared to the other methods. The mlp method performs most badly. A possible explanation is that the method usually requires large number of samples, and in this study only 200 samples are used. The other predictive models performs significantly better, with the gbm method delivering the best results. None of the models generates an allocation that is better than the best allocation in the dataset with a budget of 1000. The reason for this could be that the dataset with allocations were generated in a space filling way around the optimal solution of the MA algorithm, implying there were already near optimal solutions in the generated dataset. Another reason could be that a dataset of 200 allocations is too small to build a predictive model with enough predictive power. Another potential reason for decrease in predictive power of the models is that the travel times of the different simulations were close to each other, and it might be the case that the chosen size of the confidence interval was too big to give a good separation between the allocations. This implies that the average travel times resulting from the simulation used for OptiCL had some noise, which could have been decreased by running the simulations with a longer simulation length. Although none of the methods give better allocations than the dataset, the gbm predictive model provides us with a reasonable good allocation with an average travel

time close to the best travel time in the dataset with a budget of 1000. This supports the conviction that the method is working well in producing reasonable allocations.

Chapter 6

Conclusion and future work

6.1 Research goal

This thesis answers the research question: what is a computationally attractive model to allocate capacity to healthcare facilities in such a way that the benefit for patients is optimized? The goal is to make a model for generic geospatial healthcare facility allocation problems. The models are applied to stroke centers in Vietnam. For this test case the goal is to give the optimal allocation of beds given a maximum budget of beds to add to the stroke centers in such a way that it is most beneficial to the patients. The research question is answered using two sub-parts. In the first part a model is made for allocating capacity to healthcare facilities when the assumption is made that patients are always treated in the closest facility. Making this assumption allowed to treat each facility as a queueing system and in that case it is possible to define "most beneficial for patients" either as the minimal average queueing time per patient or the minimal probability of waiting per patient. In the second part a model is made to allocate capacity to healthcare facilities in the most beneficial way for patients when it is not possible to make the assumption that patients always go to the closest facility, i.e., when forwarding patients between facilities is possible. For the second part it was assumed that patients go first to the closest stroke center, if enough capacity is available they are treated, and if this is not the case they go to the next closest facility until a facility is found where treatment is possible. In this model is most beneficial for patients defined as the minimal average travel time for patients. For this type of model the average travel time is computed using a simulation and it has been researched whether OptiCL is a suitable method for determining the optimal allocation of capacity.

6.2 Conclusion

6.2.1 MA

In case the assumption of no forwarding between facilities can be made a MA model is used to minimize average waiting time or probability of waiting of patients. The MA algorithm gives exact results, has a short runtime and is easily scalable to larger size problems. Therefore, it can be concluded that the MA algorithm is a suitable algorithm for solving geospatial healthcare facility allocation problems where the assumption can be made that patients are always treated at the closest available facility.

6.2.2 OptiCL and simulation

The optimal allocation of capacity in case patients can be forwarded between facilities can be found using OptiCL. With this method the capacity allocation problem is formulated as a mathematical model with the goal to minimize the average travel time. A function to calculate the average travel time based on the capacity is built by training a machine learning model on a dataset with allocations of capacity and the average travel time found using the simulation. When simulating the allocations for the Vietnam test case, the differences found in average travel time between the allocations where small, which resulted in long simulation runtime in order to achieve the required confidence intervals. When using OptiCL to minimize the average travel time, it was found that there are very big differences in the optimal allocations and their corresponding average travel time between the different machine learning methods. The OptiCL model with mlp and svm models performed badly, with mlp giving the worst result. The best average travel time was achieved by the allocation from the OptiCL model with gbm as a predictive model. Although the best performing OptiCL model came close to the best average travel time in the dataset within the budget, none of the OptiCL models gave a lower average travel time compared to the smallest average travel time within the budget in the dataset used to train the predictive models. Therefore, is can be concluded that OptiCL could be a promising method to allocate capacity, but future research is needed to provide better insight in the predictive power of the prediction models in OptiCL.

