Implementing Optimization Algorithms in Swedish Healthcare

AI-Based Simulation and Optimization of Internal Patient Transports at Sahlgrenska University Hospital

Bachelor's thesis in the Department of Technology Management and Economics

Ahmed Deer
Sara Furborg
Victoria Thulin
Marius Westerlind

DEPARTMENT OF TECHNOLOGY MANAGEMENT AND ECONOMICS
Divison of Entrepreneurship and Strategy

CHALMERS UNIVERSITY OF TECHNOLOGY Gothenburg, Sweden 2025
www.chalmers.se
Bachelor Thesis TEKX18-VT25-02

Bachelor's thesis TEKX18-VT25-02 2025

Implementing Optimization Algorithms in Swedish Healthcare

AI-Based Simulation and Optimization of Internal Patient Transports at Sahlgrenska University Hospital

Implementation av Optimeringsalgoritmer i Svensk Sjukvård

AI-baserad simulering och optimisering av intern patient transport på Sahlgrenska universitetssjukhus

Ahmed Deer Sara Furborg
Victoria Thulin Marius Westerlind

CHALMERS

Department of Technology Management and Economics
Division of Entrepreneurship and Strategy
Chalmers University of Technology
Gothenburg, Sweden 2025

Implementing Optimization Algorithms in Swedish Healthcare
AI-Based Simulation and Optimization of Internal Patient Transports at Sahlgrenska University Hospital

Ahmed Deer
Sara Furborg
Victoria Thulin
Marius Westerlind

Supervisor: Christer Ljungwall,
Department of Entrepreneurship and Strategy, Technology Management and Economics
Examiner: Martin Löwstedt,
Department of Innovation and R and D Management, Technology Management and Economics Bachelor's Thesis 2025
Department of Entrepreneurship and Strategy
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Implementing Optimization Algorithms in Swedish Healthcare
AI-Based Simulation and Optimization of Internal Patient Transports at Sahlgrenska University Hospital

Ahmed Deer Sara Furborg
Victoria Thulin Marius Westerlind

Department of Technology Management and Economics
Division of Entrepreneurship and Strategy
Chalmers University of Technology

Abstract

Artificial Intelligence (AI) is increasingly being integrated into the field of medicine to improve treatment outcome, enhance efficiency and streamline internal hospital transports. This study investigates the application of AI to optimize patient transport at Sahlgrenska University Hospital. The goal is to identify different methods that can help to improve patient flow and resource allocation. Interviews at Sahlgrenska along with an expert within AI were conducted to gain a better understanding of how AI can be altered and used. Anonymous important data from Sahlgrenska with patient transport times, different departments and porter availability were analyzed over the course of a year's time. The main challenge was creating an AI-based simulation system that accurately represents Sahlgrenska and also showcases a model of how the porters transport patients throughout the hospital. This was done with genetic algorithms (GA) as well as integer linear programming (ILP) and for the systems to give valid data the hospital transport data must closely replicate the actual size and operations of the hospital.

The results show that the AI-based system could substantially improve hospital transports in regards to distributing the workload more equitably, maximizing efficiency and reducing cost as well as preventing further financial losses. Finally a conclusion that an AI-based system for coordinating patient transport could be drawn. Challenges for implementations include system integration, data privacy and ethical task allocation. Nevertheless, strategic planning along with investments, AI offers considerable potential when it comes to modernizing hospital logistics and fostering a more efficient and patient-oriented healthcare system.

Keywords: Artificial Intelligence (AI), Patient Transport Optimization, Healthcare Logistics, Simulation Modeling, Genetic Algorithms (GA), Integer Linear Programming (ILP), Workload Balancing, Resource Allocation, Efficiency Improvement, Sahlgrenska University Hospital

Sammanfattning

Artificiell intelligens integreras i allt större utsträckning inom medicinområdet för att förbättra behandlingsresultat, öka effektiviteten samt förbättra transporter mellan avdelningar. Den här studien undersöker om AI kan tillämpas för att optimera interna patienttransporter på Sahlgrenska Universitets Sjukhus. Målet är att identifiera olika metoder som kan användas för att förbättra patientflödet och resursfördelningen. Intervjuer på Sahlgrenska tillsammans med en expert inom AI har utförts för att få en djupare förståelse om hur AI kan anpassas och användas. Anonymiserad data från Sahlgrenska med transporttider, olika avdelningar och transportörtillgänglighet analyserades över ett års tid. Den stora utmaningen var att skapa en AI-baserad simulering som represetnerar de olika avdelningarna på Sahlgrenska medan den också visar hur transportörerna förflyttar sig genom sjukhuset. Detta gjordes med hjälp av optimeringsalgoritmer. För att systemen ska ge givande data måste simuleringen vara en representativ bild av Sahlgrenska, borde i storlek och logik.

Resultaten visar att ett AI-baserat system skulle kunna förbättra sjukhustransporter väsentligt när det gäller att fördela arbetsbelastningen mer lika. Det görs för att maximera effektivitet och minska utgifter samt förhindra framtida ekonomiska förluster. Till sist kan en slutsats dras att ett AI-baserat system för att koordinera patienttransporter skulle vara användbart. Utmaningar för implementering inkluderar systemintegration, dataanonymisering och etisk arbetsfördelning. Likväl, strategisk planering och investeringar, erbjuder AI stor potential när det kommer till modernisering av sjukhuslogistik och främja ett mer effektiv och patient-orienterat hälsovårdssystem.

Nyckelord: Artificiell Intelligens (AI), Optimering av patienttransporter, Sjukvårdslogistik, Simuleringsmodellering, Genetiska Algoritmer (GA), Heltalslinjärprogrammering (ILP), Balansering av arbetsbelastning, Resursallokering, Effektivisering, Sahlgrenska Universitetssjukhuset

Acknowledgements

Firstly we would like express our sincere gratitude to our supervisor Christer Ljungwall, Professor at the department of Technology Management and Economics at Chalmers University of Technology, for his excellent guidance during this project. He has been of great help and has always been there if we needed to reach out for advice, big or small. We are extremely grateful and happy for the time we got to share with him this spring.

We would like to thank Marco L. Della Vedova, Senior Lecturer in Vehicle Engineering and Autonomous Systems at Chalmers University of Technology, for helping us and giving his expert consultation. We are extremely grateful to be able to receive such expertise in this project. We would also like to thank Magnus Martinsson, Head of Porters at Sahlgrenska University Hospital, for taking his time to talk with us about where the improvement areas are located. We also want to extend our gratitude to him for the data we were able to receive to have the possibility to create a more accurate simulation model.

Lastly we want to thank everyone who helped contribute in any way, we value all of your opinions and you have helped us tremendously.

List of Acronyms

Below is the list of acronyms that have been used throughout this thesis listed in alphabetical order:

AI	Artificial Intelligence
GA	Genetic Algorithm
ILP	Integer Linear Programming
KPI	Key Performance Indicator
MILP	Mixed-Integer Linear Programming
NURA	Non-Urgent transport Routing Algorithm
OD	Origin-Destination

Nomenclature

Below is the nomenclature of indices, sets, parameters, and variables that have been used throughout this thesis.

Indices

i	Index for transport requests
j	Index for porters (transport agents)
k, l	Indices for locations or nodes in the hospital graph
t	Index for time steps or discrete time points

Sets

$\mathcal{R}$	Set of all transport requests
$\mathcal{P}$	Set of all porters
$\mathcal{L}$	Set of all locations/departments within the hospital
$\mathcal{N}$	Set of nodes in the hospital graph model
$\mathcal{E}$	Set of edges in the hospital graph model (representing pathways)
$\mathcal{T}$	Set of discrete time periods in the scheduling horizon

Parameters

N_R	Total number of transport requests
N_P	Total number of available porters
c_{kl}	Cost or travel time associated with traversing the path/edge between node k and node l
T_{req_i}	Time at which transport request i is made (request timestamp)
O_i	Origin node (location) of transport request i

D_i	Destination node (location) of transport request i
P_i	Priority value assigned to transport request i
T_{pickup_i}	Time required for pickup activities for request i
$T_{dropoff_i}$	Time required for drop-off activities for request i
W_{max_j}	Maximum workload or capacity limit for porter j
Δt	Duration of a single time step in discrete-event simulations

Decision Variables

x_{ij}	Binary variable: 1 if transport request i is assigned to porter j; 0 otherwise
s_i	Start time for the transport of request i
f_i	Finish time for the transport of request i
w_i	Waiting time for transport request i (calculated as $s_i - T_{req_i}$)
y_{jkt}	Binary variable: 1 if porter j is at node k at time t; 0 otherwise

Objective Function Components (Examples)

Z_{wait}	Total waiting time for all requests
$Z_{makespan}$	Makespan: time until the last transport request is completed
$Z_{workload}$	Measure of workload balance among porters

Contents

List of Figures

1

Introduction

In this introduction a brief explanation containing a background providing a clear understanding of the subject will be done. This is to retain a better understanding of the subject and understand all the different perspectives. The problem analysis, purpose and research questions which are presented later on, serves to clarify the specific questions that forms the basis of this study. Delimitations are introduced to limit the study in different capacities and a report structure briefly explaining the study.

1.1 Background

Innovation and groundbreaking ideas have been a hallmark of Swedish healthcare. Since the invention of the pacemaker in 1958, Swedish healthcare has remained at the forefront, with significant innovations such as the Seldinger technique, artificial kidneys and the digitalization of healthcare placing Sweden among the global leaders in the field.

One area that has not kept pace with technological advancements is patient transport. At Sahlgrenska University Hospital in Gothenburg, internal patient transport currently operates through a system where departments and clinics enter transport requests into a system, Columna. These bookings are then listed in chronological order based on the requested arrival time. On weekdays between 07:00 and 16:00, a supervisor at the patient porter service assigns transport tasks to porters. Outside these hours and on weekends, the porters themselves determine when it is time to take on transport tasks. Once a supervisor-assigned task is completed, it is again up to the porters to decide which transport in the list is most urgent and/or convenient, primarily based on their current location within the hospital. After approximately three to five transport runs (depending on distance, time, and workload) the porter returns to the porter's lounge at Sahlgrenska for a short rest before the next transport round. This process results in variations in workload distribution, inefficiencies in route planning and difficulties in prioritizing urgent transports.

Efficient management of patient flows and resource allocation is crucial for ensuring smooth and high-quality care within the hospital environment. One of the major challenges in healthcare is optimizing the use of resources, including staff and equipment, to reduce waiting times, avoid bottlenecks, and improve overall efficiency. Internal patient transport is a critical part of this flow, as inefficient transport solutions can lead to delays in patient care and departments not being able to treat as many patients as they could.

AI has the potential to optimize patient transport by automating and improving decision-making related to resource allocation and logistics. AI can analyze data and real-time information about the patient flows to create more efficient transport plans. By using machine learning, AI can identify patterns and predict future needs, enabling proactive management of bottlenecks and prioritization of tasks based on medical urgency.

An AI-based solution could automate transport assignment based on factors such as staff location, workload, and previously completed tasks. This could lead to a fairer distribution of work and reduce the risk of certain tasks being systematically avoided due to their physical demands. Additionally, AI can generate more optimal routes to minimize transport times and reduce waiting periods, thereby improving the efficiency of patient flow. Several studies have examined AI's potential in optimizing hospital logistics, including patient transport. Internationally, for example, a hospital in Tampa, USA, has implemented an AI-driven logistics solution to manage internal patient transport. As a result there have been reports of transportation times being decreased by 35% on average compared to the previous two months. [1]

Despite AI's advantages, there are several challenges associated with implementing AI in patient transport. Technical challenges include system integration, data quality and error handling. An AI system requires accurate and up-to-date information to make optimal decisions. A common issue is incorrectly booked transport requests, where a porter may need to manually correct a destination. This could lead to the AI thinking that the porter is at another location and therefore giving out tasks that are nowhere near the porters actual location. Currently, AI has limited ability to handle such situations without human intervention. The cost of developing and implementing AI solutions contributes to big economical challenges, such as when investing in AI it requires resources for system development, staff training, and maintenance. However, in the long run, an AI solution could lead to cost savings through increased efficiency and reduced staff costs. There are also several ethical challenges, two being fair work distribution and the working environment of staff. While AI can promote fairer task allocation, there is a risk that it may not consider individual factors such as health conditions or the need for less physically demanding tasks. Ensuring transparency in AI decision-making and allowing human oversight are therefore critical aspects of an AI implementation in patient transport.

The use of AI to optimize patient transport within hospitals can bring significant improvements in patient flow, workload distribution and resource utilization. Research and the previous example from Tampa general hospital [1] show that AI can reduce waiting times by optimizing logistics; but technical, economic, and ethical challenges must also be considered. A well-thought-out implementation of AI could lead to a fairer and more efficient patient flow, ultimately improving healthcare quality and working conditions for hospital staff. Making patient transport more efficient would minimize waiting times for both patients and staff and therefore lead to patients getting assessed faster. This would also reduce cost, which leads to more resources left over for other important things such as personnel shortage. With all this information the goal with this study is to reach a conclusion on if and how an AI implementation could make patient transportation at Sahlgrenska university hospital more efficient.

Since hospitals play such a critical role in society where they provide necessary medical services to ensure the health of the public the importance to keep them efficient is a priority. When this becomes compromised because of economic challenges, staff are cut and costs increase, it becomes difficult for a hospital such as Sahlgrenska to operate with the goal of both financial gain and the best care possible. During recent years there have been an increase in patients whereas the queues to receive medical services have grown increasingly longer and the patient transports' waiting time has only become longer.

1.2 Problem analysis

Dependence on manual coordination and porter self-assignment for internal patient transport at Sahlgrenska University Hospital leads to several interconnected problems concerning efficiency, equity, and patient care:

1.2.1 Operational Inefficiency and Delays

Lack of centralized, real-time overview and optimized dispatching logic frequently results in significant waiting times for patients needing transport. This can occur both before a porter is assigned and while waiting for the assigned porter to arrive. Such delays can interrupt schedules in receiving departments (e.g, radiology), delay diagnosis and treatment, and inefficiently use expensive resources and clinical staff time. The system uses its capacity suboptimally because of the inefficient routing and assignment.

1.2.2 Inequitable Workload Distribution

Workload imbalances may result from porters choosing among available tasks based on perceived convenience or location. Some porters may consistently handle more frequent, shorter, or less physically demanding transports, whereas others may be assigned longer or more complex assignments. This unfairness may lower staff morale and increase burnout or dissatisfaction.

1.2.3 Absence of Dynamic, Standardized Priority

Urgent requests are nominally prioritized. However, each porter making manual decisions as to what transport should be done next makes it impossible to dynamically adapt to changing demands and different medical urgency levels across the system.

In a truly optimized system, all pending requests would be compared against available resources to balance urgent need with system efficiency.

1.2.4 Limited Adaptability and Exception Handling

The manual system is limited in its ability to react quickly to unexpected events/errors. The porter must detect and correct the incorrect bookings and wrong destinations manually. The system lacks mechanisms to reroute or redistribute tasks in response to elevator failures or unexpected staff absences.

All these systemic issues create a transport service that is less effective, less predictable, and potentially less fair than it could be. This directly affects the quality and timeliness of patient care, operational friction, costs, and working conditions of transport staff. The situation calls for exploring a more structured, data-driven, intelligent approach based on AI and optimization algorithms while acknowledging significant technology, cost, and ethics implementation challenges.

1.2.5 Specific Problem sub-goals

Based on the overarching Problem, this thesis investigates AI-driven solutions to address the following deficiencies of the current system:

- A robust prioritization model for dynamically managing transport requests beyond chronology.
- Designing mechanisms to achieve a more equitable load sharing among transport staff.
- Potential technical barriers to AI-based solution implementation and mitigation strategies.

1.3 Purpose

This study aims to investigate Artificial Intelligence (AI) and associated optimization algorithms for improving the internal patient transport system at Sahlgrenska University Hospital. Recognizing this service as a critical logistical component influencing patient flow and resource utilization, the study provides a quantitative evaluation of AI-driven approaches.

More specifically, this research seeks to:

- **Analyze the Present System** Understand the Current patient transport workflow at Sahlgrenska, identifying key processes, bottlenecks, performance indicators, wait times, workload distribution, and stakeholder perspectives through qualitative inquiry and quantitative data analysis.
- **Development of simulation Platform** Based on historical data, design and implement a functional discrete-event Simulation model of the hospital internal transport environment, including layout, resources (porters), and request patterns. This simulator will be the main evaluation instrument for

different management strategies. Test selected AI-driven assignment and optimization Algorithms (Integer Linear Programming and Genetic algorithms) in a simulation environment.
- **Benchmark Performance** Quantitatively compare the Performance of AI strategies with a baseline model reflecting the current operational practices (formed by qualitative analysis) using relevant Key Performance Indicators (KPIs) such as waiting times, transport durations, and workload fairness.
- **Identify Implementation factors** Examine possible ethical hurdles to adopting such an AI system in a real hospital setting.

This study ultimately aims to provide evidence-based insights on whether and how AI can contribute to a more reliable, equitable, and patient-centered internal transport service at Sahlgrenska. These findings highlight benefits and considerations for future decision-making regarding the adoption of AI technologies in this operational domain.

1.4 Research Questions

To guide the investigation towards fulfilling the study's purpose, the central research question is formulated as: *How can Artificial Intelligence (AI) and associated optimization algorithms be utilised to optimize the internal patient transport system at Sahlgrenska University Hospital, considering efficiency, workload distribution, and implementation feasibility?*

To provide a structured approach to answering this overarching question, the following specific sub-questions are addressed:

1. How can an AI-based prioritization model be designed and integrated into an assignment strategy to manage transport requests more effectively than current practices, considering factors such as medical urgency, waiting time targets, resource availability, and estimated transport duration?
2. To what extent can AI-driven assignment algorithms (specifically ILP and GA) facilitate a more balanced and equitable workload distribution among patient transport staff compared to the current system, as evaluated through simulation?
3. What mechanisms can be incorporated into AI optimization models to implicitly or explicitly consider staff rest or workload recovery, and how might these impact overall system efficiency and fairness?
4. What are the primary anticipated technical barriers (e.g., data integration, system compatibility, real-time requirements) and ethical considerations for implementing an AI-driven patient transport solution at Sahlgrenska, and what strategies could mitigate these technical barriers?

1.5 Delimitations

To keep this study focused and to ensure its feasibility within the bounds of a bachelor thesis project, a few delimitations were drawn:

1.5.1 Geographical Delimitation

The empirical investigation and context are strictly limited to Sahlgrenska University Hospital in Gothenburg, Sweden. While methodologies like simulation, AI algorithms, and general results regarding logistics optimization may be of broader relevance, the specific quantitative results, workflow analysis, and recommendations are tailored to this hospital environment. Established contacts and relevant data from Sahlgrenska supported this focus.

