Delft University of Technology

MSc Transport, Infrastructure & Logistics Thesis

Comparing Distribution Strategies to Achieve Peak Shaving at Occupancy Levels of the Baggage Handling System of an Airport

Author:
Bruno Nijdam

Supervisors Technical University of Delft:
Dr. Alessia Napoleone
Dr. Jaap Vleugel

Supervisor Amsterdam Airport Schiphol:
Christopher Roos

Reportnumber: 2024.TIL.9011
Studentnumber: 4486846

December 10, 2024

Preface

In 2015, I began my journey at TU Delft with a Bachelor's degree in Civil Engineering. While I enjoyed the program, I gradually realized that my interests were shifting toward problem-solving rather than the more straightforward calculations involved in structural engineering. This realization led me to pursue a Master's in Transport, Infrastructure Logistics, a program that aligned more closely with my passion for tackling complex logistical puzzles.

When it came time to look for a graduation internship in late 2023, I already knew where I wanted to focus: the aviation industry. As a child, I had dreamt of becoming a pilot—a dream that hasn't (yet) come true, but my passion for aviation has remained steadfast. Through a friend, Aliex van Wingerden, I connected with Micha Dijkhuizen and Christopher Roos. After insightful conversations with both, Christopher invited me to undertake my graduation research on baggage optimization within the Innovation Hub at Royal Schiphol Group. I am grateful for this incredible opportunity to combine my research with an internship at Schiphol. The absolute highlights of my internship have undoubtedly been the visits to the airside.

I would like to thank several people in particular for their guidance throughout my graduation research. From TU Delft, I owe special thanks to Alessia Napoleone, who supported me through all phases of my research, helping me overcome challenges along the way. I truly appreciated our collaboration and felt comfortable discussing any aspect of my graduation journey with you. My thanks also go to Jaap Vleugel, who joined my supervision team later but always posed the right questions and offered well-directed feedback that energized me to push forward. Additionally, I want to thank Stefano Fazi for his valuable support in setting up my research.

At Royal Schiphol Group, I am grateful to Christopher Roos for the opportunity to complete this internship at the Innovation Hub. Chris was always available to answer my questions or brainstorm about the research, which was incredibly valuable. I would also like to thank Micha Dijkhuizen for his feedback and insightful discussions. I am thankful for the brainstorming sessions I had with Roman Kazus, Cardo van der Lee, Huy Vu, Jeroen Kamp, and Simon van der Weij, as well as the rest of the Innovation Hub team for their support and the positive work environment.

I would also like to extend my gratitude to Bram Hoeksma, Daniela Barge, Frederic Barge, and Joost Witteman for their feedback and support in approaching this research, and to Lucas Bouma for his assistance with the coding aspects of the project.

B.R.M. Nijdam
Amsterdam, November 2024

Summary

This thesis investigates how buffering and strategic distribution of cold transfer baggage can reduce peak occupancy levels at transfer infeed points within the Baggage Handling System (BHS) at Amsterdam Airport Schiphol (AAS). At hub airports like AAS, fluctuating inflows of transfer baggage create peaks and troughs in system occupancy, leading to operational challenges, resource imbalances, and potential delays. The research focuses on cold transfer baggage due to its longer layover times, which allows for controlled buffering and delayed processing, offering an opportunity for peak shaving.

The study begins by analyzing the operational and logistical background of baggage handling, emphasizing the need for an optimized BHS to handle the dynamic demands of a major transit hub. A literature review examines existing optimization techniques in baggage handling, with a specific focus on dynamic routing, robust scheduling, and queuing models. While previous studies have explored general peak shaving techniques, this research addresses a notable gap in the literature by developing distribution strategies specifically tailored to the buffering and reintroduction of cold transfer baggage.

To explore these strategies, a simulation model was developed to mimic the transfer baggage journey from aircraft arrival to transfer infeed points, integrating a time series forecasting model to determine optimal reintroduction times. Six unique distribution strategies emerged from combining three buffering approaches—Fixed Target, Polynomial Target Function, and a Combination approach—with two reintroduction timings, Early Release and Optimized Release. Each strategy was tested across scenarios that modeled different operational conditions.