6.3 Contribution to existing literature

This research made some extensions to the original MA algorithm from Weber [15]. In the first place this research changed the goal of the algorithm from minimizing the sum

of average travel time per facility to minimizing the average travel time per patient. Secondly, this research concluded that the MA algorithm can not only be used to minimize the average waiting time, but also to minimize the probability of waiting. Besides that, the OptiCL method has never been applied to capacity allocation problems of this form. This thesis provides a first insight in whether this method can be suitable for solving capacity allocation problems.

6.4 Future work

Finally, the opportunities for future work are described. First we explain future work on the MA algorithm, the simulation and the OptiCL method and after that some suggestions are presented on how to combine the facility location model with capacity allocation.

6.4.1 MA

Although the MA algorithm is currently limited to M/M/s queues, it is possible to extend the algorithm to other types of queues. Valuable extensions could include queues with other type of service distributions or extending to a M/M/s/s type of queue. With a M/M/s/s type of queue, it is assumed that patients do not join the waiting line, but are gone in case they cannot be immediately served. A performance indicator could in that case be the blocking probability (Erlang-B formula). Additionally, more research can be dedicated to socio-demographic factors leading to strokes. This could answer the question if in some area's the probability of getting a stroke is higher or lower due to specific socio-demographics in some regions. Taking these factors into account in the algorithm, will make the algorithm more realistic. A big limitation of the algorithm discussed is that we made the assumption that everyone goes to the closest facility. In practice, a patient has the possibility to go the next closest facility in case it is located nearby, hence reducing the impact of a long waiting queue at the closest facility. A possible solution is to see nearby facilities as one group, apply the MA method and divide the allocated capacity again over the facilities within the group using MA.

6.4.2 Simulation

The most important improvement in the simulation is to decrease the simulation runtime. This can be done by improving the implementation, for example by using a framework especially designed for discrete event simulation such as SimPy. Next to

that, the usability of the simulation for general type of capacity allocation problems can be increased by extending the simulation with other types of treatment time distributions. Finally, to make the calculated travel times as close as possible to the real travel times, the calculation of the travel time can be further improved, for example by using a routing machine instead of assuming a constant speed. Next to that, future research can be done to the socio-demographic factors leading to stroke and incorporating this into the simulation. The simulation could also be extended with other patterns of choosing a facility, for example going to the closest facility with available capacity.

6.4.3 OptiCL

It is important to invest in future research regarding the use of the OptiCL method for capacity allocation problems. Since there where big differences between the predictions of the different machine learning methods, it would be good to further research the predictive power of the predictive models. It could also be of interest to use other predictive models, which are especially suitable for smaller datasets, since running a simulation is costly in terms of computing power. An example of such a model is a kriging model, which is currently not implemented in the OptiCL package. It would also be good to do more research on how far the resulting allocation from OptiCL is away from the optimal allocation.

It would also be interesting to make a comparison between the allocation of capacity by the OptiCL method and the MA method, to see what the influence of the assumption of no forwarding between facilities is. One could compare both allocations resulting from OptiCL and MA by running simulations for both optimal allocations and comparing the resulting average travel time of the patients.

6.4.4 Combining capacity allocation with facility location

Future work will be aimed at combining the capacity allocation problem discussed in this thesis with the facility location problem. The first idea on combining the capacity allocation problem with facility location could be by using queueing formulas, comparable to the MA method. In that case the goal function of the facility location problem could be reformulated to a function where we minimize the average queueing time of patients and maximize the number of people that can reach the facility within a maximum travel distance, or alternatively minimizing the average queueing time per patient and giving each patient that cannot reach a facility within the maximum travel distance an extreme high average queueing time. The average queuing time at each facility can be calculated using queueing formulas when assuming M/M/s queuing type, which is

a convex function in the number of beds added to a facility. Furthermore some extra constraints are needed, such as $\rho < 1$ and that the added capacity to the facilities cannot exceed the budget. A similar approach could be used by minimizing the probability of waiting instead of the average travel time. An alternative approach to a capacitated facility location problem could be the following. The goal could be formulated as minimizing the average travel time of patients and that we can minimize their travel time both by building new facilities and by allocating capacity to the facilities. To calculate the average travel time of patients the simulation of this thesis can be used and extended with the possibility to add new facilities. Since the number of decision variables of such a model will be high, OptiCL is not the obvious choice for solving the problem. Instead, methods such as genetic algorithm or local search could be used.