1.5.2 Technical Delimitation

The investigated technological solutions are restricted to AI and Operations Research algorithms for optimizing logistics and decision-making aspects of patient transport. The work evaluates Integer Linear Programming (ILP) as an exact optimization method and Genetic Algorithms (GA) as a metaheuristic approach. It excludes the exploration of physical automation (autonomous mobile robots) or novel hardware for transport. The simulated solutions work within the existing hospital infrastructure, possibly using the existing communication devices of porters. Further advanced AI techniques like Reinforcement Learning (RL) were also considered during initial planning and expert consultation, but were not implemented within this project, which compares established optimization paradigms (ILP vs GA).

1.5.3 Methodological Delimitation

This is a mixed-methods study combining qualitative data (interviews, observations) to understand the current system, inform model design, and quantitative data (historical transport logs) to provide baseline analysis and simulation input. A functional discrete-event simulator was developed as the primary evaluation tool for different assignment strategies. The evaluation is fundamentally based on realistically comparing the performance of optimization algorithms (ILP, GA) with a baseline algorithm implemented within the simulator, designed to mimic the current operational logic derived from qualitative analysis. This comparative analysis is done in the simulation environment. The study does not involve implementing, piloting, or testing any proposed AI system within Sahlgrenska operations. All performance evaluations and conclusions regarding AI potential are based on historical data analysis and simulator results.

1.5.4 Simulator System Delimitations (Based on Expert Consultation)

The developed simulation system captures key dynamics of the transport process for benchmarking purposes. However, it contains some necessary simplifications and is limited as a proof-of-concept model. These boundaries are defined according to the project scope and based on expert consultation (Section 3.2.3):

1.5.4.1 Scope of Optimization Algorithms

The primary focus is on comparing ILP, GA, and a qualitatively derived baseline. No simulator is intended to search for the globally optimal strategy across all possible AI paradigms (e.g., deep learning prediction models or RL-based policy optimization are excluded).

1.5.4.2 Graph Model Fidelity

The hospital layout is represented by a graph constructed algorithmically from historical data. This covers connectivity and historically efficient routes at appropriate scales, but edge weights (travel times) are based on historical medians or estimates and were not calibrated against precise, real-time measurements. Such a limitation makes the simulator more suited for assessing relative performance differences between algorithms rather than making exact absolute predictions for deployment without further graph refinement and validation.

1.5.4.3 Static Environment Assumption

The simulation assumes a static hospital environment. Dynamic variables that may impact real-time transport times, such as variable corridor congestion levels, temporary route blockages, elevator availability, and delays, are not explicitly modelled.

1.5.4.4 Simplified Transporter Agent Model

The simplified Transporter Agent Model is a simulation that models transporters with basic states of idle, moving, and working, as well as cumulative workload scores. However, it simplifies complex human factors by omitting detailed fatigue/recovery models, specific break scheduling rules beyond returning to a lounge, or individual porter speed or skill variations.

1.5.5 Delimitation conclusion

These delimitations are important for interpreting the simulation results and defining the gap between this research prototype and a possible deployable operational system.

1.6 Report structure

This thesis has seven chapters. Chapter 2 reviews the theoretical background and literature relevant to hospital logistics, patient flow optimization, applicable AI techniques (ILP, GA, Dijkstra), simulation modeling in healthcare, and ethical considerations. Chapter 3 describes a mixed methods research methodology concerning data collection procedures (qualitative and quantitative), data analysis techniques (start time recalculation method), expert consultation, simulator development process, and ethical and validity issues. Chapter 4 describes the AI-based simulation system developed for this study. Chapter 5 summarizes the research results, including Sahlgrenska's current state analysis and simulation benchmarking's comparative performance results. These results are analysed and discussed in Chapter 6 with interpretations of their significance, potential implications for AI, implementation barriers, and relations to the theoretical framework. In Chapter 7, the thesis is summarized, research questions answered, and future directions for research are suggested.

1.7 Related work

In this section similar works are highlighted and their contribution is discussed. Genetic Algoritms have been used for routing problems before, for instance Abid et al worked on a time-efficient method with a Genetic Algorithm to find optimal routes for mobile stroke units [2]. Their study concluded that their novel Genetic Algorithm compared to exhaustive search, did better for varying population sizes. Highlighting the effectiveness for Genetic Algorithm in optimization problems.

Oliveira et al. demonstrated a heuristic model using Mixed Integer Linear Programming for non-emergency patient transport services[3]. Key findings were that for problems with smaller capacities e.g. with one vehicle, the algorithm provided good results for both accuracy and computation time.

Kühn et al. discussed process optimization in hospitals using Genetic Algorithm [4]. Their findings resulted in 40 percentage reduced waiting times for cancer treatment center. Highlighting once again the effectiveness for GA in hospital settings.

2

Theoretical Framework

2.1 Artificial Intelligence in Logistics and Healthcare

To visualize Sahlgrenska university hospital, a conceptual model of the hospital was designed in python. The conceptual model utilized two different optimization algorithms. On this section the two different optimization algorithms will be discussed in regards to scalability: How scalable the model will be if the hospital size varies. Implementation: How complex the algorithm will be to implement in a hospital environment. Long-term cost efficiency: How manageable the algorithm is in regards to benefit to cost ratio. The specific Genetic Algorithm used on this project is demonstrated on figure 1. A more substantial describtion of GAs will be provided in the next subsection.

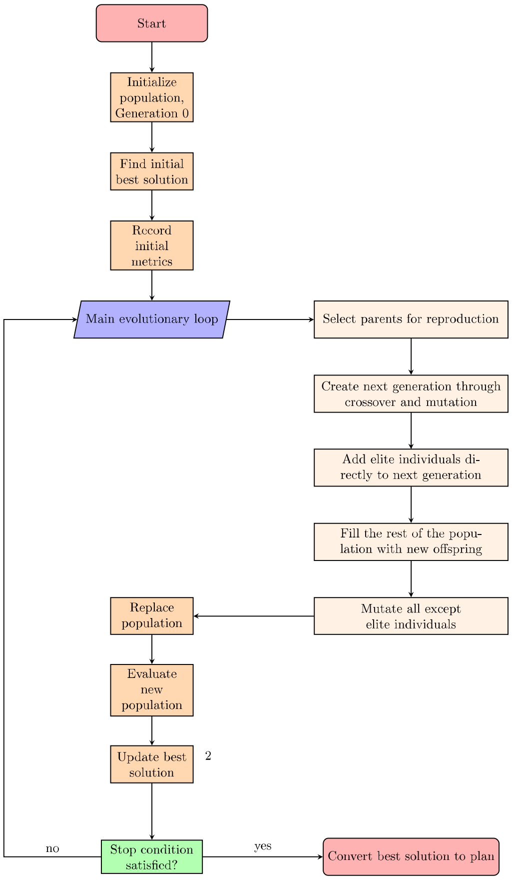


Figure 2.1: Flowchart for GA

2.2 Genetic Algorithms

GA optimization is a model inspired by natural selection. The optimization model roughly works by first initializing a population. The initialization marks the potential solutions to the algorithm. During the initialization process constraints such as population size, number of generations, selection method and crossover method are mapped. The subsequent process is the initial evaluation. During this process the fitness of each individual of the population is calculated and scored whereas the individuals with the highest score is selected for reproduction, which marks the start of the selection process. For the selected individuals, often referred to as parents, the next process for them are the so called cross-over process. This, as the name suggests combines the genetic qualities of each parent to create a new offspring. Essentially introducing a new set of solutions to the algorithm. The algorithm also introduces mutation for the offspring, much as like in nature to avoid homogeneous population. To find an arbitrary set of solution, the processes mentioned prior are executed for a 'n' amount of times. This process is generally mentioned as generations and each generation's purpose is to replace the previous generation, once a break condition is fulfilled and each generation does not improve the algorithm further, the best solution is returned.

2.2.1 Scalability for Genetic Algorithms

GA are scalable depending on constraints or if the problems are non-linear objectives. As an example, an article from 2023 showcased that for a GA used on a mobile stroke unit outperformed the Exhaustive Search 8.75 times faster[2]. On another article, NURA which stands for *Non-Urgent transport Routing Algorithm* which describes a GA with a enhanced scheduling algorithm that assists the algorithm to generate more accurate routes for ambulances outperformed human experts [5]. On 3 different scenarios the human experts induced an average waiting time of 64,3 minutes while for NURA the average waiting time was approximately 58.4 minutes. Albeit being a novice algorithm, it outperformed human experts by an average of 6.0 minutes. The different scenarios mentioned earlier, describes the amount of services and ambulances retained, whereas services depicts patient transports that are for patients that are not in immediate danger. This could be a patient that need a hospital service but is unable to take public transports. For the third scenario, 94 services and 10 ambulances were studied and the total amount of reduced patient waiting time by the NURA was 6.33 minutes. Highlighting that even for complex high dimensional problems were solved better with a GA compared to an expert.

2.2.2 Complexity for Genetic Algorithms

One problem GA might face is that they are rather expensive to compute. According to a literature review from 2021, one of the main struggles with more complex GA are the computational cost [6]. This might indicate that implementing a GA have to run for a considerable amount of time before they yield sufficient results. Which could mean that implementing a GA on a hospital setting could be rather costly in regards to development time. Another issue mentioned in the same article is that for poorly configured GAs struggles with convergence. Convergence is the ability to provide improved solution sets throughout each generation. A GA with low convergence provides identical solutions for each new generation instead of improving them. The occurrence for low convergence could be caused by either having a low genetic diversity in the initial population or having a small population size.

The trade-off for having an algorithm that is rather complex to develop is that the algorithm is reusable over a longer period of time. An article describing tuning for GAs on PLC loops, discusses that by adjusting the fitness evaluation and string encoding, the GA could be reused on a different plant [7]. This could mean that albeit the GA is mainly focused on Sahlgrenska University Hospital, it could be reused on smaller hospitals or health clinics.

2.2.3 Long–term cost efficiency for Genetic Algorithms

An article published 2022 discussing models for stroke rehabilitation care using GAs, showcased by providing a better resource allocation, an estimated amount of 254 500 00 could be saved [8]. Similar finding were done on an article discussing patient transport in intra hospital settings, key findings was that for a GA could reduce patient transport delays up towards 42% [9].

2.3 Integer linear programming

ILP which is a subset of linear programming have had a solid foundation since 1930. During these times both Leonid Kantorovich and Wassily Leontief applied linear programming to both resource allocation and input-output model for economic applications [10].One thing that early developers of linear programming share is that it had the potential to compute actual answers to real-world problems. Since then linear programming have been used in network flow theory, stochastic programming and other commercial applications [11].

ILP works by firstly defining decision variables. The decision variables could be a parable of the problem description, e.g. for a capital budgeting problem i_j could represent a investment variable. While the decision variable has been defined the subsequent step is to define the object function. The object function is the core of linear programming, being that it is the deciding factor for the object outcome. Substantially, the object function decides what problem to maximize or minimize. An example could be: *Minimize the total transport cost or time for patient transport.* Whilst

$$\sum_{i=1}^{m}\sum_{j=1}^{n} c^{ij}x^{ij}$$

is the object function and c^{ij} be the cost or time for the vehicle j and for the patient i. If the object function is well designed and the decision variable is formed the next step is to define the constraints. The constraint describes as the name suggest, a list of conditions for the object function. Which could be the amount of transport each porter could perform or time limits for the operation. The last step is to implement a solver which, together with the constraints, variable bounds and the problem objective solves it and finds the optimal integer solution. The algorithm that the group implemented had "Gurobi" as the solver, which has a lightweight and intuitive programming interface. For the ILP used for the Sahlgrenska simulation, a flowchart is demonstrated for the algorithmic steps below.

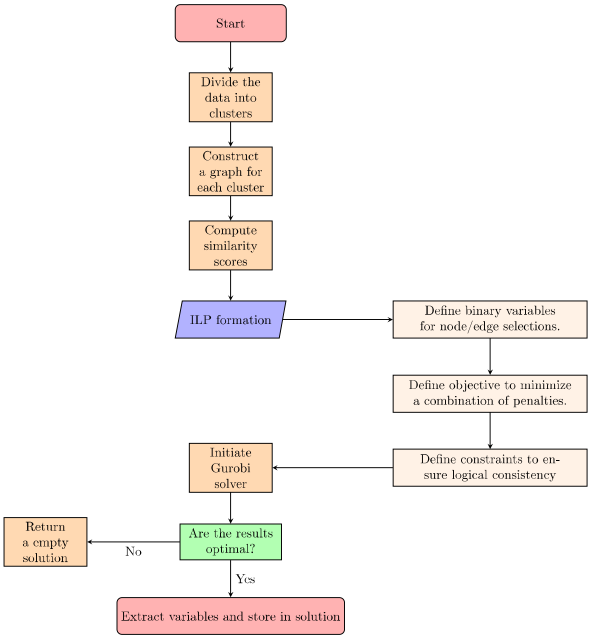


Figure 2.2: Flowchart for ILP

2.3.1 Scalability for ILP

Albeit ILP success when applied to smaller problem sets, it struggles with large datasets. One of the reason for is that ILP are NP hard[12]. NP that stands for *Nondeterministic Polynomial time* are problems that can be checked if they are correct even if they are hard to solve. NP-hard problems describes problems that are not easily checkable and as hard as the hardest NP problems. This in short describes problems that algorithm cannot effectively solve in polynomial time [13]. As the variable size increases so does the solution time, often exponentially [12]. One problem that is hard to avoid when it comes to ILP is that it involves simplification and assumptions that are hard to avoid. Comparing to a GA which can yield better results with less computational complexity [14].

These finding might indicate that hospitals with larger data sets might be difficult to implement ILP algorithm and instead more useful for smaller scale operations.

2.3.2 Complexity for Integer linear programming

Modeling an ILP algorithm for a hospital setting could be very complex. One of the reasons being the amount of constraints needed to to generate a successful algorithm [15]. These constraints could be schedules for breaks, different patient types and legal regulations. Increasing the amount of constraint results in longer computational time and in some cases only 8 hour of computational time could be generated [3]. To implement a successful algorithm expertise in mathematical modeling might be required, being that ILP both the constraint and object function could be mathematical complex. Software required for implementing a successful algorithm could be incompatible with current system used at Sahlgrenska university hospital. Thus, potentially resulting interoperability issues.

2.3.3 Long-term cost efficiency for Integer linear programming

As mentioned earlier with the high computational time needed for ILP, the cost could correspond towards to that time. Licensing fees required for ILP solver could be costly. With an average cost of €9 244,71 according to website "*solver.com*". Combining these factors, ILP might not be as long term cost efficient as GA:s.

2.4 Ethical Considerations for AI in Healthcare

Constructing an algorithm that simplifies patient transport might have some ethical dilemmas that should be accounted for, one of these being that the algorithm shouldn't invoke impairments on the patients health. One way that might happen is if the algorithm prioritizes a less urgent trip, which might occur if the model prioritizes efficiency over safety. If such an occurrence would happen, a procedure for error handling and suggestions on correct course of action is detrimental.

Integrity is another aspect that has to be considered since algorithms that are used in healthcare need patient data to train the model, which is confidential and covered by both GDPR and Patientdatalagen (HIPAA). To develop a sufficient model, large amounts of data is required [16] and since patient data is confidential a solution could be to train the model on synthetic data. Synthetic data differs because it is not traceable to an individual albeit the correlation and statistical relations are still the same [16]. Integrity in regards to porters is essential to account for as well, both union, collective agreements and internal hospital policies when gathering data for the algorithm.

A risk that might be invoked by implementing the algorithm is *Automation bias* [17]. This generally occurs when hospital staff have unrestrained trust on an algorithm, which could place patients in risk if porters aren't vigilant to faulty predictions the algorithm might display. In addition, algorithms might invoke a general loss of competence. Currently medical information resides in literatures, registries and health information systems [18]. If an AI would replace these and serve as a digital storage, a system collapse would invoke safety risk for patients since staff relied on the model for information [18].

Another important aspect to keep in mind when it comes to the conceptual framework of the algorithm is that it needs to not actively discriminate. When it comes to clinical trial data the majority of the data that is gathered is from men with a European heritage [19]. This means that the sampling cohort is disproportionately prevalent, which means that the database is conceptualized after just that. This could exacerbate the algorithm's suggestions for people not within the same sampling cohort who are not as well represented. This is most known under the term of data related bias and is something that is strongly advised to keep away from when constructing the algorithm.

The last key consideration to keep in mind would be distribution of responsibilities. Nursing staff can in case of misjudgement be held accountable and receive repercussions from it, whereas if AI has made the decision the responsible department could have a way harder time deciding on how to move forward. If this decision caused a fatality, who would be held liable? Is it a person with authority that is held responsible or is it the potential developers of the algorithm?

2.5 Supporting Algorithmic Methods and Predictive Analysis

In addition to optimization tools like Genetic Algorithms and Integer Linear Programming, this study also uses complementary methods that support smarter, data-driven decision-making. These supporting algorithms and predictive tools play a crucial role in improving performance and adaptability of the system. Two key areas are explored here: pathfinding algorithms for route optimization and machine learning techniques for prediction and scheduling.

2.5.1 Pathfinding Algorithm (Dijkstra)

Dijkstra's Algorithm can be used to find routes between a starting point and all other reachable points in a network where connections (edges) have a defined positive cost, estimated travel time [20]. Its job is to find the path with the minimum total cost. The method starts at the source location and works outwards step by step, always picking the closest reachable but unseen location. In our simulation, Dijkstra's algorithm takes the hospital layout map as input (a graph with nodes for locations and edges for connections with associated costs). It outputs the best route (sequence of nodes) and total cost (estimated travel time). Dijkstra's was chosen because it is proven to be effective and provides a guarantee of finding optimal paths with defined edge costs. However, it assumes that such costs are static (travel times do not suddenly change because of congestion), as per the parameters of our simulation model.

2.5.2 Machine Learning for Prediction and Scheduling

Besides the specific optimization algorithms like ILP and GA mentioned above, Machine Learning (ML) provides novel ways to improve healthcare logistics through Prediction and dynamic Scheduling in addition to traditional optimization approaches. Although the assignment aspect of patient transport remains the primary focus of this thesis, understanding ML's potential in related areas provides context for future developments and alternative strategies. These applications are briefly reviewed in Oluwafemi et al. [21, p. 6, 8].