The research found that strategic buffering and reintroduction of cold transfer baggage could reduce peak occupancy by up to 26.8%. The Polynomial Target Function with Optimized Release achieved the highest level of peak shaving but required a larger number of Automated Guided Vehicles (AGVs) to handle the transport of the baggage, illustrating a trade-off between peak shaving effectiveness and resource demands. Conversely, the Fixed Target approach, while slightly less effective in peak reduction (25.7%), offered a resource-efficient alternative by needing considerably less AGVs. The Early Release option, though yielding lower peak shaving results, proved valuable in maintaining buffer capacity, making it advantageous when buffer capacity must be minimized.

Implementing these strategies presents operational considerations, particularly in terms of AGV availability and buffer management. The findings underscore the importance of predictive modeling and flexible buffering to dynamically manage occupancy levels, which can help AAS to stabilize BHS operations, optimize resource use, and improve system resilience under varying demand conditions. This research contributes actionable insights into how airports can utilize data-driven, adaptive strategies for peak shaving, helping improve operational efficiency and passenger satisfaction in complex airport environments.

Nomenclature

Abbreviation	Definition
AAS	Amsterdam Airport Schiphol
AGV	Autonomous Guided Vehicle
AIBT	Actual In Block Time
BHS	Baggage Handling System
BSM	Baggage Source Message
CoV	Coefficient of Variation
GRASP	Greedy Randomized Adaptive Search Procedure
IATA	International Air Transport Association
ICAO	International Civil Aviation Organisation
KLM	Koninklijke Luchtvaart Maatschappij
KPI	Key Performance Indicator
MAC	Mean Absolute Change
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
NABO	Narrow Body
RMSE	Root Mean Squared Error
RO	Robust Optimization
ShoCon	Short Connection
SIBT	Scheduled In Block Time
STD	Standard Deviation
SVAP	Stochastic Vector Assignment Problem
ULD	Unit Loading Device
VOP	Vliegtuig Opstel Plaats
WIBO	Wide Body

List of Figures

List of Tables

Contents

1 Introduction

In this chapter, the background information for this research is provided. Then, the problem statement is explained. Next, the research objectives and research questions are examined. Finally, the approach of this research is discussed.

1.1 Research Background

The Baggage Handling System (BHS) at airports is essential for efficiently managing the movement of passengers' baggage. It ensures a smooth transition of bags from check-in counters to aircraft for departing flights, and from incoming flights to baggage claims or connecting flights. The BHS relies on a complex network of automated sorting systems, conveyor belts, and advanced barcode scanners to ensure that baggage is sorted, routed, and delivered quickly and correctly. Optimizing the BHS is crucial for several reasons. Firstly, it enhances operational efficiency, ensuring that baggage moves seamlessly through the system with minimal errors and delays. This optimization can significantly reduce operational costs by minimizing the need for manual interventions and reducing the risk of lost or mishandled bags. Secondly, an optimized BHS improves passenger satisfaction by ensuring that their baggage arrives on time and in the correct location, thus enhancing the overall travel experience. Furthermore, efficient baggage handling can lead to faster turnaround times for aircraft, enabling more flights to operate on schedule and increasing the airport's capacity to handle more passengers. Ultimately, the optimization of the BHS is vital for maintaining a competitive edge in the aviation industry, where efficiency and customer satisfaction are paramount.

Each airport is divided into landside and airside. The landside includes areas before security, where passengers check in. The airside is beyond the customs area and includes the planes, runways, taxiways, and gates.
Baggage handling at a hub airport involves several tasks and is divided into three stages (Barth et al., 2021). The first stage is check-in baggage. Check-in baggage is fed into the BHS at the landside. The second stage is transfer baggage, which involves moving bags from one flight to another for transit passengers. The third stage is reclaim baggage. Reclaim baggage has arrived at the airport and is transported from the airside to the landside, where arriving passengers can collect their baggage. Figure 1.1 shows a visualisation of the baggage journey.