Bibliography

[1] Nguyen Thi Trang Nhung, Tran Khanh Long, Bui Ngoc Linh, Theo Vos, Nguyen Thanh Huong, and Ngo Duc Anh. Estimation of Vietnam national burden of disease 2008. *Asia-Pacific Journal of Public Health*, 26:527–535, 2014.

[2] Thang Nguyen, Seana Gall, Dominique Cadilhac, Hoang Nguyen, Daniel Terry, Binh Pham, Trung Nguyen, An Nguyen, Nha Dao, Chinh Duong, Bau Phan, and Hoang Phan. Processes of stroke unit care and outcomes at discharge in Vietnam: findings from the registry of stroke care quality (RES-Q) in a major public hospital. 2:119–127, 2019.

[3] Nguyen Huu Cong. Stroke care in Vietnam. *International Journal of Stroke*, 2: 279–280, 2007.

[4] Marilyn M. Rymner, Naveed Akhtar, Coleman Martin, and Debbie Summers. Management of acute ischemic stroke: Time is brain. *Missouri Medicine*, 107:333, 2010.

[5] Joyce Antonissen. An optimization tool for facility location in developing countries. Master's thesis, Tilburg University, 2021.

[6] Fleur Theulen. Solving large maximum covering location problems with a GRASP heuristic. Master's thesis, Tilburg University, 2022.

[7] Geospatial planning and budgeting platform, 2022. URL https://github.com/Analytics-for-a-Better-World/GPBP_Analytics_Tools. Accessed: 2022-08-01.

[8] The World Bank Group and the International Monetary Fund (IMF), 2022. URL https://www-worldbank-org.vu-nl.idm.oclc.org/en/about/history/the-world-bank-group-and-the-imf. Accessed: 2022-06-10.

[9] Who we are, 2022. URL https://www-worldbank-org.vu-nl.idm.oclc.org/en/who-we-are. Accessed: 2022-06-10.

[10] Richard Church and Charles ReVelle. The maximal covering location problem. *Papers of the Regional Science Association*, 32:101–118, 1974.

[11] Amir Ahmadi-Javid, Pardis Seyedi, and Siddhartha S. Syam. A survey of healthcare facility location. *Computers Operations Research*, 79:223–263, 2017.

[12] Chawis Boonmee, Mikiharu Arimura, and Takumi Asada. Facility location optimization model for emergency humanitarian logistics. *International Journal of Disaster Risk Reduction*, 24:485–498, 2017.

[13] Donald Gross, John. F. Shortle, James M. Thompson, and Carl M. Harris. *Fundamentals of queueing theory*. A JOHN WILEY SONS, INC., 4 edition, 2013.

[14] H Takagi and B H Walke. Appendix a: Derivation of formulas by queueing theory. *Spectrum Requirement Planning in Wireless Communications*, pages 199–218, 2008.

[15] Richard R Weber. On the marginal benefit of adding servers to G/GI/m queues. *Management Science*, 26:946–951, 1980.

[16] S.C. Brailsford and N.A. Hilton. A comparison of discrete event simulation and system dynamics for modelling health care systems. In J. Riley, editor, *Planning for the Future: Health Service Quality and Emergency Accessibility*. Glasgow Caledonian University, 2001.

[17] Winfried K. Grassmann. When, and when not to use warm-up periods in discrete event simulation. In *Proceedings of the 2nd International Conference on Simulation Tools and Techniques*, Simutools '09. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2009.

[18] Sigrón Andradñttir and Nilay Tanik Argon. Variance estimation using replicated batch means. *Winter Simulation Conference Proceedings*, 1:338–343, 2001.

[19] Donato Maragno, Holly Wiberg, Dimitris Bertsimas, S. Ilker Birbil, Dick den Hertog, and Adejuyigbe Fajemisin. Mixed-integer optimization with constraint learning. 2021.

[20] Bruce Schmeiser. Batch size effects in the analysis of simulation output. *Operations Research*, 30(3):556–568, 1982.

[21] Opticl, 2022. URL https://github.com/hwiberg/OptiCL. Accessed: 2022-08-19.