A significant application of ML - particularly Supervised Learning - is predictive analytics for demand and duration in healthcare [21, p. 6, 8]. In training models on historical data (such as patient transport logs, hospital admission records), patient demographics, health trends and seasonal patterns ML algorithms can predict future demands for critical resources [21, pp. 8–9]. Examples relevant to patient transport logistics are forecasting patient transport volume by predicting arrival or discharge rates to anticipate demand peaks [21, pp. 8–9]. Models may also predict resource requirements such as the need for porters, wheelchairs, or stretchers based on expected volumes [21, p. 8]. Also, ML techniques such as regression models may estimate task durations such as surgeries or diagnostic procedures, which can improve overall scheduling efficiency when transport is required [21, p. 10, 13].

ML techniques permit more adaptive and dynamic scheduling and resource allocation approaches than static plans. ML algorithms can optimize workforce scheduling to produce staff rosters (including porter rosters) based on predicted demand and availability to reduce understaffing and overstaffing [21, p. 9]. This may include dynamically allocating staff across roles or departments according to real-time requirements [21, p. 9]. Reinforcement Learning (RL) is another paradigm under which an agent learns optimal decision-making policies through environment interaction and reward. RL could develop adaptive scheduling policies for staff/resources that are dynamic according to current conditions and predicted needs, continuously improving the allocation strategy [21, p. 6, 11]. Though not covered in this thesis

(Section 1.5.2), RL implementation is still of great research interest.

Additionally, RL is a theoretical way of overcoming the limitations of static graph models used for simulation (like the one used in this thesis, Section 4.2.3) through iterative edge weight refinement by real-world feedback. Actual hospital transport times vary because of dynamic factors like congestion and elevator delays, which are not captured in static models (Section 1.5.4). An RL agent can adjust the graph's edge weights (W_{ij}) to fit the observed reality of the system better. The agent's "actions" (a_t) are weight adjustments, hospital transport system the "environment" and the agent's "state" (s_t) Current Graph weights and real-time operational data (processed perhaps with graph Neural Networks, GNNs) could also be included [22, Sec. 3.5, 4]. Feedback is provided by completed transports: the actual time ($T_{actual}(i, j)$) is compared with W_{ij} prediction. A reward function ($R(s_t, a_t)$) [22, Sec. 4], would reward accuracy; positive rewards for good predictions and negative rewards for significant deviations (penalties). Repeated interaction teaches the RL agent a policy to adjust weights dynamically - weights are increased for consistently slower routes and decreased for faster ones. This continuous refinement cycle would make the graph model more adaptive, thus making subsequent optimizations based on it more accurate (like ILP or GA). This fits the future research direction of dynamic modeling (Section 7.4).

Furthermore, ML models can support patient flow management by predicting discharge times, better bed management, and anticipation of transport needs [21, p. 10]. These ML applications show capacities that may enhance or replace the ILP and GA strategies evaluated here. They use data for proactive need anticipation and adaptive resource allocation to facilitate better hospital operations [21, p. 8]. However, successful implementation is not without challenges like data quality, integration with existing systems, and ethical issues like bias and transparency [21, pp. 14–17].

2.6 Simulation Modelling and Digital Twins in Healthcare

Simulation modeling effectively analyzes, understands, and improves systems such as Healthcare [23]. It allows decision-makers to assess existing system efficiency [24], to investigate hypothetical "what-if" situations [24]. Simulation is especially useful in healthcare because patient flow, resource allocation, and scheduling are intrinsically complex, variable, and uncertain processes [23]. The simulation model evaluates the impact of possible process performance changes to support evidence-based decision making for improving efficiency, reducing costs, and improving patient satisfaction [24]. Simulation modeling supports managers in decisions regarding process management activities such as strategy design for better outcomes, operation cost reduction, system performance assessment under different conditions, and optimal configuration of resources or workflows [23]. It also helps managers and stakeholders understand systemic issues better because the model represents key aspects and interactions of the real process [23].

2.6.1 Discrete-Event Simulation/DE

Discrete-Event Simulation is a common Simulation technique in healthcare [24]. In particular, DES is suitable for modeling systems whose system state changes at discrete points due to particular events [23]. Such events in this study include patient transport requests, porter assignments, arrival at origin, destination, or task completion. DES models typically represent system components such as entities (e.g., patients, transport requests), resources (e.g., porters, beds, specific equipment), queues (e.g., patients waiting for a porter), and events that drive the process [23]. This method applies to operational and tactical issues such as patient flow management, resource utilization, waiting times, scheduling procedures, and capacity planning [23]. While alternative simulation approaches like Agent-Based Modeling (ABM) focus on emergent behaviors induced by complex interactions of autonomous individual agents, DES is more in line with the aim of this study to evaluate centralized optimization algorithms within a defined workflow defined by specific events and resource flows. One advantage of DES is that it can model complex patient flows and resource contentions in detail [24]. Capturing details about entities, events, and resources allows DES to evaluate performance metrics and understand the impact of process design changes [23].

2.6.2 Simulation as a Decision Support tool

Simulation is an important Decision Support tool in healthcare management [24]. It allows managers and analysts to analyse complex systems, test interventions, compare alternative strategies such as different staffing levels, new scheduling policies, or altered process workflows in a risk-free virtual environment before committing to real-world implementation [24]. Through "what-if" experiments, simulation can predict changes in the future [23], reducing financial risks of failed implementations [24]. This allows for identifying process bottlenecks, better understanding the resource requirement, comparing performance against key indicators (KPIs), and finding configurations that optimize desired outcomes such as reduced waiting times or improved resource utilization [23]. Modern simulation software's visual and interactive nature contributes to decision support by facilitating communication and understanding among various stakeholders and by 'selling' insights and proposed changes [24]. If the simulation results are not immediately implemented, building a simulation model often gives managers new perspectives on system interdependencies and operational dynamics [24].

2.6.3 Digital twins

The advent of Digital Twins in Healthcare models often combines simulation with real-time data connectivity and advanced analytics [25, 26]. A DT is a dynamic virtual representation or replica of some physical entity, process, or system [26]. In healthcare, this physical counterpart may be a medical device, a hospital ward, an entire hospital operation (such as the patient transport system), a biological organ, or even an individual patient [26]. A typical DT has three main components: a physical Twin (the real-world entity or system), a Digital Twin (the virtual model), and a Digital Thread (the data connection providing dynamic, often bidirectional mapping and synchronization between the Physical and virtual counterparts) [26]. DTs use IoT for data sensing from the physical twin, AI or ML for analysis and prediction in the digital twin, and simulation for modeling behavior and testing scenarios [25, 26]. DTs are meant to monitor, analyze, simulate, and predict a physical counterpart's behavior and performance in near real time [26]. Healthcare applications of DTs range from personalized or precision medicine (e.g., generating physiological responses of a patient to different therapies) [26, 25] and surgical planning or training [26]. Simulation models focused on patient transport, for instance, can be considered important components or foundational steps towards full-blown Digital Twins for patient transport systems [26]. This is particularly true if such simulation models are complemented in the future by real-time data feeds from hospital operational systems. Though DTs in healthcare represent a popular and rapidly developing research trend, successful, fully operational implementations are still emerging, although hospital operations and medical devices applications may be more mature due to transferable knowledge from manufacturing sectors [25, 26].

3

Methodology

3.1 Research Approach

This study used a mixed-methods design with sequential explanatory strategies. This approach was chosen because it allows an in-depth investigation of the socio-technical complexity of internal patient transport. The first phase used qualitative methods to understand the operational context, current practices, and problems from stakeholder perspectives, as well as to critically inform the design of subsequent quantitative phases. After this foundational qualitative work was completed, quantitative data analysis as well as simulation modeling were applied to objectively measure the system performance, test hypotheses specific to the system performance derived from the qualitative work, and assess the impact of proposed AI interventions. This mixture of quantitative and qualitative methods leads to insights from the former, which shape and contextualize the latter, leading to a stronger understanding than either method by itself could achieve.

3.2 Data Collection

A variety of Data Collection methods were used to collect information needed to understand the current system at the Sahlgrenska University Hospital and to build and validate the simulation model. This included qualitative data collected through interviews and observations, quantitative information from historical transport logs, and expert consultations.

3.2.1 Qualitative Data (Interviews and Observations)

To strengthen the purpose of this study a semi-structured interview with the head of porters at Sahlgrenska University Hospital was conducted. Magnus Martinsson was interviewed and since he has the role of *head of the porters* he could provide accurate answers to the questions asked. The full interview was carried out in person in Swedish while at Sahlgrenska and was recorded with the intention of later being transcribed (the transcript can be found in Appendix A). This facilitated the possibility to revisit the interview at a later stage for the purpose of summarization. The interview lasted for about an hour and the data along with the answers acquired would be used in the research to identify areas of improvement.

3.2.2 Quantitative Data (Transport Logs)

The primary input for quantitative analysis and simulation was a historical dataset of Sahlgrenska University Hospital's internal patient transport log, which was reported as the *Columna* system. This dataset contains more than 120,000 records of all recorded completed transports during the 2024 calendar year. By ethical requirements and data privacy regulations such as GDPR and Patientdatalagen, the dataset was anonymised before being handed over to the research team - transporter identities were replaced by generic aliases such as "transporter 1", for example. Key data fields used for this study were: unique transport identifier; request creation timestamp; Bestallningstid; porter-logged task start and completion timestamp; Sluttid; origin department code; Från avd; destination department code; assigned priority level; priority level. This anonymized data was granted formal access via Sahlgrenska University Hospital and transferred on physical media (USB stick).

3.2.3 Expert Consultation

It was requested that the methodological rigour and practical relevance of simulation development and the chosen AI approach be assured. Two meetings were held with Marco L. Della Vedova, a Data Science and AI Professor at Chalmers Faculty of Technology, whose research involves simulation and optimization. These consultations covered the presentation of proposed simulation architecture, rationale for Integer Linear Programming (ILP) and genetic Algorithms (GA) as relative optimization techniques, planned data analysis steps, and benchmarking strategy. Discussions included appropriateness of chosen methods for bachelor thesis scope, possible challenges like graph modeling fidelity, acknowledged alternative approaches such as reinforcement learning, which was excluded from the primary scope, and validation considerations. This expert input was a valuable validation check of the study's technical methodology.

3.3 Data Analysis

The data collected will be analyzed differently based on the type and objective of the research.

3.3.1 Quantitative Analysis (Descriptive Statistics and Baseline Assessment)

The quantitative transport log data were analysed mainly with descriptive statistics, which characterised the current system's performance and provided empirically grounded input parameters to the simulation model. This involved: Frequency calculations, distributions, summary statistics, mean, median, percentiles, and standard deviation for variables such as requests per minute, transport durations (at first with raw timestamps), and waiting times (at first with raw timestamps). Analyzing patterns based on origin-destination pairs, priority levels, and temporal factors such as time of day and day of week can help us understand demand characteristics and operation dynamics. Exploring initial workload distribution by transporter ID number or raw duration based on anonymized transports. This initial quantitative analysis helped quantify the operations scale and identify potential areas of high variability or significant delay. It also noted inconsistencies with the start timestamp, reinforcing the need for the data processing steps described in the next section. Basic queueing theory concepts interpreted these descriptive statistics.

3.3.2 Data Processing: Adjusting the starting time and waiting time

A finding derived mainly from qualitative interviews was that porters sometimes log the Starttid timestamp in their system before actual patient transport commencement (e.g., upon task acceptance at the origin department rather than when movement with the patient starts). This discrepancy leads to systematic bias in calculations based on raw timestamps. In particular, it leads to underestimating true patient waiting time from request creation until actual transport start and possibly overestimating active transport duration. A dedicated data processing step was applied to estimate a more realistic "actual" start time, establishing a realistic quantitative baseline to enable robust comparison with simulation results. Method: Estimation of RealStartTime for transport record k using the following procedure:

- Raw duration: Calculate the Duration as recorded by the system. Let $D_{Recorded,k}$ be the recorded duration for transport k, Sluttid_k be the completion timestamp, and Starttid_k be the porter-logged task start timestamp:

  $$D_{Recorded,k} = \text{Sluttid}_k - \text{Starttid}_k$$

- Find Benchmark Durations: Group all transports by origin-destination route r. For each route r, find the subset of transports S_r corresponding to the fastest 10% based on $D_{Recorded,k}$. The median duration in this fastest subset is $\text{BenchmarkDuration}_r$:

  $$\text{BenchmarkDuration}_r = \text{median}\{D_{Recorded,k} \mid k \in S_r\}$$

  This benchmark duration is interpreted as representing an efficient transport time of route r, which corresponds to logged Starttid instances closer to the start of movement.
- Estimate Real Start Time: Estimate the actual start time (RealStartTime_k) by working backward from the recorded completion time, where r is the route for transport k:

  $$\text{RealStartTime}_k = \text{Sluttid}_k - \text{BenchmarkDuration}_r$$

- Calculate Recalculated Waiting Time: Calculate wait time from request creation to estimated actual start. Let $\text{RecalculatedWaitingTime}_k$ be the recalculated waiting time and Beställningstid_k be the request creation timestamp:

  $$\text{RecalculatedWaitingTime}_k = \text{RealStartTime}_k - \text{Beställningstid}_k$$

Cases where this calculation produced a negative value due to a limitation of the benchmark method or other data anomalies were dealt with appropriately (flooring the value at zero or flagging for further investigation).

Outcome: This Recalculated Waiting Time thus provides a more realistic estimate of the requester's delay. This recalculated measure forms the basis of quantitative baseline performance reported in the Results chapter (Section 5.1.2), and can be used more meaningfully to evaluate the performance of simulated scenarios.

3.3.3 Gini Coefficient Analysis for Workload Inequality

To objectively quantify the inequality in workload distribution among transporters, this study employed the Gini coefficient. The Gini coefficient (or Gini index) is the most commonly used measure of inequality [27].

3.3.3.1 Theoretical Foundation

The Gini coefficient was developed by the Italian statistician Corrado Gini (1884–1965) [27]. Although generally employed as a measure of income inequality, the author notes that it can be used to measure any distribution's inequality: wealth distribution, life expectancy, or - as here - workload [27].

The Gini coefficient rates inequality from 0 to 1, where higher values indicate greater inequality. This can be written as a percentage from 0 to 100%, known as the Gini Index [27].

- A value of 0 indicates perfect equality, where everyone (or every entity being measured, e.g., transporter) has the same share (e.g., income or workload) [27].
- A value of 1 indicates perfect inequality, where one person (or entity) receives all of the share, and everyone else receives nothing [27].

A key advantage of the Gini coefficient is its ability to summarize the dispersion of a distribution in a single number.

3.3.3.2 Temporal Structure of Analysis and Methodological Considerations

An important methodological choice was the time window to analyze workload inequality. The hospital transport demand is variable throughout the day, with peak periods occurring during daytime hours and lower demand at nighttime. Comparing workload distribution between high- and low-demand periods directly would be methodologically unsound and could lead to misleading inequality measures. Hence, comparing workload at 2:00 PM (high demand period with many active transporters) versus 2:00 AM (low demand period with fewer transporters) would inappropriately imply high inequality due to differences in active staffing decisions that are often deliberately matched to changing demand.

Therefore, this study examined workload inequality at the same hour across days (e.g., comparing all 2:00 PM slots). In this way, like-for-like periods are compared with expected demand.

However, even with an hourly analysis window, several operational factors may influence the interpretation of workload data and the Gini coefficient:

- Porters have Breaks at specific times and lengths specified in general operational descriptions (Section 1.1). A transporter on a legitimate break during a given hour will have no recorded workload. This might temporarily bias the hourly workload data, making their contribution look lower even when they were active in adjacent hours.
- Transporter shifts start or end in the middle of an analytical hour. Any porter starting partway through an hour or ending their shift halfway through will naturally have a lower recorded hourly workload than someone working the full hour.

A longer analysis window of several hours or an entire shift was considered to reduce the impact of short breaks. Nevertheless, this introduces other major confounding factors. Variable shift start/end times are particularly troublesome in long windows: a porter who worked only the first couple of hours of an extended analysis block would make it appear that she had a very low workload for that block. Furthermore, longer windows would average out the distinct peak and off-peak demand periods, omitting how workload inequality may vary with operational tempo.

With regard to these considerations, an hourly analysis window was maintained to balance the need to compare similar demand periods with short-term biases from breaks and precise shift cutoffs within an hour. Gini coefficients calculated this way should, therefore, be interpreted in the light of these operational realities.

3.3.3.3 Data Preparation and Processing

The anonymized CSV dataset with transport records was first processed to extract relevant workload information. For each hour across dates (e.g., Hour 14 across all days in the dataset), all transports during that hour were identified. A few key processing decisions occurred:

1. Definition of Workload: This data processing included only active transportation time in a transporter's work. Walking distance to the origin of transport was not included. That decision was made because there was no precise data to differentiate walking-to-origin time from other non-transport activities or idle time.
2. Filtering Extreme Durations:All individual active transport records exceeding 30 minutes were excluded. That decision was made after hospital staff (Porter Sara) advised that legitimate single active transports rarely exceed this duration. Some transporters were credited with transports lasting hundreds of minutes in one hour because of logging errors (e.g., not ending a task in the system). This filtering step removed such outliers to avoid disproportionately affecting the workload calculations.

Following those steps, for each transporter active during an hour (and for transports meeting the duration criteria), their working duration in minutes from their completed transports was calculated. All these durations were aggregated for each transporter to get the hourly workload for that hour. The workload was converted to percentages representing each transporter's share of total work (sum of all valid transport durations by all porters) for that hour across all analysed dates.

3.3.3.4 Gini Coefficient Calculation

There are two distinguished conceptual approaches to the Gini coefficient [27]. Suppose one uses the Lorenz curve, where the share of total variable, e.g., workload, is plotted against the population share. The Gini coefficient measures this Lorenz curve deviation from the line of perfect equality (Gini = A/(A + B), where A is the area of deviation) (Hasell, 2023).

A second approach, related more directly to the formula used here, defines the Gini coefficient as the expected difference between any two randomly selected individuals' shares [27]. This is calculated as the expected gap as a share of twice the mean income [or mean share] [27]. This is equivalent to half the relative mean absolute difference.

For a discrete set of workload percentages $x_1, x_2, \ldots, x_n$ for n transporters, where the percentages are sorted in ascending order ($x_i \leq x_{i+1}$), the Gini coefficient (G) was calculated in this study using the formula for the relative mean absolute difference:

$$G = \frac{\sum_{i=1}^{n} \sum_{j=1}^{n} |x_i - x_j|}{2n \sum_{i=1}^{n} x_i}$$

That formula calculates the absolute difference between all possible pairs of workload percentages and normalizes it accordingly. Changes in the middle of the distribution are more sensitive to changes around the middle of the distribution than at the very top and bottom, which may be relevant to variations in workload [27].