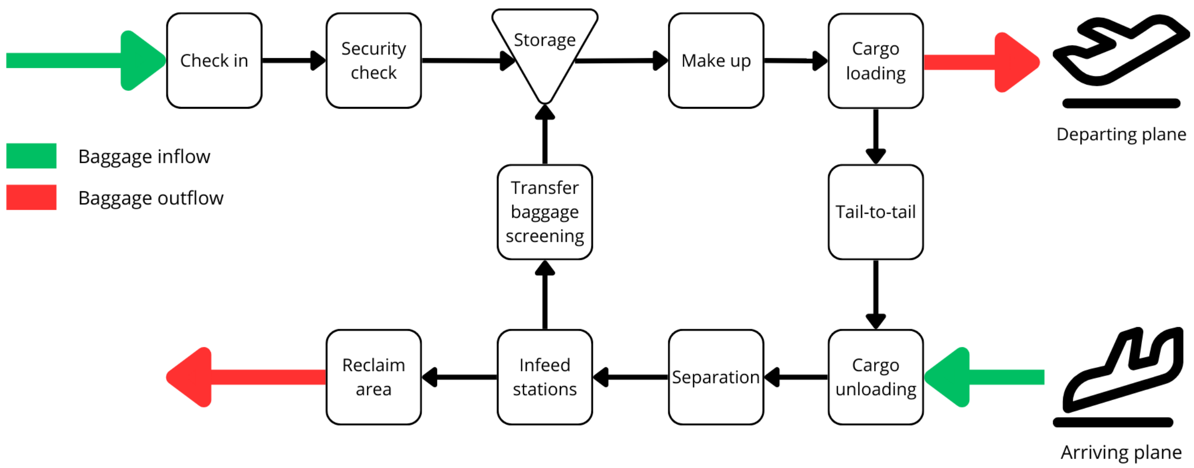


Figure 1.1: Visualization of the baggage journey

When zooming in on the transfer baggage, four streams are distinguished. First is the transfer baggage that needs to depart shortly after arrival and does not have time to be processed through the BHS, classified as tail-to-tail baggage. This baggage is transported directly to the apron of the outbound aircraft without going through the BHS. The second stream is short connection baggage (ShoCon). This baggage has too short a layover time to be buffered but enough time to be processed through the BHS. ShoCon baggage is brought to one of the transfer infeed points upon the arrival of the airplane. Transfer infeed points are locations where transfer baggage can be fed into the BHS. Besides transfer infeed points, there are also reclaim infeed points for the reclaim baggage. The third transfer baggage stream is cold transfer baggage. Cold transfer baggage has longer layover times than ShoCon and does not need to be processed immediately by the BHS. It is stored in the storage area of the BHS. The fourth stream is the super cold transfer baggage category. Although it functions similarly to cold transfer baggage, super cold transfer baggage is considered a separate category due to its classification as a different type of "cold." Nonetheless, its operation is the same as cold transfer baggage: it can be stored before it is handled.

Airports continuously seek improvements in their operational processes, including the BHS. At hub airports, the concentration of many flights in the morning, which then depart throughout the day, offers numerous transit opportunities for passengers. However, this leads to fluctuating volumes of transfer baggage, causing operational challenges.

1.2 Problem Description

Airports are complex systems where the design of infrastructure and processes plays a critical role in determining operational efficiency. However, once the design is in place, the primary challenge lies in operating the system effectively under varying conditions. This is particularly true for the BHS at hub airports, where fluctuating baggage flows result in significant operational challenges.

Over the years (excluding the COVID years), there has been an upward trend in the demand for travel and an increase in flight movements (International Air Transport Association (IATA), 2024; International Civil Aviation Organization (ICAO), 2022). Hubs are characterized by a high influx of aircraft at certain intervals a couple of times during the day, followed by a more continuous flow of departing aircraft. This uneven distribution leads to peaks and troughs in baggage volumes, significantly impacting the BHS and its subsystems (Hanwu & Juan, 2009). While the infrastructure is designed to accommodate average demand levels, the variability in baggage flow tests the system's ability to operate efficiently within its design constraints. The fluctuations result in peaks and troughs in the occupancy levels of subsystems of the BHS that can be seen in Figure 1.2 where the averaged occupancy levels of all transfer infeed points over 10 weeks are being shown over 24 hours at Amsterdam Airport Schiphol (AAS).