3.3.3.5 Visualization and Problem Identification using Gini Coefficient

The methodological approach to visualizing calculated hourly Gini coefficients is intended to produce a quantitative measure of workload inequality. Such quantitative analysis is crucial to support and strengthen qualitative findings from staff interviews objectively (see Section 5.3.3) in which perceptions of uneven workload distribution were reported.

Specifically, the plan is to plot the hourly Gini coefficients on a bar chart. This chart will show the hours of the day (0-23) on the x-axis and the average Gini coefficient calculated for each hour on the y-axis. For context, when reading such a chart, a Gini value of 0.2 is often quoted in literature as indicating moderate inequality. Although hhigher Gini values indicate more inequality without specifying numerical thresholds for terms like moderate, this 0.2 figure can be considered an indicative benchmark [27].

Ultimately, this visualization (with actual graphical results presented in Chapter 5) is not an attempt to propose an operational staffing or assignment solution now, but rather to characterize the extent and temporal patterns of the workload imbalance problem. Providing hospital administrators and stakeholders with an objective, data-driven representation of when workload inequality is most significant, this analysis provides a strong basis to understand one of the key inefficiencies that the proposed AI model (detailed in 4) seeks to correct. Thus, this Gini coefficient analysis quantitatively demonstrates the challenge and points to an AI-driven solution to increase fairness in workload distribution. The interpretation of these Gini values always takes place within the framework of the operation factors and data limitations described previously in Section 3.3.3.2.

3.4 Simulator Development and Implementation

To explore how different AI-based optimization tools could improve patient transport at Sahlgrenska University Hospital, a custom simulation environment was developed. This simulator was designed to replicate the real conditions as closely as possible in the timeframe of this study, while allowing for experimentation with different algorithmic approaches. The following sections describe the purpose behind using simulation, the technical architecture and the development steps that enabled testing and comparison of optimization methods.

3.4.1 Purpose and Rationale for Simulation Approach

A dynamic simulation model was developed to evaluate the influence of AI-driven optimization strategies on the Sahlgrenska internal patient transport system. Simulation is an important controlled testbed since experimenting in a real hospital is not feasible and carries inherent risks when experimenting directly in a live hospital environment. The simulation model enables: (i) modelling intrinsic stochasticity in transport request arrivals and service times; (ii) developing and testing different complex optimization algorithms (ILP and GA) under identical, controlled conditions; (iii) quantitatively comparing the performance of these algorithms with a baseline representing current practices using defined and relevant metrics; (iv) exploring possible impacts of operation parameter changes (e.g. staffing levels, policy adjustments) via "what if" scenario analysis. The development of the simulation model was guided by initial data analysis, qualitative insights, an existing literature review, and expert consultation. While this simulator was designed mainly to benchmark AI-driven assignment strategies, its development and data integration efforts also lay the foundation for a possible Digital Twin of Sahlgrenska's patient transport system if enhanced with real-time data capabilities.

3.4.2 Technology Selection and Justification

The simulation platform was developed mainly in Python, with large libraries for scientific computing, data analysis, optimization, and web development. It was selected based on the project's functional requirements:

- Optimization Engine was Google OR-Tools for implementing the integer linear programming (ILP) models. Its robust solvers and Python API allow the formulation and solution of general combinatorial optimization problems, like task assignment under constraints.
- Data Manipulation & Numerics: The Pandas library was used to read, clean, and analyze the historical CSV log data. Numpy provided fundamental numerical computation, array handling, and probabilistic sampling capabilities for the simulation core and request generation module.
- A real-time visualization dashboard was implemented with Flask as a lightweight web framework for the Backend server. Socket.IO (specifically a Python implementation like python-socketio was added to enable persistent, real-time communication so that the simulation engine can push state updates dynamically to the browser-based dashboard.
- Visualization Front End: The interactive dashboard interface was rendered in the web browser using Standard client-side web technologies: JavaScript (JS), Cascading Style Sheets (CSS), and Hypertext Markup Language (HTML).
- Post-simulation analysis and visualization of benchmark Results were carried out using Matplotlib and Seaborn to produce informative plots (e.g., histograms, and comparative charts) to compare different scenarios and algorithmic performance.

This combined technology stack enabled optimization and simulation logic development, large-scale data handling, real-time interaction, and result presentation while leveraging the development team's experience and expertise.

3.4.3 Development Process and Key Methodological Decisions

The simulator was developed as a modular software design with separation of Key functions (described in Chapter 4) such as graph handling, request generation, routing logic, assignment strategies, agent modeling/visualization, and benchmarking capabilities. This modularity enabled development, systematic testing, and future extensions.

Methodological decisions included data-driven hospital graph construction. Rather than storing the hospital layout as a graph of departments and edges representing pathways, a custom Python implementation was used. The graph topology was extracted algorithmically from historical data analysis: unique Origin-Destination (OD) pairs were identified, and edges representing the fastest median travel times from BenchmarkDuration calculations were iteratively added to reflect frequently used and efficient routes. Edge weights largely relied on these median times or informed estimates. For the purpose and limitations of this graph model, this data-driven approach produced a graph model of suitable scale and complexity for comparative benchmarking of assignment algorithms. However, it is acknowledged as a simplification of the real hospital environment. In particular, precise calibration of edge weights against detailed time-motion studies or the deep integration of qualitative geographical knowledge (e.g. specific corridor layouts, elevator locations) was beyond the scope of this proof-of-concept phase due to inherent complexity and time constraints. Thus, the graph is primarily a testbed for the relative effectiveness of different assignment algorithms rather than a high-fidelity predictive tool suitable for direct deployment without further refinement and validation.

Another area was algorithm implementation. ILP models were written in OR-Tools API with a genetic Algorithm (GA) custom-implemented for chromosome design (solution representation), fitness evaluation logic, and standard Genetic operators (selection, crossover, mutation). A custom Dijkstra algorithm for pathfinding was also written for the simulation graph structure. Choosing to implement and compare both an exact optimization method (ILP) and a metaheuristic approach (GA) was confirmed through expert consultation as an appropriate strategy to explore the optimization landscape within the project constraints.

Lastly, integration and calibration were important. Processed historical data patterns like arrival rates and OD pair frequencies guided the parameter setting of the probabilistic request generator module. The JSON graph model was loaded during initialization of each simulation run. Calibration efforts were focused on ensuring that the generated request load and temporal patterns produced plausible system dynamics during baseline (current practice mimicking) simulations. The Validation strategy for the simulator was based on logical correctness and suitability for comparative benchmarking, considering the delimitations in precise environmental calibration discussed above.

3.4.4 Testing and Validation Approach

In unit testing, key algorithmic components such as Dijkstra's algorithm implementation, GA operators, and critical state management logic in the simulation engine were tested independently with controlled inputs to confirm their functional correctness. Integration testing included complete simulation runs so that all modules interacted correctly. That meant confirming that tasks moved through the system as expected from request creation to assignment, execution, and completion according to the defined logic of the selected assignment strategy (Baseline, ILP, or GA). A baseline plausibility check was also conducted. The simulator ran the baseline algorithm, which was simplified to mimic current operational practices. The resulting output distributions - e.g., shape of waiting time histograms, general workload patterns - were qualitatively compared with the patterns in the recalculated historical data. This was an important check that the simulator produced broadly realistic system behaviour. That implies that comparative results between baseline, ILP, and GA strategies are meaningful within the modeled environment, even though absolute performance values may not precisely match historical averages due to model abstractions. No extensive parameter tuning beyond ensuring this basic plausibility was performed in line with the proof-of-concept nature of the developed simulator.

3.4.5 Outcome

This systematic development led to a functional prototype discrete-event simulator, details of which are given in Chapter 4. Such a simulator provides the platform for Key analytical work in this thesis: controlled comparative experiments to evaluate Performance differences between the baseline operational logic and proposed AI-based assignment strategies (ILP and GA) using relevant Key performance Indicators (KPIs) in a modeled environment representative of Sahlgrenska University Hospital patient transport operations. In addition to batch processing requests for primary benchmarking, attempts were made to build a benchmarking platform where requests could be generated and processed individually. A visual implementation exists where the system can be seen sequentially processing individual requests, but a complete benchmarking platform for this single-request mode was not completed. Finalizing this aspect to produce more granular KPIs under such conditions is identified as an area of potential improvement and future work.

3.5 Reliability and Validity

To ensure the trustworthiness and rigor of this research, several measures were systematically implemented to improve Reliability (consistency of research approach) and validity (accuracy of findings and interpretations).

3.5.1 Triangulation

Triangulation was used across different data sources and research methods to confirm findings and gain a holistic picture of Sahlgrenska's patient transport system. Data Source Triangulation: Information was gathered from historical transport logs (more than 120,000 records for 2024), semi-structured interviews with transport staff and management. For instance, qualitative interview findings indicating uneven workload distribution among porters could be followed by quantitative analysis of anonymized transport log data indicating different task allocation and duration metrics across transporter aliases. Methodological Triangulation included qualitative analysis, quantitative data analysis, and discrete-event simulation. Qualitative insights on current workflows, decision-making heuristics, and perceived inefficiencies guided the design of the simulation model, especially the baseline scenario, which was a reasonable abstraction of the real system. Analyzing historical logs provided a data-driven baseline understanding of system performance and identified problems not initially highlighted in the qualitative phase. The simulator then became an experimental tool to test whether AI-driven interventions could improve upon these identified problems versus the current system.

3.5.2 Clear Documentation

Clear Documentation was attempted throughout the research process. This Chapter details the overall research methodology, including data collection procedures, specific analytical techniques, and the rationale for the simulation design. Chapter 4 describes the specific logic and architecture of an AI-based simulation system. An appendix with a link to the simulator's GitHub repository provides a view of its codebase. The data processing steps, including critical recalculation of start and waiting times (see Section 3.3.2), are described to enable transparency on how the baseline performance metrics were derived and assumptions made.

3.5.3 Methodological Congruence

Careful consideration was given to ensure that the chosen research methods and analytical approaches were in sync with the study's overarching research questions (as defined in Section 1.4) and its purpose. So, for instance, in response to the research question of designing an AI-based prioritization model (RQ1), the simulation system was designed to implement and test different optimization algorithms (ILP and GA) to minimize makespan, balance workload, and handle urgent requests efficiently. Whether these AI-driven algorithms could enable more equitable workload distribution (RQ2) was directly assessed by comparing simulation outputs with relevant workload metrics. The possibility of AI models including staff rest considerations (RQ3) was explored conceptually to identify how rule-based constraints or optimization objectives could be designed within the algorithms managing rest periods, even though a fully optimized dynamic rest scheduling mechanism was not implemented in the current iteration of the simulator. Known technical and ethical barriers (RQ4) were addressed by analyzing the current data infrastructure at Sahlgrenska, considering existing technology (e.g. As an example, the main technical challenge anticipated is to obtain a perfectly accurate hospital model (graph fidelity) which could be iteratively improved in future work by reinforcement learning to update graph edge weights on the fly with real-time feedback. Ethical aspects have been considered from both a worker perspective and an AI-training perspective.

3.5.4 Peer Debriefing and Supervision

Peer and academic supervisor discussions were regular interactions that helped refine the research approach, validate methodological choices, and interpret results. Internal Discussions: Continuous dialogue within the research group, including contributions from members with direct operational experience at Sahlgrenska, e.g. Supervisory Feedback: Meetings with academic supervisors at various points provided critical guidance. Talks by Marco L. Della Vedova focused on technical aspects of simulation, such as the appropriate scale of the model and the decision to limit Reinforcement Learning exploration due to the project scope and complexity. He also pointed out that robust hospital modeling is a precondition for meaningful optimization results and suggested the implementation sequence—develop optimizers only after the core simulation environment is stable. Consultations with Christer Andersson guided the research towards practical relevance and relevant operational challenges for Sahlgrenska.

3.5.5 Expert Consultation (Cross-Referenced)

As in Section 3.2.3, Consultation with Marco L., a Data Science/AI expert, Della Vedova validated the core technical methodology. This involved confirmation that comparing ILP and GA was appropriate for this type of optimization problem, that the simulation should be developed keeping in mind the operation scale of Sahlgrenska, and advising on the possible AI techniques to explore in the limited time window and under the constraints of a bachelor's thesis. This expert input helped shape a feasible, methodologically sound technical approach.

3.5.6 Member Checking (Informal)

Although formal Member Checking - reporting detailed preliminary findings to all interview participants for systematic verification - has not yet been conducted due to project timelines, staff and management were enthusiastic about the project aims and proposed approach. A group member with direct operational experience (Sara Furborg) kept an informal feedback loop regarding the transport system's representation, ensuring practical relevance. Incorporating these measures conscientiously produced credible and reliable findings that lay the basis for further exploration of AI's potential in optimizing patient transport at Sahlgrenska University Hospital.

4

The AI-Based Simulation System

This chapter presents a technical description of a simulation system modelled and evaluated for AI-driven optimization strategies in internal patient transport of Sahlgrenska Universität Hospital. The system's design and implementation was guided by the methodology described in Chapter 3 and expert consultation (Sec 3.2.3).

4.1 System Architecture Overview

The simulation System is a high-fidelity virtual twin of the patient transport process in a hospital environment. While not connected to live hospital data streams in this phase, it replicates real-world operational dynamics using processed historical data and calibrated synthetic request generation. Modeled elements are hospital layout, transport agents, or porters with behavior characteristics like workload accumulation, dynamic path finding, and various task assignment logic. Its main objective is to provide a testbed for comparing the performance of various AI-based assignment strategies, namely Integer Linear Programming (ILP), and Genetic Algorithm (GA) against baseline approaches.

Although designed for offline analysis and benchmarking, the AI-based simulation system described herein resembles several key features of a Digital Twin precursor. It represents a virtual representation of the patient transport process, and its modular design allows for future possible integration with real-time data streams - an important feature of fully realized Digital Twins.

The system is implemented in Python and follows a modular, object-oriented Model-View-Controller (MVC) architectural design for scalability, tests, and maintainability. The Model component is the core of this architecture and contains the core of the simulation logic. A HospitalSystem class manages the simulation lifecycle, including hospital environment setup, graph loading, transporter agent initialization, time, simulation events, updating request arrivals/task completions, and execution of chosen assignment strategies. This class delegated specific tasks to special modules and classes for graph representation and enrichment, Transporter agent behaviour (managed by transporter class), transport request management, implementation of assignment algorithms ILP, GA, Baseline, and pathfinding utilities such as Dijkstra's algorithm, etc.

The View component handles interactions via an API and a visualization layer.

It sends simulation commands as well as user inputs to a controller. It also receives state updates from the model via the controller to show real-time information. The view then displays transporter locations and status, current workload levels, open request queues, and major simulation events. The Controller takes API calls from the view and acts as an intermediary. It then calls appropriate HospitalSystem or other model methods to carry out the requested logic, such as a simulation step or an assignment. The controller then returns results or state updates to the view. This modular MVC architecture separates the concerns so that the core simulation engine, control logic, and user interface/visualization components can be developed independently and tested.

4.2 Data Foundation & Hospital Model

The simulation's realism derives from Sahlgrenska's historical operational Data, which feed The Hospital environment Model and request generation.

4.2.1 Data Source and Initial Preprocessing

The model is based on an anonymized dataset imported into the Sahlgrenska transport logging system with more than 120,000 complete transport records for the year 2024. Key fields utilized from this dataset include origin (Fran avd) and destination (Till avd) department codes, transport type, anonymized transporter IDs, request creation (Bestallningstid), task start (Starttid), and task completion (Sluttid) timestamps, along with the assigned priority level.

Initial data preprocessing, handled by dedicated scripts (e.g., functionality found in transport_data_analyzer.py), involved several steps to ensure data quality and consistency. Cleaning included removing records that did not have origin or destination locations. Transports of negative durations/durations over a plausible maximum, for example, three hours, which may indicate data entry errors, were also filtered out. Furthermore, department names and codes were standardized through normalization techniques, employing methods like prefix matching and Jaccard similarity to reconcile variations in naming conventions found in the raw data, ensuring consistency.

4.2.2 Statistical Analysis for Calibration

Statistical analysis of cleaned historical data was carried out to extract operational patterns relevant to calibrating the simulation model: request generator and graph edge weights. This analysis computed median and 10th percentile transport durations per unique origin-destination pair that are used to derive realistic travel time estimations in the simulation graph. The distribution of transport request arrivals over the day, hourly patterns were chosen to allow realistic temporal request generation. Also, frequencies of different request types and priorities were calculated for probabilistic sampling in the request generator.

4.2.3 Hospital Graph Representation

The physical layout and connectivity of the hospital are represented as a weighted, undirected graph, stored persistently in JSON format. This graph's nodes represent distinct hospital departments or key locations, identified by unique codes and associated with abstract (x, y) spatial coordinates for visualization purposes. Edges represent feasible pathways between these locations. The construction of these edges employed a data-driven, efficiency-filtered algorithm based on historical transport data, as detailed in Section 3.4.3. Origin-destination pairs were processed in order of increasing historical duration, and a direct edge between two locations A and B was added only if no existing path connecting them within a specified tolerance, representing an acceptable detour, was already present in the graph. This method chooses historically efficient connections, implicitly based on probable adjacencies or common routes. The edge weights are estimates of the average travel time, based on median historical transport durations derived from data analysis. These weights are important in the Dijkstra pathfinding algorithm. This graph model provides a spatial and temporal basis for transporter movement simulation and route cost calculation. It simplifies the environment without explicitly modelling multi-floor navigation, specific elevator usage dynamics, or one-way restrictions - these are still areas for future refinement.

4.3 Transport Request Generation module

A separate Module generates dynamic Transport requests over time following the historical patterns to drive the simulation. Implemented with Python libraries like NumPy and Python Pandas, this module works probabilistically based on the statistical analysis discussed in Section 4.1. Request arrival times are generated based on historical hourly distribution patterns to simulate peak and trough times in demand during a typical day. Key attributes are assigned to each generated request by sampling empirically from historical distributions. These attributes are origin and destination nodes selected from observed OD pair frequencies, request type sampled from historical proportions (Wheelchair, stretcher, bed), and priority sampled from historical priority distributions. Thus, each generated request object has attributes such as creation time, origin/destination nodes, priority, and type. Request type may affect assignment eligibility if some transporters are restricted, estimated transport duration calculations used for optimization, and possibly trigger specific handling logic like adding estimated escorting time. Attributes such as specific equipment requirements or detailed patient load classes could be considered, but were simplified to 'Type' in the present implementation.

4.4 Route Planning Module (Dijkstra Implementation)

Whenever Route calculation between two points of the hospital graph is required - for example, to estimate travel time, inform assignment decisions, or direct a transporter - the system uses Dijkstra's algorithm. As discussed in Section 3.4.3, a customized Python implementation for Dijkstra was created instead of utilizing typical libraries. This was done because it needed to be tightly integrated with the simulation's specific graph structure. This custom implementation exposes custom edge metadata like duration, gracefully handles potentially incomplete graph segments during development, and exposes a path cost evaluation interface for assignment modules. Outputs of this module are the shortest path as a sequence of nodes and the total cost as the sum of edge weights corresponding to the estimated travel time. This cost information is an input to the optimization algorithms. Path costs directly form part of the objective function in the ILP module: minimizing total travel time as a function of makespan and/or constraints. In the GA module, path costs are a key factor in the fitness function used to evaluate the quality of potential assignment solutions, known as chromosomes.

4.5 Assignment Strategy Modules

A key feature of the simulation system operates under different transport Assignment strategies for direct performance comparison. The system supports several pluggable assignment modules, including ILP and GA approaches and a baseline strategy. The selection of ILP, an exact method, and GA, a metaheuristic, was validated via expert consultation (Section 3.2.3) to explore different points on the solution quality versus computational effort spectrum.

4.5.1 Integer Linear Programming (ILP) Module

An exact optimization assignment strategy is implemented in this module. When invoked by the simulation controller, typically at discrete decision points such as regular intervals or when significant events occur, it receives the current system state. This state includes a list of pending transport requests, transporter availability/locations, accumulated workload data, and relevant path costs from the Dijkstra module, Section 4.4. Using the Google OR-Tools library , an ILP model is formulated based on this state data. This model includes operational constraints and a configurable goal, such as makespan minimisation or workload balancing. This optimal assignment plan maps requests to transporters and is then returned to the central HospitalSystem controller. The controller, in turn, updates the simulation state accordingly and initiates the tasks assigned to the transporter agents.

4.5.2 Genetic Algorithm (GA) Module

A heuristic assignment scheme based on a custom-implemented Genetic Algorithm alternative to the exact IPL method is presented in this module. It is triggered the same way as the ILP module, and gets the same system state snapshot - pending requests, transporter status, and path costs. The GA module then begins its evolutionary search. During this process, it evaluates potential assignment solutions (chromosomes) based on a fitness function that considers factors like estimated completion times, workload distribution, and request priorities, using cost data from the

Dijkstra module. Upon completion, the best-found assignment plan returns to the HospitalSystem controller for implementation within the simulation based on this heuristically determined solution.

4.5.3 Baseline Algorithm Module

A simpler Baseline assignment Algorithm was applied as a reference to evaluate the AI strategies. This strategy is computationally inexpensive and represents a fundamental, non-optimized strategy. It should be noted that this implemented baseline has random elements and is not comparable to the complex, nuanced human decision-making observed qualitatively; rather, it is a simple performance benchmark. The assignment logic treats pending requests chronologically according to their creation time - Bestallningstid. For transporter selection, a randomly selected transporter from the idle transporters is selected for the subsequent chronological request without considering proximity or estimated travel time. Transporter availability is based on predefined shift schedules. Request priority levels are in the data but not explicitly used in this baseline strategy to change the chronological assignment order or random transporter selection; priority fulfillment is a result rather than an input into the decision logic at this baseline.

4.6 Transporter and Workload Model

The simulation includes a model of the transporters (porters) as active agents tracking their state, movement, and accumulated workload. Every transporter agent is in one of a few defined states managed by the simulation engine at any given point in simulation time. These are Idle, movingToOrigin (pickup of patient/item), waitingAtOrigin (potentially brief wait), movingToDestination (active transport), workingAtDestination (handover time) and ReturningToLounge (associated with the rest behavior). Events such as task assignment, arrival at locations, and completion times trigger state transitions that are updated in the visualization module. A quantitative workload measure is tracked for each transporter. In current implementation, this workload score is the sum of Dijkstra path costs for MovingToOrigin and MovingToDestination movements for all assigned tasks. This is a proxy for physical effort and activity level. The workload score is used in the AI assignment strategies for workload balancing, in the ILP objective or constraints and in the GA fitness function.

4.7 Visualization and Benchmarking System

The system comprises real-time monitoring components and systematic offline performance evaluation. Web-based dashboard application for live visualization of simulation state. Created with Flask and Socket.IO backend and standard JavaScript / HTML / CSS frontend technologies, it lets observers watch transporter status (such as Idle or Moving) and position on an abstract representation of the hospital layout, transport request queue, individual transporter workload scores and key simulation events like request creation, assignment completions.

The system can also be run in automatic benchmarking mode to compare different scenarios systematically. This mode runs several simulation runs with different key parameters. They might include the assignment strategy used (Baseline, ILP, or GA), the number of available transporters, and the request volume (e.g., scaling historical arrival rates). In these benchmarking runs, Performance data and Key performance Indicators are logged. Such include overall makespan, total workload for each transporter and variance of final workload scores among transporters. Other metrics include a normal distribution of the baseline algorithm working, where min, max, median, and mean times are displayed. Aggregated results and comparative plots are generated with libraries like Matplotlib and Seaborn to allow quantitative analysis of the performance trade-offs between the different strategies. Such outputs provide the foundation for the results presented in Chapter 5.respectively and throughput% of completed transports. Aggregated result / comparative plots are generated with libraries like Matplotlib / Seaborn to allow quantitative analysis of the performance trade-offs between the different strategies. Such outputs provide the foundation for results presented in Chapter 5.

5

Results

5.1 Current State Analysis at Sahlgrenska

This section describes the current patient transport system at Sahlgrenska University Hospital. It aims to establish a baseline understanding of the operational landscape, including workflow characteristics, quantitatively defined performance baselines, and staff views on the system's functionalities and challenges. Foundational analysis is critical for contextualising the subsequent evaluation of AI-driven optimization strategies and identifying potential improvements these strategies might bring. Findings presented here are based on a mixed methods approach that combines qualitative and quantitative information from historical transport logs described in Chapter 3.

5.1.1 Workflow Mapping and Identified Inefficiencies

How patient transport works today at Sahlgrenska starts with the porters sitting in their break room inside the hospital. Departments enter jobs into a system called Columna, where they specify the location of the patient, the destination, arrival time, means of transport, and sometimes additional information such as the patient's name or comments about special needs. This information then appears in a list on an app on the phones that each porter carries during their shift. On weekdays between approximately 07:00 and 16:00, a supervisor sits at a computer with this list open and coordinates when it's time for a porter to go out and carry out the transport. The supervisor assigns the job to the porter who has been in the break room the longest, tracked by a list where each porter signs in upon returning to the break room.

After completing the assigned transport, the porter selects their next job themselves by Browse through the list of available tasks. Often choosing a task nearby but sometimes there are urgent tasks to prioritize or the porter may choose less demanding tasks. After usually three to five transports, or about 40 minutes to 1.5 hours, the porter returns to the break room, signs in on the list, and takes a break—usually lasting between 10 to 25 minutes depending on the workload. In the evenings and on weekends, when there is no supervisor present, it is up to the porters themselves to decide when it's time to go out and work. The porters also have to decide which tasks to prioritize and when it is time to execute urgent tasks. In the app the tasks present in a list where if clicked on more information can be seen and a button for "booking" the task, meaning that a porter is claiming the task and it will disappear from the other porters lists. A task can be booked long beforehand and multiple tasks can be booked at the same time. After a task is booked there is a button that says "start" which porters press when they are on their way to the patient's current location. Porters can execute multiple tasks at the same time if for example there are two walking patients that are going from and to the same location.

5.1.2 Quantitative Baseline (Based on Recalculated Data)

Historical transport log data from Sahlgrenska University Hospital for the year 2024 were analysed to establish a robust quantitative baseline for the current patient transport system performance. Critical to that analysis was the recalculation of waiting times due to discrepancies in how porters logged task start times. The porter-logged "Starttid" sometimes reflected task acceptance rather than patient movement start, as described in the methodology (Section 3.3.2). Such a discrepancy may result in falsely underestimating patient waiting times and overestimating active transport durations. Hence, a method to estimate a more realistic start time was applied, resulting in "Recalculated Waiting Times" the basis of the metrics presented here. Figure 5.1 illustrates the characteristics of these recalculated waiting times for patient transports.

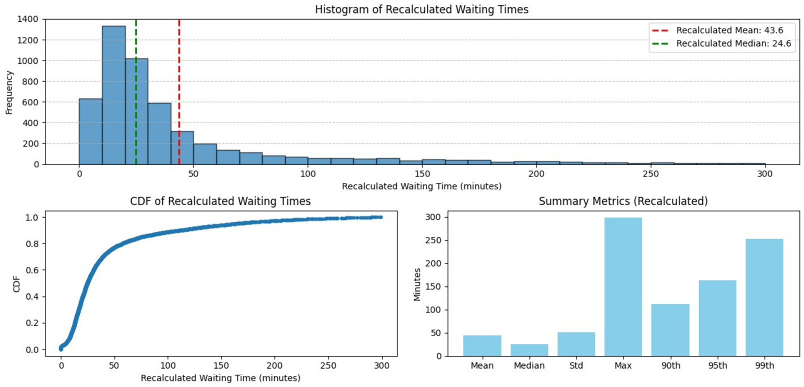


Figure 5.1: Queue Analysis

The top panel of Figure 5.1 (Histogram of Recalculated Waiting Times) displays the frequency distribution of waiting times experienced by patients. It shows a right-skewed distribution, indicating that while many transports have relatively short waiting times, a significant number experience considerably longer delays. The bottom left panel (CDF of Recalculated Waiting Times) illustrates this trend further by showing the proportion of transports that spend their Waiting time less than or equal to a given value. For example, approximately 50% of transports wait X minutes or less and 90% wait Y minutes or less. The bottom right panel (Summary Metrics (Recalculated)) provides key descriptive statistics as a quantitative summary of the waiting time data:

- Mean Waiting Time: The average recalculated waiting time is 43.6 minutes.

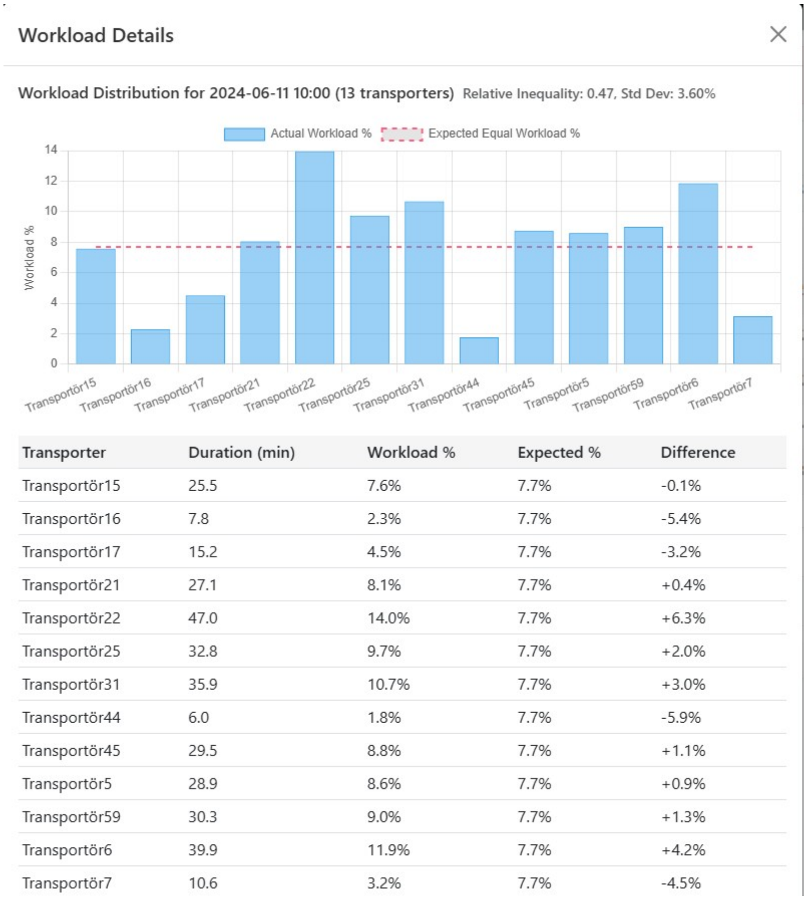
- Median Waiting Time: The median waiting time is 24.6 minutes. The difference between the mean and median further underscores the skewness of the distribution, with the mean being inflated by longer delays.
- Standard Deviation: The standard deviation is approximately 50 minutes, indicating substantial variability in waiting times.
- Maximum Waiting Time: The maximum observed waiting time in this dataset reached 300 minutes as waiting times longer than that were removed from the sample.
- Percentiles:
    - 90th Percentile: 10% of transports experienced a waiting time of around 110 minutes or more.
    - 95th Percentile: 5% of transports waited for approximately 160 minutes or more.
    - 99th Percentile: 1% of transports faced very long delays, waiting around 250 minutes or more.

These recalculated figures represent a baseline for patient waiting times within the present system. The long tail of the distribution indicates that while typical waiting times are moderate, many transports suffer from long delays, which may represent an area for improvement with optimized assignment strategies.

5.1.3 Analysis of Transporter Workload Imbalance

An important aspect of the current state analysis is the distribution of workload among the patient transport staff. To quantitatively assess this, historical transport data was analyzed to understand patterns of workload allocation and identify potential imbalances.

Figure 5.2 presents a snapshot of the workload distribution among a group of 13 transporters at 10:00 on a specific day (2024-06-11). This illustrates the potential for uneven distribution at a given point in time.



Workload Details

Workload Distribution for 2024-06-11 10:00 (13 transporters) Relative Inequality: 0.47, Std Dev: 3.60%

Transporter	Duration (min)	Workload %	Expected %	Difference
Transportör15	25.5	7.6%	7.7%	-0.1%
Transportör16	7.8	2.3%	7.7%	-5.4%
Transportör17	15.2	4.5%	7.7%	-3.2%
Transportör21	27.1	8.1%	7.7%	+0.4%
Transportör22	47.0	14.0%	7.7%	+6.3%
Transportör25	32.8	9.7%	7.7%	+2.0%
Transportör31	35.9	10.7%	7.7%	+3.0%
Transportör44	6.0	1.8%	7.7%	-5.9%
Transportör45	29.5	8.8%	7.7%	+1.1%
Transportör5	28.9	8.6%	7.7%	+0.9%
Transportör59	30.3	9.0%	7.7%	+1.3%
Transportör6	39.9	11.9%	7.7%	+4.2%
Transportör7	10.6	3.2%	7.7%	-4.5%

Figure 5.2: Snapshot of workload distribution at 10:00 on 2024-06-11

The bar chart in Figure 2 shows each transporter's actual workload percentage compared to an expected equal share (dashed red line at 7.7%). For instance, Transporter22 handled 14.0% of the workload (+6.3% above average), while Transporter16 handled only 2.3% (-5.4% below average). The accompanying table details work duration in minutes and percentage workload, with a noted "Relative Inequality: 0.47" and "Std Dev: 3.60%" for this instance, quantifying the imbalance.

To assess if such imbalances are persistent throughout the day, the Gini coefficient of transporter workload was calculated on an hourly basis using historical data. The Gini coefficient, where 0 indicates perfect equality and 1 indicates maximum inequality, provides a standardized measure of workload dispersion. Figure 5.3 depicts this.

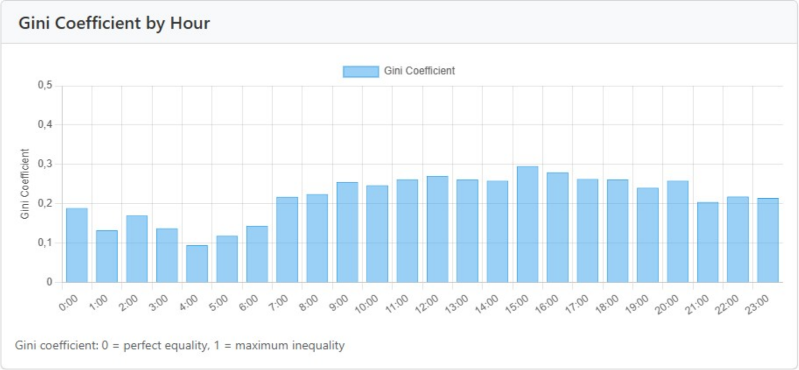


Figure 5.3: Gini Coefficient by Hour

This graph (figure 5.3) depicts the Gini coefficient for porter workload across different hours. Values range from approximately 0.1 in very early morning hours (indicating greater equality when activity is low) to around 0.3 during peak operational periods (e.g., 10:00 to 15:00). A Gini coefficient of 0.3 suggests a discernible level of inequality in workload distribution during busier times.

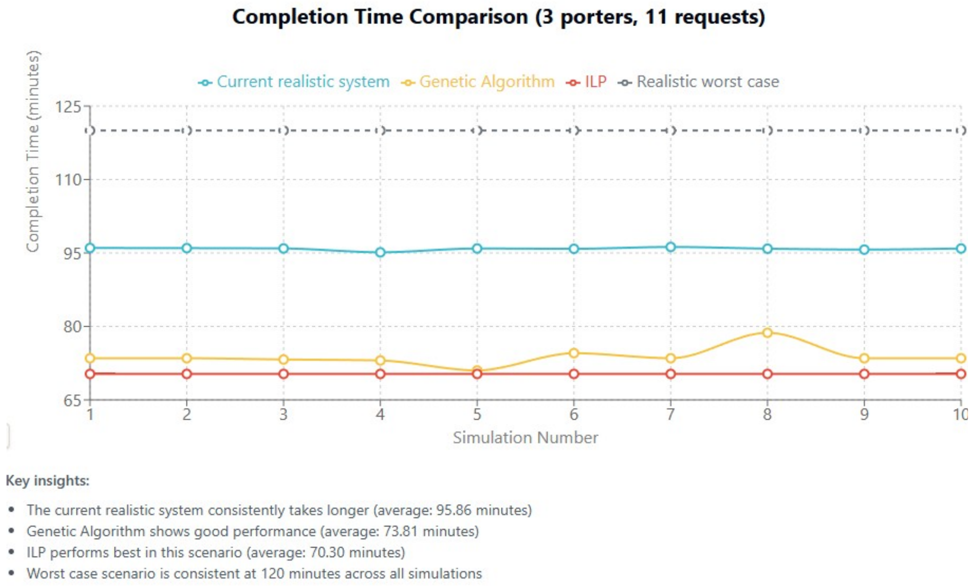
Although individual break patterns may influence hourly metrics, consistent Gini coefficient values around 0.2 to 0.3 during main operational hours (Figure 5.3) indicate that workload imbalance is underlying the current system and not just a side effect of short breaks alone. This corresponds to the qualitative assumption that the present assignment mechanisms do not necessarily promote equitable task distribution (Section 1.2.2). Such findings establish a baseline understanding of workload dynamics and highlight an area for future potential improvement through optimized assignment strategies.

5.2 Simulation and Benchmarking Results

This section presents the results obtained by using the simulator developed in Section 3.4 and detailed in Chapter 4

5.2.1 Performance Comparison of Assignment Strategies

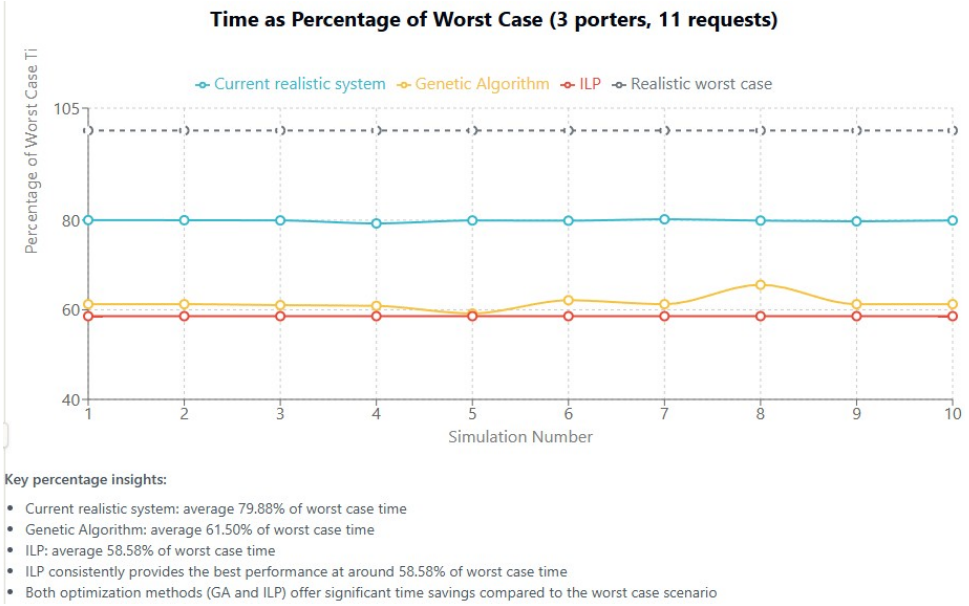
A simulation was made using three porters and generating 11 request. The graph depicts completion time on the y-axis, where the unit of measurement is minutes. The X-axis demonstrates the simulation number. In the figure it can be determined that the optimization algorithms substantially outperformed the realistic system median and worst case.



Key insights:

- The current realistic system consistently takes longer (average: 95.86 minutes)
- Genetic Algorithm shows good performance (average: 73.81 minutes)
- ILP performs best in this scenario (average: 70.30 minutes)
- Worst case scenario is consistent at 120 minutes across all simulations

Figure 5.4: Completion time for each simulation with three porters and 11 requests, the realistic algorithm line is the median of 1000 runs, the worst case is the worst of 1000 runs



Key percentage insights:

- Current realistic system: average 79.88% of worst case time
- Genetic Algorithm: average 61.50% of worst case time
- ILP: average 58.58% of worst case time
- ILP consistently provides the best performance at around 58.58% of worst case time
- Both optimization methods (GA and ILP) offer significant time savings compared to the worst case scenario

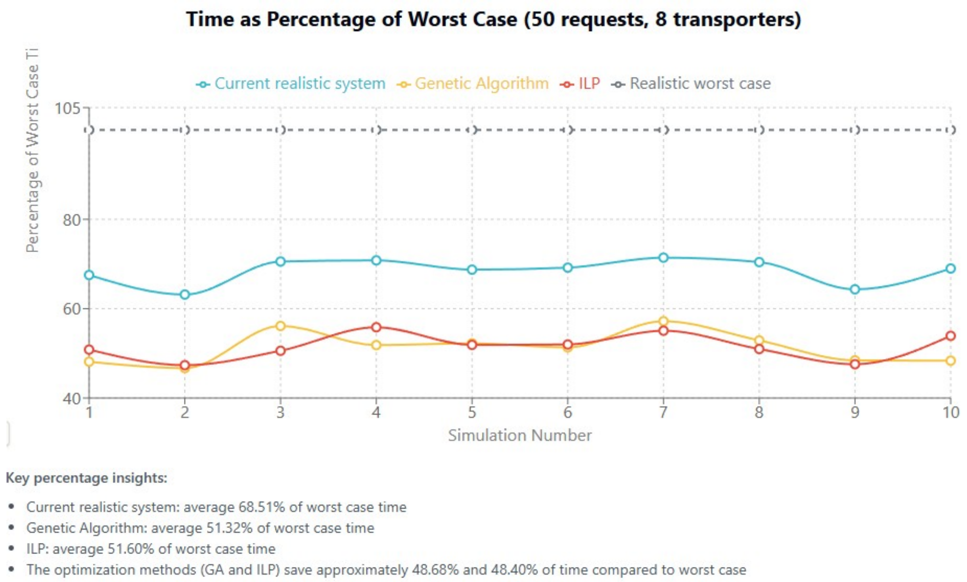
Figure 5.5: Same as the figure above but in percentage

Another simulation was run using eight porters and generating 50 request. The y-axis demonstrates the completion time, and the x-axis highlights the simulation number. As seen in the figure both algorithms showed significant improvement over the median and worst case of 1000 runs of the current realistic system.

Key insights:

- The current realistic system consistently takes longer (average: 108.98 minutes)
- Genetic Algorithm shows the best performance in most simulations (average: 81.60 minutes)
- ILP performs similarly to Genetic Algorithm (average: 82.07 minutes)
- Worst case scenario is significantly higher (max: 172 minutes)

Figure 5.6: Completion time for each simulation with eight porters and 50 requests, the realistic algorithm line is the median of 1000 runs, the worst case is the worst run of the 1000 realistic runs.



Key percentage insights:

- Current realistic system: average 68.51% of worst case time
- Genetic Algorithm: average 51.32% of worst case time
- ILP: average 51.60% of worst case time
- The optimization methods (GA and ILP) save approximately 48.68% and 48.40% of time compared to worst case

Figure 5.7: Same as the figure above but shown in percent

5.2.2 Analysis of Workload

One Key Performance Indicator that was studied was the standard deviation of workload balance. The y-axis demonstrates the workload index and the x-axis showcases the simulation number. The workload balance signified how the workload was distributed across each porter with a lower workload standard deviation indicating a fair workload.

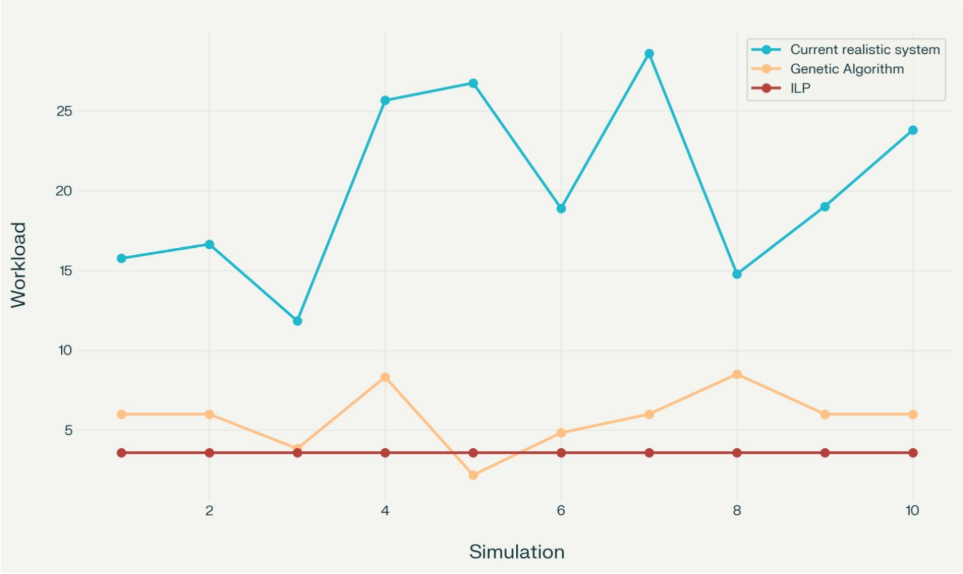


Figure 5.8: Standard deviation of workload in minutes between transporters in the ILP, GA, and median run of the 1000 runs of the realistic algorithm

5.3 Identified Implementation Barriers

A few of the anticipated challenges for an AI-driven patient transport optimization system are described below. They cover technical prerequisites and modeling considerations. Organizational and staffing challenges are noted as important areas that warrant further investigation beyond the scope of the current work. This discussion also draws upon insights from direct interviews regarding the current system and perceived needs.

5.3.1 Technical Challenges

Many prerequisites for an AI-based patient transport optimization system are already in place at Sahlgrenska University Hospital - a system for job entry (Columna) and historical transport data. However, some technical aspects are still critical for deployment.

A key input data missing for a dynamic, real-time optimization system is the live position of porters within the hospital. Accurate and continuous porter location tracking is required for the AI to make informed decisions regarding optimal task assignment based on proximity and availability.

Besides, expert consultation (Marco L. Della Vedova, Section 3.2.3 - Expert Consultation) stressed that the most important thing for an AI implementation here is an accurately modelled representation of the hospital environment. This includes physical layout (graph model) and realistic travel times between locations. One such model is envisioned as being developed iteratively:

1. Initially developed quantitatively, based on historical transport data to establish routes and baseline travel times.
2. Subsequently, it is refined qualitatively, incorporating expert knowledge from staff who are familiar with the hospital's nuances and common pathways.
3. Finally, potentially improved and maintained dynamically through machine learning techniques, such as reinforcement learning, where the model adapts over time based on real-world operational feedback and observed transport durations.

Addressing these aspects, acquiring real-time porter location data and ensuring a high-fidelity, adaptable hospital model, are paramount technical challenges.

5.3.2 Organizational and Staff-Related Challenges

Organizational and staff challenges related to an AI-driven patient transport system are extensive to analyse in detail. It is a big task covering human factors, change management, and workplace culture issues and while some of these areas are touched upon in the interview with Hospital staff (Section 5.3.3), such as staff acceptance, workload perception, and the role of supervisors, further exploration of mitigation strategies for these organizational barriers is recommended for future work.

5.3.3 Results from interview at Sahlgrenska University Hospital

The interview that was carried out at Sahlgrenska University Hospital with Magnus Martinsson gave broader insights to the project. In the interview Martinsson states that at Sahlgrenska they lack certain parts within their software, for example the transporters do not have the possibility to see the history of a what they have done throughout the day and the software can not create any kind of queue system for the transportations. The statistics are also very shallow because there is no grading system for how hard a transport is, it only calculates number of transports and not the distance, kind or elevation. Because of this Martinsson states that the employees finds it to be an unfair distribution of workload since some people only chooses the transportations that are easier and more fun. This also makes it inevitable for arguments and other disputes to happen when higher authority are not at the scene. Martinsson says that some transports have a higher priority but that they still are late because the transporters do not want to take it which means that the transport gets delayed. In that case he says that they would benefit from a system that distributes all of the transports to be more efficient, Martinsson contends the systems that Sahlgrenska has are simple, ineffective and limited.

In the interview Martinsson said they have experienced a little bit of problems with the firewalls in the software system that ultimately slows his work down. The stronger firewalls were implemented a few years ago because of a hacker attack to keep personal data from being breached. Sahlgrenska is however working to be able to lift some of their firewalls to easier move data from within the hospital and still keep it anonymous from the outside or anyone who does not have the authority to see it. The automatization is still not enough and needs to be tended to, which more suitable AI could help with and make more effective. Martinsson says that if the AI is trained well it could help Sahlgrenska with equal work distribution, which would lead to less arguments about unfair workload and make the transporters more satisfied. It could also help with predicting future transports and make them flow more seamlessly to increase efficiency and time management. Martinsson says that if the efficiency is improved then Sahlgrenska would save more money because of punctuality. The queues for medical attention everywhere in the hospital would decrease and thousands of kroner in delays would no longer be a big issue.

There are however downsides that Martinsson also states could be, lack of social contact with authority and that the people who cheats the system probably will find a new way to do it even with AI in charge. He says that for AI to be able to work you would still need a foreman that oversees the transporters work and makes sure that everything is running smoothly. Martinsson claims that if you implement and use AI in the right way (with transparency, overall functions and correct data) then there is a big possibility that it could work.

6

Analysis and Discussion

For this chapter, the goal is to provide insightful arguments and discuss the different simulations and what these might imply. Further more different optimization algorithms will be discussed to provide which scenario's they are suitable for. The potential impact of AI implementation will be discussed were workload distribution for both ILP and GA will be compared to the current system at Sahlgrenska. A discussion concerning the ethics for the proposed model will be presented as well.

6.1 Interpretation of Simulated Results

Comparing to the current system used at Sahlgrenska, both the ILP and Genetic Algorithm performed between 20-25 percent better than the median of Sahlgrenska. For smaller instances, e.g. when the amount of porters were 1-3 the ILP provided results better than GA and current system. This is because ILP solvers are designed to find the exact optimal solution to problems. For instances where the computation time increased, such as 6-9 porters GA resulted in better results. Compared to ILP, GA are heuristic by nature the computational time does not get affected by the same degree as for ILP.

6.1.1 Dynamic Simulation

For a simulation performed 10 times with 50 request made by 8 porters the GA achieved the fastest completion time 6 out of the 10 simulation made. The Genetic Algorithm also provided the fastest simulation time on simulation 10 which provided a impressive result of 76.4 minutes. Highlighting that for more complex and dynamic systems, the GA provide vastly better results.
Compared to the current system used at Sahlgrenska, the GA would save approximately 27.4 minutes on average transport which is a 25 percent improvement. The saved time could instead be utilized on improving patient care allowing for more efficient resource utilization. One thing to keep in mind is that the standard deviation for the GA was slightly higher than for the ILP, 3.64 minutes compared to 2.51 minutes which showcases that the GA can sometimes be less consistent than the ILP. Also, note how much better the optimizers are than the worst-case.

Key insights:

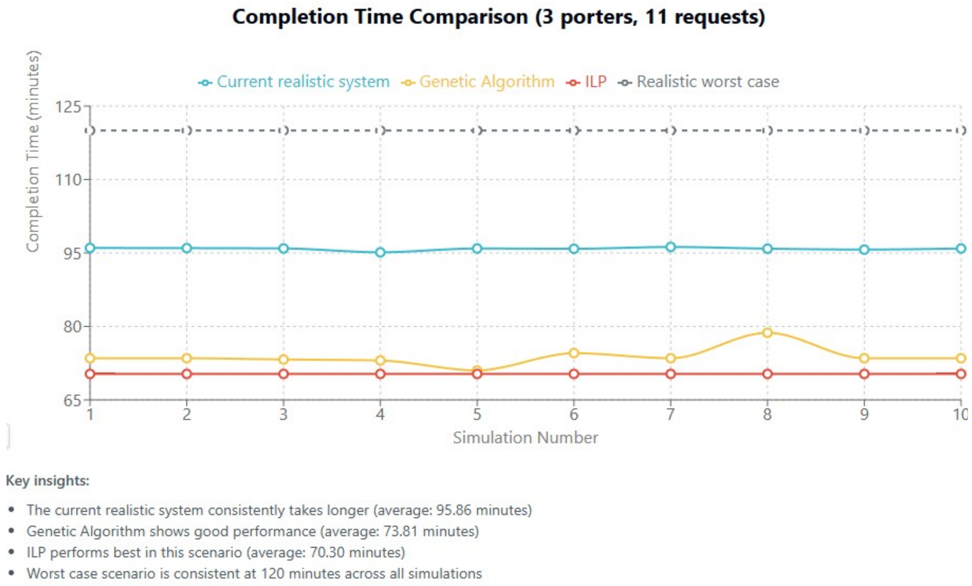
- The current realistic system consistently takes longer (average: 108.98 minutes)
- Genetic Algorithm shows the best performance in most simulations (average: 81.60 minutes)
- ILP performs similarly to Genetic Algorithm (average: 82.07 minutes)
- Worst case scenario is significantly higher (max: 172 minutes)

Figure 6.1: Completion times for each simulation, (current realistic system is the median of 1000 runs, and worst case is the worst run out of 1000)

6.1.2 Basic Simulation

Simulating 11 request made by 3 porters, the ILP performed better than the GA and current system at Sahlgrenska. For all 10 simulations, the ILP maintained 70.3 minutes in completion time whilst the GA averaged 73.81 minutes and the current system at Sahlgrenska averaged 95.82 minutes. The ILP performed 25.56 minutes better in average which is a 26.7 percent improvement. The ILP also maintained perfect consistency with a standard deviation of 0 minutes whilst the GA resulted in 1.94 minutes of standard deviation. The reason for the perfect standard deviation for the ILP is because ILP solvers are designed to find the optimal solution. For a convergent ILP, the exact optimal solution will be provided each time, while GA is heuristic, it will show varying results depending on mutation rates and other varying factors. An optimization that provides zero standard deviation will substantially improve predictable operation and could provide an accurate hospital model combined with ML.

Even here, it should be noted how significantly the optimizers outperform the worst-case scenario. By using optimizers we get a more reliable system with a substantially better worst case.



Key insights:

- The current realistic system consistently takes longer (average: 95.86 minutes)
- Genetic Algorithm shows good performance (average: 73.81 minutes)
- ILP performs best in this scenario (average: 70.30 minutes)
- Worst case scenario is consistent at 120 minutes across all simulations

Figure 6.2: Completion time for each simulation with three porters and 11 requests (current realistic system is the median of 1000 runs, and worst case is the worst run out of 1000)

6.2 Potential Impact of AI Implementation at Sahlgrenska

Both ILP and GA outperformed the current system at Sahlgrenska when the emphasis was on completion time for patient transports. One thing both optimization algorithms improved was the workload balance. This metric shows how the work is distributed across each porter, a lower number indicating more even distribution of work. For 11 request and 3 porters, ILP maintained a standard deviation workload of 3.57 across all simulations. While the GA had a standard deviation of 5.76. Both algorithms outperformed the current realistic system which had a standard deviation of 20.18 indicating an unfair workload currently at Sahlgrenska University Hospital. For the ILP the improvement of the standard deviation approximately 81 percent was achieved and for GA approximately 70 percent. Indicating that the variance in workload was greatly improved. This is a substantial improvement which would permit a more equitable workplace and possibly prevent potential burn-outs. Having a predictable workload would benefit employers long-term, and possibly lowering the amount of overtime needed as the transportation requests are now being handled more effectively.

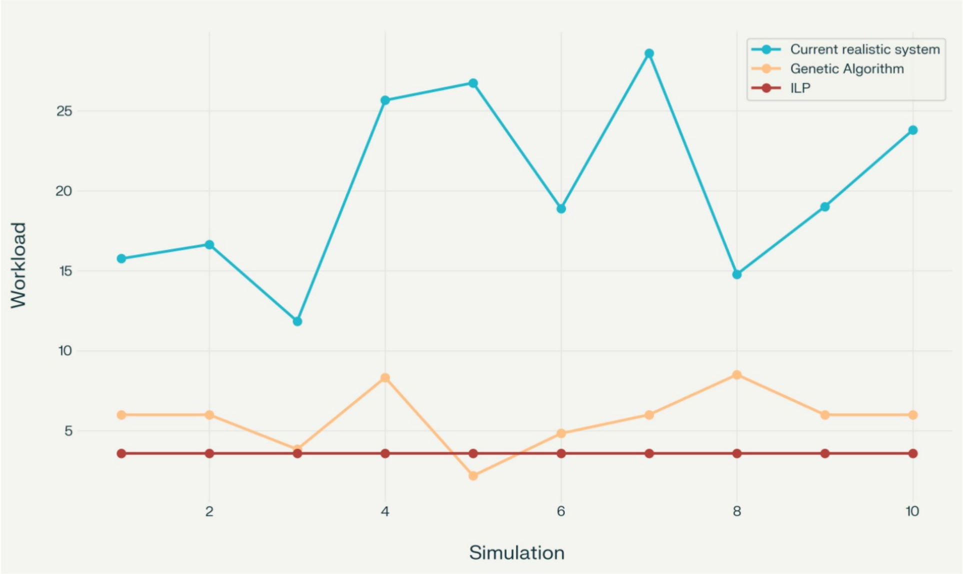


Figure 6.3: Workload distribution for each simulation

6.3 Addressing Ethical Implications in the Proposed Model

Ethics for the proposed model is crucial for designing a lasting sustainable model. This was mentioned for 2.4. Ethical Considerations for AI in Healthcare. Outside insuring that patient information is properly secured in regards to HIPPA and GDPR, further data security policies to prevent data breaches might be needed to be implemented. While ILP offers transparent mathematical formulas where the constraint made explicit, GA can sometimes provide a "black box" solution being that the algorithms optimal solution varies depending on each generation that converges. Although the algorithms remove the burden of decision-making on the porters, human oversight must be provided. There needs to be a clear responsibility for the algorithmic outputs.

6.4 Alternative perspectives and economical aspects

During the process of this study the group had the opportunity to speak with Maria Holmström at Capio Ortho Center in Gothenburg. Holmström noted that although AI is not currently used to the same extent as at Sahlgrenska University Hospital, its implementation could still offer significant benefits. Holmström explained that Ortho Center, being a private clinic, does not face the same level of queue-related challenges as Sahlgrenska. However Ortho Center offloads Sahlgrenska by accepting some of their patients, which reduces the pressure on public healthcare queues. Holmström also acknowledged that the use of AI could help shorten wait times and improve overall logistics and efficiency. It would be possible to use the genetic algorithm described in 2.1.1.2. Complexity for Genetic Algorithms on a clinic such as Ortho Center.

However with this in mind a combination of algorithms - such as the ones explained in subsection 1.2.4. Related work - would make it suitable to implement AI to help streamline hospital transport at Sahlgrenska. In return it would shorten hospital queues, improve punctuality and transport coordination. Not only would this happen but it would also bring notable economic changes, shorter waiting times and more efficient resource allocation all contributes to reduce cost. This was also brought up by Magnus Martinsson in an interview, where he believed that thousands of kroner could be saved, which is directly correlated to the optimization of logistical operations. In the pervious subchapter 6.1.2 the ILP was proven to perform 26.7% better than than average, which means that in the long run Sahlgrenska could optimize resource utilization with fewer staff. Sahlgrenska could maintain the quality of their care but with a smaller workforce, which lowers the personnel expenditures.

7
Conclusions

7.1 Summary

This study has explored how artificial intelligence can be used to improve efficiency, fairness and coordination of internal patient transport at Sahlgrenska University Hospital. Through simulation models built on Genetic Algorithms (GA) and integer linear programming (ILP), notable gains in both completion time and workload distribution were realised compared to the hospital's current system. ILP showed consistent optimal performance in simpler scenarios, while GA offered advantages in more complex and dynamic settings.

The results indicate that a coordinating AI system could on average reduce transport times by over 25%, balance the porters workloads by up to 80% and provide greater predictability and fairness for the porters. These benefits not only support improved operational efficiency and cost savings but also contribute to better working conditions for staff and shorter waiting times for patients.

However, for successful implementation there are several challenges to be assessed. These include data integration with existing systems around the hospital, patient privacy and ensuring ethical considerations such as task allocation and human oversight. While technical and organizational barriers remain, our results show that with good planning and investment AI has strong potential to improve hospital logistics.

7.2 Answers to Research Questions

This study was guided by four primary research questions. The following section provides concise answers to each, based on the analyses and discussions presented in this thesis.

7.2.1 Answer to RQ1

Research Question 1: *How can an AI-based prioritization model be designed and integrated into an assignment strategy to manage transport requests more effectively than current practices, considering factors such as medical urgency, waiting time targets, resource availability, and estimated transport duration?*

Answer: An AI-based prioritization model can be effectively designed and integrated into an assignment strategy using the optimization algorithms Integer Linear Programming (ILP) and Genetic Algorithms (GA). As demonstrated in this study (Chapter 4), these models can be designed to incorporate factors like resource (Porter) availability, current locations, and accumulated workload. They can also utilize estimated transport durations, derived from pathfinding algorithms like Dijkstra's on a hospital graph model, and optimize for objectives such as minimizing overall completion time (makespan) or balancing workload, which both indirectly and directly address efficient request management.

The simulation results (Chapter 5, Section 5.2; Chapter 6, Section 6.1) show that such AI-driven strategies handle transport requests better than the baseline representing current practices. Specifically, ILP and GA models improved average transport completion times by about 20-27%. This implies that systematically processing real-time (simulated) information and optimizing assignments according to a global view of requests and resources could enable AI models to achieve timely patient transport. While the input data (priority levels) included medical urgency and specific waiting time targets, their explicitly dynamic effect on prioritization beyond the intrinsic optimization of makespan and workload is an area of further refinement in more advanced models.

7.2.2 Answer to RQ2

Research Question 2: *To what extent can AI-driven assignment algorithms (specifically ILP and GA) facilitate a more balanced and equitable workload distribution among patient transport staff compared to the current system, as evaluated through simulation?*

Answer: AI-driven assignment algorithms like ILP or GA facilitate a more equitable and balanced workload distribution among patient transport staff than in the present system. Analysing the current state at Sahlgrenska (Chapter 5, Section 5.1.3) revealed workload imbalances during peak hours. Simulation results (Chapter 6, Section 6.2, Figure 6.3) showed that ILP and GA, designed with workload consideration in their optimization functions (Chapter 4, Section 4.5 and 4.6), obtained a much more equitable distribution of tasks. In contrast to the baseline model representing current practices, both the ILP model and GA showed substantial improvement in the deviation of workload. This implies that AI could assign tasks to reduce natural unevenness inherent to manual and less structured assignment processes, thereby ensuring fairness of conditions for staff.

7.2.3 Answer to RQ3

Research Question 3: *What mechanisms can be incorporated into AI optimization models to implicitly or explicitly consider staff rest or workload recovery, and how might these impact overall system efficiency and fairness?*

Answer: AI optimization models could have mechanisms, including implicit and explicit staff rest and workload recovery considerations. The current simulation model (Chapter 4, Section 4.6) implicitly models transporter workload as a sum of effort. The models implicitly manage overall strain by including workload balancing as a key objective in the ILP and GA assignment strategies. They may influence the natural cadence of porters returning to a designated lounge for rest, making these breaks more timely or equally distributed relative to exertion. Routing and assignment may also reduce unnecessary travel and thus reduce fatigue.

Not explicitly implemented and tested for dynamic rest scheduling in the current iteration of this study's simulation, mechanisms that could be implemented include setting maximum workload thresholds per porter per timeframe, including predefined or dynamically determined break schedules in assignment constraints, and developing more sophisticated agent models for fatigue simulation and recovery, where the AI could prioritize rest for porters exceeding some fatigue threshold. Such mechanisms likely will induce a trade-off: explicit rest scheduling may slightly reduce short-term throughput or increase makespan when porters are offline during peak demand. However, excessive fatigue would enhance fairness over the long term, reduce burnout, and perhaps ensure a more consistent service quality level. The workload balancing demonstrated in this study (RQ2) suggests fairness improvement.

7.2.4 Answer to RQ4

Research Question 4: *What are the primary anticipated technical barriers (e.g., data integration, system compatibility, real-time requirements) and ethical considerations for implementing an AI-driven patient transport solution at Sahlgrenska, and what strategies could mitigate these technical barriers?*

Answer: The main technical barriers to an AI-driven patient transport solution at Sahlgrenska (Chapter 5, Section 5.3.1) are continuous live location tracking of porters and real-time integration with the transport request system Columna. Another barrier is developing and maintaining a high-fidelity, dynamic digital representation (graph model) of the hospital environment, including pathways, travel times, and disruptions. Assisting the AI system with existing hospital IT infrastructure, including firewalls and data security protocols, is also a challenge. Solutions to these technical barriers (Chapter 5, Section 5.3.1) include: investing in appropriate location tracking technology, iterative development of the hospital model (possibly including machine learning for dynamic updates), and providing robust APIs and middleware for system integration alongside compatibility testing in a phased rollout.

The primary ethical considerations (Chapter 2, Section 2.4; Chapter 6, Section 6.4) concern the protection of sensitive patient and staff data by GDPR and Patientdatalagen. It is also important that AI does not perpetuate or introduce biases in task assignments and that workload distribution is just and equitable. Knowing how the AI decides, especially for complex models, and defining clear lines of responsibility for AI-driven outcomes (transparency & accountability) are critical. Managing automation bias, preventing skill degradation among staff, and addressing concerns regarding job roles and autonomy (human oversight/staff impact) are also important ethical points. Strategies to address these ethical considerations include robust data anonymization and security protocols, regular auditing of algorithms for bias, designing systems that allow human oversight and intervention, and training staff throughout the design and implementation process.

7.3 Future research

Future research may combine machine learning for demand forecasting, transport duration estimation, and adaptive scheduling. Deeper studies of organizational barriers, such as staff acceptance, training needs, and workflow changes, are also needed. Step one would be to make the simulation model a Digital Twin of Sahlgrenska's patient transport system. Those integrations include real-time porter location tracking, live updates from the Columna system, and, potentially, the mentioned machine learning models for dynamic recalibration of travel times and demand forecasting. This advanced system extends this study's offline benchmarking to real-time operational decision support and continuous improvement. Tests of enhanced AI systems like a Digital Twin prototype in real-life hospital environments will also determine practical feasibility and refine models for day-to-day operations in a dynamic healthcare environment.

End result: An AI-based system for coordinating patient transport may be the first step towards a resilient, efficient, and patient-centred healthcare system at Sahlgrenska and/or other hospitals with similar internal patient transporting structures.

Bibliography

[1] D. Overman, "Ai platform increases efficiency at tampa general hospital," *AXIS Imaging News*, 2022.

[2] M. A. Abid, S. A. Mahdiraji, F. Lorig, J. Holmgren, R.-C. Mihailescu, and J. Petersson, "A genetic algorithm for optimizing mobile stroke unit deployment," in *27th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems (KES 2023)*, vol. 225 of *Procedia Computer Science*, pp. 3536–3545, Elsevier B.V., 2023. Open access under CC BY-NC-ND license.

[3] J. A. Oliveira, J. Ferreira, L. Dias, M. Figueiredo, and G. Pereira, "Non emergency patients transport - a mixed integer linear programming," in *Proceedings of the International Conference on Operations Research and Enterprise Systems (ICORES)*, pp. 262–269, SCITEPRESS, 2015.

[4] M. Kühn, T. Baumann, and H. Salzwedel, "Genetic algorithm for process optimization in hospitals," in *Proceedings of the 26th European Conference on Modelling and Simulation (ECMS 2012)* (K. G. Troitzsch, M. Möhring, and U. Lotzmann, eds.), (Koblenz, Germany), pp. 250–253, European Council for Modelling and Simulation, 2012.

[5] M. Fogue, J. A. Sangüesa, F. Naranjo, J. Gallardo, P. Garrido, and F. J. Martínez, "Non-emergency patient transport services planning through genetic algorithms," *Expert Systems with Applications*, 2016. Preprint submitted on May 18, 2016.

[6] A. Vie, A. M. Kleinnijenhuis, and D. J. Farmer, "Qualities, challenges and future of genetic algorithms: a literature review," *arXiv preprint arXiv:2011.05277*, 2021.

[7] P. Dadone and H. VanLandingham, "Genetic algorithms for tuning plc loops," in *Proceedings of the 1999 IEEE Midnight-Sun Workshop on Soft Computing Methods in Industrial Applications (SMCia/99)*, (Kuusamo, Finland), IEEE, June 1999.

[8] C. Yan, N. McClure, S. P. Dukelow, B. Mann, and J. Round, "Optimal planning of health services through genetic algorithm and discrete event simulation: A proposed model and its application to stroke rehabilitation care," *MDM Policy & Practice*, vol. 7, no. 2, p. 23814683221134098, 2022. Open access under CC BY-NC 4.0 license.

[9] T. Kropp, Y. Gao, and K. Lennerts, "Data-based optimisation of intra-hospital patient transport capacity planning," *OR Spectrum*, 2024. Open access.

[10] R. E. Bixby, "A brief history of linear and mixed-integer programming computation," in *Documenta Mathematica, Extra Volume ISMP*, pp. 107–121, European Mathematical Society Publishing House, 2012.

[11] G. B. Dantzig, "Linear programming," *Management Science*, vol. 9, no. 3, pp. 191–202, 1963.

[12] E. Ekin, "Solution approach to p-median facility location problem with integer programming and genetic algorithm," *Afyon Kocatepe University Journal of Social Sciences*, vol. 26, no. 2, pp. 547–562, 2024. Submitted: June 2022; Accepted: September 2023.

[13] P. E. Black, "Np-hard," in *Dictionary of Algorithms and Data Structures [online]*, U.S. National Institute of Standards and Technology, 2021. Entry modified 5 January 2021.

[14] W. Jin, Z. Hu, and C. Chan, "An innovative genetic algorithms-based inexact non-linear programming problem solving method," *Journal of Environmental Protection*, vol. 8, no. 3, pp. 231–249, 2017.

[15] D. Landsman, H. Ma, J. Knight, K. Gough, and S. Mishra, "A flexible integer linear programming formulation for scheduling clinician on-call service in hospitals," *arXiv preprint arXiv:1910.08526*, 2019.

[16] AI Sweden and Region Västerbotten, "Syntetisk data inom intensivvård," tech. rep., AI Sweden, Feb. 2022.

[17] R. Challen, J. Denny, M. Pitt, L. Gompels, T. Edwards, and K. Tsaneva-Atanasova, "Artificial intelligence, bias and clinical safety," *BMJ quality & safety*, vol. 28, no. 3, pp. 231–237, 2019.

[18] D. S. Char, N. H. Shah, and D. Magnus, "Implementing machine learning in health care—addressing ethical challenges," *New England Journal of Medicine*, vol. 378, no. 11, pp. 981–983, 2018.

[19] P. M. Rothwell, "Factors that can affect the external validity of randomised controlled trials," *PLoS clinical trials*, vol. 1, no. 1, p. e9, 2006.

[20] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*. Cambridge, Massachusetts: MIT Press, 4th ed., 2022.

[21] G. Olaoye, R. Faith, J. Badmus, and H. Luz, "Machine learning for healthcare resource allocation," *Health Education and Public Health*, 2024.

[22] Q. Wang and C. Tang, "Deep reinforcement learning for transportation network combinatorial optimization: A survey," *Knowledge-Based Systems*, vol. 233, p. 107526, 2021.

[23] M. Ruiz, E. Orta, and J. Sánchez, "A simulation-based approach for decision-support in healthcare processes," *Simulation Modelling Practice and Theory*, vol. 136, p. 102983, 2024.

[24] J. B. Jun, S. H. Jacobson, and J. R. Swisher, "Application of discrete-event simulation in health care clinics: A survey," *Journal of the Operational Research Society*, vol. 50, no. 2, pp. 109–123, 1999.

[25] T. M. Machado and F. T. Berssaneti, "Literature review of digital twin in healthcare," *Heliyon*, vol. 9, no. 9, p. e19390, 2023.

[26] G. Pellegrino, M. Gervasi, M. Angelelli, and A. Corallo, "A conceptual framework for digital twin in healthcare: Evidence from a systematic meta-review," *Information Systems Frontiers*, 2024. Advance online publication.

[27] J. Hasell, “Measuring inequality: what is the gini coefficient?,” *Our World in Data*, 2023. https://ourworldindata.org/what-is-the-gini-coefficient.

A

Appendix

Figure A - Interview at Sahlgrenska with Magnus Martinsson

Intervju med Magnus Martinsson på Sahlgrenska

[Sara]
Jag tänkte först fråga lite frågor, lite öppna frågor och sen kommer jag förklara våran idé och sen har jag lite till frågor. Men projektet handlar om hur vi kan använda AI för att effektivisera patientflöden och vi har valt att fokusera på patientvaktmästeriet eller transporten av patienter. Så det är där vi har lagt vårt fokus.

Men så jag tänkte börja, finns det några större problem inom patientvaktmästeriet som hade kunnat underlättas? Om du kan komma på några stora problemområden? Det får ju även vara sådant som inte AI hade kunnat hjälpa.

[Magnus]
Alltså det som saknas i IT-systemet som vi har är turordningslista och historiklista. Du kan ju inte se vad du själv har gjort under dagen, det finns ju inte liksom. Och det finns inga funktionssystem som fördelar uppdragen.

Den lösningen finns i andra system, antingen att man fördelar, men den är väldigt automatisk. Det är egentligen ingen AI-funktion, den bygger på en enkel algoritm, alla de system som har det. Precis.

Men det skulle ju lösa problem som finns med konflikter, att folk tycker att det är ojämnt fördelat. Eller att någon har jobbat mer än någon annan, eller mindre. Och om det var en bra samfunkomst skulle man även kunna fördela till exempel huden, transporten av någon som är för backen, jämt.

Precis. Så att ingen kommer undan.

[Sara]
Nej, det vet jag själv.

[Magnus]
Sen har vi problem med logistiken, det är ju några försörjningar som ska hämtas. Men det viktigare är att vissa försörjningar är tidiga jämfört med andra. Och de är ju oftast inte i närheten här från centralen.

Och vi har ju svårt att hinna med en retur och komma fram i tid för att hämta den som faktiskt behöver vara i tid sen. Då hade det varit bra med en funktion som faktiskt räknar ut det. Jag har ju suttit där ute och försökt göra det. Det tycker jag är jättesmart, men det blir hela tiden att man hamnar efter. Man är ju nöjd, klockan är redan tio i två, nu borde den redan vara i väg. Nu har jag inte ens skickat någon åt det hållet.

[Sara]
En förebyggare på något sätt.

[Magnus]
Så man hade en funktion som planerade jobbet, det hade varit jättebra. Och som sagt, det finns ingen sån funktion på marknaden alls. Utan den är ju ganska dum.

Den som skulle ha varit i millennium, som inte blir av som det ser ut nu. Den hade bara fem regler, tror jag. Kartan med platser som man själv får lägga in och konfigurera hur nära varandra de är.

Så att den hade den informationen. Och sen innan tio minuter, om du checkade ut på en plats och du är klar 137. Så innan tio minuter så kan systemet skicka nästa körning till dig.

Beroende på hur nära den är. Ja, precis. Det är två regler.

Och sen vet systemet att om det finns fyra körningar så ska den gå efter prioriteringslistan. Så den som ska gå till operation fördelar den först. Och den som tog sig fördelar den allra sist.

Men den har ingen framförhållning alls. Och finns det ingen tillgänglighet så händer ingenting. Den försöker inte säga att vi behöver hjälp här och där.

Det har ju kolumna den funktionen. Vi kan göra körningar orange så att folk ska plocka dem och se att det händer något. Så det var ganska dumt.

Det enda den gjorde var att fördela.

[Sara]
Men den blev lite så platt.

[Magnus]
Den hade inte hjälpt alls med förseningar. Den blev nästan beroende av att vi skulle komma någonstans för att få nästa körning.

[Sara]
Ja, men det är jättebra. Hur funkar statistiken över körningar? Jag har hört någon gång att akuten röntgenkörning räknas samma som en huden till spirokörning i systemet.

Men det kan ju minnas fel.

[Magnus]
Ja, det stämmer ju tyvärr. Så jag börjar alltid med att kolla. Om jag ska kolla arbetsfördelningen så ser jag först hur många körningar har någon gjort.

Och sen får jag göra ett stikprov så att jag inte tycker att någon är fantastisk. Men som älskar akuten som bara kör de här fram och tillbaka.

[Sara]
Ja.

[Magnus]
Det är inget fel med det som sagt, men jag skulle inte få fel intryck på det. Det systemet vi har nu har ingen som helst avståndsfunktion. Det är bara en lista med avdelningsnamn.

[Sara]
Det är ju bara mängdkörningar egentligen. Ja.

[Magnus]
Så det finns inget så här. Det är en statistikundervarande. Det är bara namn på avdelningarna och en ID på avdelningarna.

Men det finns ingenting som säger var någonstans avdelningen är. Så jag kan inte använda den heller för att få fram någonting. Utan att lägga in någon så här massiv Excel-fil i Power BI.

Det har jag inte orkat till. Det förstår jag. Så det saknas.

Vi saknar ju till och med en bra statistik. Jag gör den själv oftast.

[Sara]
Ja, för det är ju något man kanske hade kunnat automatisera.

[Magnus]
Vi har några grejer i Power BI. Det är mest ekonomi. Stora siffror liksom.

Jag måste göra min egen tabell om jag ska se hur mycket var det att göra i vardags. Så gör jag alltid egna tabeller från datorn.

[Sara]
Mycket manuellt jobb då?

[Magnus]
Ja, pivåtabeller.

[Sara]
Har du använt AI tidigare?

[Magnus]
Det beror på vad definitionen är. Vi använder väl CoPilot. Det är den enda vi har egentligen här.

Men det är väldigt enkla funktioner som vi har använt hittills. Det är så här då. Till exempel hjälpmedelsförrådet.

Om man skannar in Forms-formulär så flyttar den datan automatiskt och fyller i en Excel-fil också. Men det är liksom bara sådana här enkla grejer. Ja, det funkar.

Annars så har jag den här jobbansökan från ChatGTB som jag gjorde själv. För att vi skulle ha koll på vilka som använder den när de söker jobb. Nu är det typ alla, så nu har jag slutat läsa dem.

[Sara]
Jag vet, Lina läste upp och man hörde så tydligt.

[Magnus]
Jag vill ge mer av mig själv till vården. Annars är det inte så mycket mer. Jag tror vi har försökt någon gång.

Vi har lite problem med brandväggar i vårt system. Det är väldigt låst. Man gillar inte att kommunicera.

Det kan vara svårt att nå en fil. Tänk att jag har en fil hos mig någonstans, SharePoint eller liknande. Då vill jag ha det som fullstatistik från den filen.

Det kommer aldrig att fungera. Jag måste ha den någonstans där det är helt öppet. Det i sin tur gör att de måste sitta och anonymisera data för att jag ska kunna lägga ut den.

Så blir det ändå att jag ska sitta och jobba med massa grejer. Jag tror vi har försökt med vissa sådana här läsningar. Så jämnar man upp lite på grund av de här murarna som ligger.

Det är efter ett hackerangrepp för några år sedan. Så har vi ganska jobbiga sådana här brandväggar. Men det har ju börjat lite nu med att vi ska lyfta in de här läsningarna istället.

Så jag tror det är något mer på gång än kopparligt. [Sara]

Spännande.

[Magnus]
Så om man har det innanför brandväggen så ska det vara lättare att hantera. [Sara]

Ja, så tänkte jag förklara lite vår idé då. Det är ju så som du sa innan att man använder en AI i den appen som finns nu på telefonerna. Och att den fungerar lite som en förman och delar ut jobb.

Men att den då kan ta in, alltså den använder all statistik och körningar och sådant innan. Så den kan räkna ut om personen har kört väldigt många tunga körningar kanske den får lättare körning. Och den kan även se att det kommer komma körningar från röntgen snart.

Så att den kan tänka i förväg precis som en vanlig person kan göra lite så. Så den ska egentligen kunna fungera som en förman och dela ut jobb. Så att folk inte behöver prioritera jobb själva när man är ute och sådant.

Eller bestämma själv när man ska gå ut till exempel på helgen och sådant utan den bestämmer åt dig. Men då ska den också veta om att du ska få sitta och vila ett tag. Men det är ju tanken egentligen.

Sen kommer vi inte bygga någon AI utan mer fokuserar lite på hur algoritmerna hade kunnat se ut.

[Magnus]
Som en kravlista ungefär? Ja, precis.

[Sara]
Men så det är egentligen grundidén. Men tror du att det hade kunnat underlätta eller gör det bara mer komplicerat om man hade använt AI på det sättet? Jättesvår fråga, men...

[Magnus]
Nej, alltså om man verkligen... Man får ju ta det lite på allvar så att säga. Så då måste man ha alla de här.

Man måste ha helheten och då fungerar det. Men om man på vägen så att säga behöver... Nej, men den klarar inte av att göra det, den klarar inte av att göra det.

Eller vi vet inte hur vi ska få den att lära sig det. Ju mindre man gör den desto sämre kommer den att funka. Till slut blir det som den här millennium.

Som inte är särskilt smart egentligen då. Så man kan hålla den kravlistan nedlanda hela vägen. Då kommer det absolut fungera, tror jag.

Och det kommer ju vara... Då kommer ni ju även då, som sagt, kunna förutse och jobba med en bredare intelligens.

[Sara]
Precis.

[Magnus]
Eller bättre fokus kanske är bättre ord. En en person kan göra så att den ska sitta och hålla reda på alla trådar samtidigt. Men man har inte det fokuset.

Det är liksom ingen som har det. Sen tror jag ändå att man kan utveckla ännu längre genom att testköra.

[Sara]
Ja. För det är ju tanken att man i sådana fall skulle använda gammal statistik och sen egenpåhittad. För att köra den så mycket som möjligt.

För desto mer den körs desto mer lär den sig själv. Så att man ser till att den prioriterar rätt och mycket sånt. Men vad tror du hade varit de största fördelarna om man hade använt en sådan modell?

[Magnus]
Jämfört med idag?

[Sara]
Ja. Om det finns ställen du tror att det hade hjälpt. [Magnus]

Det stora fördelen hade varit att när det är mycket att göra, då hade den kunnat prioritera rättkörningar åt alla som jobbar där. Så det är inte bara arbetsledare utan även de som plockar själva. Och jag har ju jättesvårt med det och jag förstår det verkligen.

För när man scrollar så tar det som aldrig slut. Och så ska du hitta någon som är jätteviktig i den rören. Samma sätt är det när det ringer en massa frågor och du har en sån här stort fönster med rutor.

Man ska hitta någonting där, det är nästan en omöjlig uppgift. Samtidigt så är den punktligheten där, den är ju värd tusentals kronor för varje körning. Om vi lyckas komma i tid då, även om vi ligger i underläge. Så det hade ju varit den stora fördelen, tänker jag. Man har det som en hjälpare som är lite övermänsklig där.

[Sara]
Men tror det finns lite nackdelar? Vad är de största farorna man använder en sådan AI? [Magnus]

Jag kan komma ihåg lite olika saker. En initial nackdel kan ju vara att den bara upprepar mönster som redan finns och den hjälper inte alls. Med någon förbättring, den bara tycker att det är så här det ska vara, det säger min data.

Och så fortsätter den att lämna av körningar för senare till exempel, eller några såna här dumgrejer egentligen. Att den bara fortsätter som det har varit, att den förändrar ingenting. Sen finns det arbetssociala nackdelar, eller faror, som kan vara överdrivna.

Men när du jobbar någonstans och du ska vara här i nio timmar med rast, så pratar du inte med någon. Du pratar givetvis med patienter och sköterskor och så, men du pratar inte med någon som leder ditt arbete. Det skapar en psykologi som oftast upplevs som negativ.

Då måste man lösa det här på något sätt. Man måste ha något socialt sammankopplingspunkt.

[Sara]
Det var min nästa fråga också. Jag trodde att arbetsmoralen hade förändrats om AI blev implementerat, och på vilket sätt positivt och negativt. Då tänker jag att om vissa kanske jobbar lite mindre än andra, den delen kan ju bli positiv.

Men det kan också vara negativt att det inte är lika lätt att smita undan, utan man blir tvingad att jobba mer än vad man tycker att man borde.

[Magnus]
Det som vi kallar mufflare, folk som vill undvika tjänningar på olika vis, de får bara ett nytt sätt att arbeta. Ingen vill ha ett AI-system som ger dig uppdrag innan du är klar med något annat. Då kommer man bara inte göra sig klar.

Så får man sin lediga tid. Det kommer helt enkelt inte förändra något, så det är fortfarande jag som chef som måste följa upp de som ligger efter. Sen kommer jag ha den statistiken på något sätt och kanske är bättre än innan.

Men det kommer inte automatiskt lösa det problemet. Sen är frågan, jag tror inte man ska fokusera på dem, jag tror det är en större risk med dem som faktiskt jobbar hårt i dag och som gör det för att de får välja själv. Så man kan knäcka deras motivation lite.

När de inte längre får det, då blir inte de sådana superhjältar längre. För det är inte lika roligt att jobba.

[Sara]
Det är inte lika roligt att ta en hudre-körning om man inte har gjort det åt någon annan på något sätt. Jag tog den.

[Magnus]
De tog den när de ville det. Då var det jättelätt. Det är inte alls lika kul när de blir tvingade att göra det någon annan tid på dagen när de inte alls känner för det.

Och så sprider sig den känslan över hela dagen, att de inte själv får välja på samma sätt. De får ju inte välja till 100% idag, men det kan ju vara 40% som de får välja själv idag. När den försvinner helt, då är det inte lika roligt för dem.

Sen finns det ju andra som jobbar här, som hade stormtris och bara går runt och funderar på sina rabatter. Som bara vill att det bara ska plinga och så gör man som den säger.

[Sara]
Slippa tänka lite.

[Magnus]
De hade kanske triss mer. Det är ju ingen enkel bild. Det blir ju inte på antingen ett sätt eller ett annat.

Det blir nog på flera utfall samtidigt. Sen finns det ju en ovilja mot förändringar i hela samhället. Minsta villa fel så kommer man ju döma ut hela systemet.

Men det är inte alltid så heller. Om det finns små tydliga fördelar så köper man ändå förändringar. Beroende på vad fördelarna är.

Man lägger in till exempel att systemet planerar din rast. Då kan systemet vara mycket trevligare.

[Sara]
Det är ju också det att det behöver fungera otroligt säkert. Det har med sjukvård att göra också. Det får inte gå sönder någonstans eller göra fel för att det kan innebära livsvara i värsta fall.

Men det är ju också en risk med att använda det. Tror du att det hade varit en möjlighet i framtiden att AI blir implementerat i patienttransporter?

[Magnus]
Ja, det tror jag. Man har redan varit på väg mot automatisk tilldelning i 15 år. Och det har funnits olika dumma versioner av det.

Men det är absolut på väg.

[Sara]
Vet du om det är några problem eller avgränsningar från facket vi borde hålla koll på? Vår handledare

sa att vi ska vara noggranna med att inte använda viss statistik. Att inte facket ska komma och säga att det får ni inte göra.

[Magnus]
Jag kan inte tänka mig att det skulle vara som att man hänger ut hela personalgruppen som ineffektivare. Så det handlar mer om ordval. Eller att citera mig som bara en av sakerna jag sa att de hatar förändringar.

Annars skulle det komma under om ni intervjuar någon av dem också. Okej.

[Sara]
Men det finns ju inga namn.

[Magnus]
Nej, precis.

[Sara]
Om vi hade fått mäta och göra lite mer data på körningar. Då är det till exempel mäta tid någon är ute och kör versus tid man är inne och sitter. Då självklart anonymt, men det kan ju kännas lite gränsfall då.

Om det visar sig vara mer än man borde sitta så kanske det inte är jättebra att ha med i sådana fall.

[Magnus]
Jag tror att ni kommer att ha problem att få en bra bild av det. För att det finns ju... Hej!

Hej, sorry. Ipaden har inte laddat så jag gör det imorgon. Vad hade jag tänkt på nu?

Du har ju en person. Den personen har tid att göra vissa saker. En stor del av den tiden som man inte är igång och kör körningar är man på väg emellan.

På olika sätt. Så i datan kan man inte riktigt se när personen sitter i en soffa.

[Sara]
Nej, det hade ju fått bli att jag är här och mäter när jag själv är inne och sitter versus kör till exempel. Så det hade ju blivit lite jobbigt.

[Magnus]
Vi har haft studiepersoner förut, jag har aldrig varit med om att man måste godkänna. Jag tror arbetsgivarna har alla rättigheter att godkänna det åt. Men det ska vara anonymt.

[Sara]
Ja, med det tänker jag att vi alltid kommer att se till. Jag är väl den enda som inte är anonym.

[Magnus]
Det är en annan grej. Det som inte heller finns i datan som jag har problem med. För jag kan inte se hur många som ligger i listan när en person gör någonting.

Det finns ju den datan någonstans, men jag har inte fått något enkelt sätt att få fram den. Men det är ju den som laggar sig i hur många som ligger här nu. Den är ju ganska central.

Tog du lång tid på det? Den har jag haft jättesvårt att få fram egentligen. Jo, många jobbar olika hastighet beroende på hur mycket resan ligger och väntar.

Den kan också vara lite svår. Datan själv säger ju inte till mig om någon jobbade snabbare den dagen. Man kan ju få ut hur lång tid det tog från A till B, men hur lång tid borde det ta?

[Sara]
Ja, för det är ju då och då att de med operationskörningar så är de inte helt redo. Men man vet att de bör prioritera, så då väntar jag ändå i tio minuter nästan.

[Magnus]
Det är nog bara en av tio som fyller i den avvikelsen och den ligger i datan.

[Sara]
Kan man fylla i det? Ja, precis. Visste jag inte ens.

[Magnus]
Här är en liten gammal lapp som hänger på den där.

[Sara]
Okej.

[Magnus]
När du har klart uppdraget så finns det en liten val längst upp där du kan trycka på så får du en hisskronge eller en patient som inte är klar.

[Sara]
Det ska jag börja fylla i då.

[Magnus]
Ja, gör det gärna.
[Sara]

Det är ju också en sån bra grej att man inte vet alla de funktionerna kanske. Att om det effektiviseras lite mer eller att om det blir en sån här ändring så kommer man ju gå igenom det. Då får ju alla en ny upplärning med systemet.

Sen kanske inte det löser de problemen helt ändå, men då hinner man ju kolla igenom det en gång. Får vi namnge dig i rapporten som källare till information?

[Magnus]

Ja. Det går att hitta mig ändå på nätet.

[Sara]

Ja, det är det. Annars hade vi ju skrivit chef på patientprofessoriet. Men vi kommer ju inte citera och hålla på utan mer referera till den här intervjun och om jag frågar något på mail och lite så.

Att vi har fått information där. Och sen appen får jag ta kort på den och visa upp på mina gruppmedlemmar för att förklara hur den fungerar. Om jag ser till att inte ha några namn så att jag kan skapa ett eget jobb.

Och fotar.

[Magnus]

Den är nog lite svårare. Inte jag kan dela några sådana instruktioner från de offentliga handlingarna.

[Sara]
Men det hade ju varit bra. För det är ju främst det att kunna visa hur den fungerar. [Magnus]
Den ägs ju av en annan partner nu så jag vågar nog inte.

[Sara]
Nej.

[Magnus]
Det är jag som hittar den här.

[Sara]

Ja, det vet du väl.

[Magnus]

Det där är för beställarna. Det här är den riktiga instruktionen.

[Sara]

För de bilderna. Det kanske vi behöver kolla med dem i sådana fall. Att vi kanske skulle kunna använda de bilderna i vår rapport.

DEPARTMENT OF SOME SUBJECT OR TECHNOLOGY
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

CHALMERS