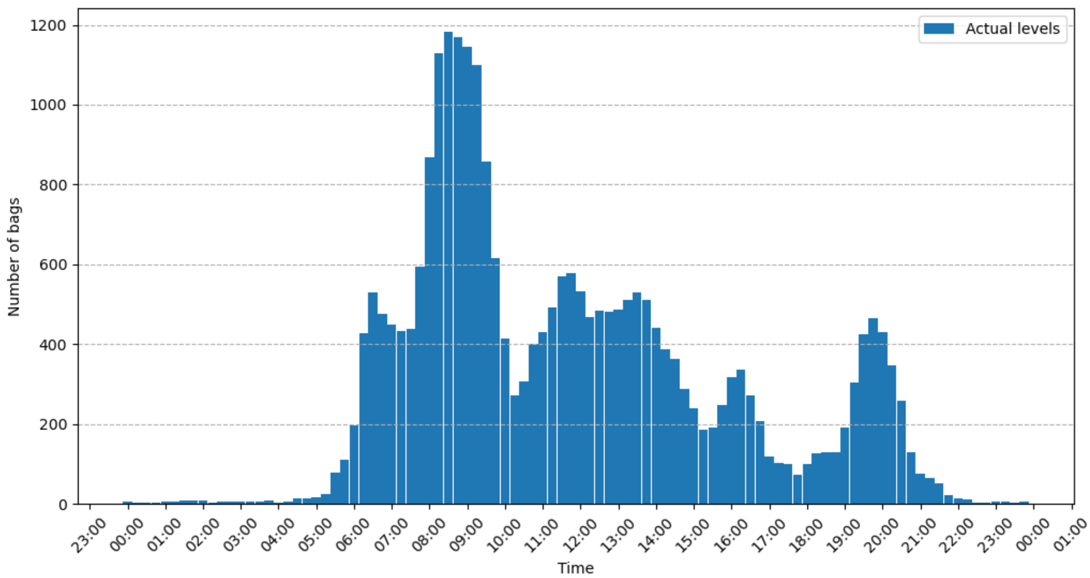


Figure 1.2: Averaged occupancy levels of transfer infeed points per 15 minutes over a dataset of 10 weeks at AAS

These fluctuations result in operational challenges. One of the problems is the staffing imbalance at the baggage handling stations, where handlers are frequently understaffed during peak periods and overstaffed during off-peak times. The need for additional personnel during these peak periods poses a logistical and financial challenge, as the operational model of airports does not lend itself easily to the flexible scheduling of work hours (Littler & Whitaker, 1997). Employing additional staff exclusively for peak times is impractical and financially untenable.

Additionally, the variable flow of transfer baggage significantly disrupts other components of the BHS. For instance, a surge in transfer baggage during peaks not only strains the sorting systems, which operate at fixed capacities, but also burdens the internal buffers designated for temporarily storing early-checked baggage. These fluctuations of transfer baggage manifest downstream in the system, leading to bottlenecks and inefficiencies that can affect the entire operation, from sorting to final baggage claim areas, reflecting the interconnected nature of airport operations. Some airports, besides these issues, also deal with limited expansion possibilities due to a lack of space. At some airports there is little room to expand the current BHS, so smart innovations must help in optimizing the BHS without taking up too much space.

1.2.1 Proposed Solutions

To mitigate the operational challenges caused by fluctuating baggage volumes, two potential solutions can be explored:

- **Check-in baggage earlier**: One way to reduce the peak stress on the BHS is by processing check-in baggage earlier than usual. This can be achieved by encouraging passengers to check in their bags earlier in the day or even the day before their flight. By processing check-in baggage well before peak transfer periods, the load on the BHS during these critical times can be reduced. Essentially, the peak created by check-in baggage is "shifted forward," allowing more capacity to handle the surge in transfer baggage later in the day. This strategy would rely on efficient scheduling and coordination, but it can greatly alleviate the pressure on sorting systems during peak times.
- **Buffering cold transfer baggage**: The second approach focuses on cold transfer baggage—baggage with longer layover times. This baggage can be buffered for an extended period, meaning it does not need to enter the BHS immediately upon arrival. By strategically delaying its processing and feeding it into the system during off-peak times, the load during peak periods is effectively reduced. However, it is crucial that this baggage still meets its outbound flight schedule, so the delay in processing is carefully timed to avoid any impact on departure times. This approach, known as peak shaving, allows for a more balanced distribution of baggage handling workloads over time, thereby improving overall system efficiency.

In Figure 1.3, the impact of these two solutions is illustrated. The figure shows how advancing check-in baggage processing shifts that workload earlier in the day, while buffering transfer baggage moves the processing load later, smoothing the peak and reducing operational pressure during the critical hours.

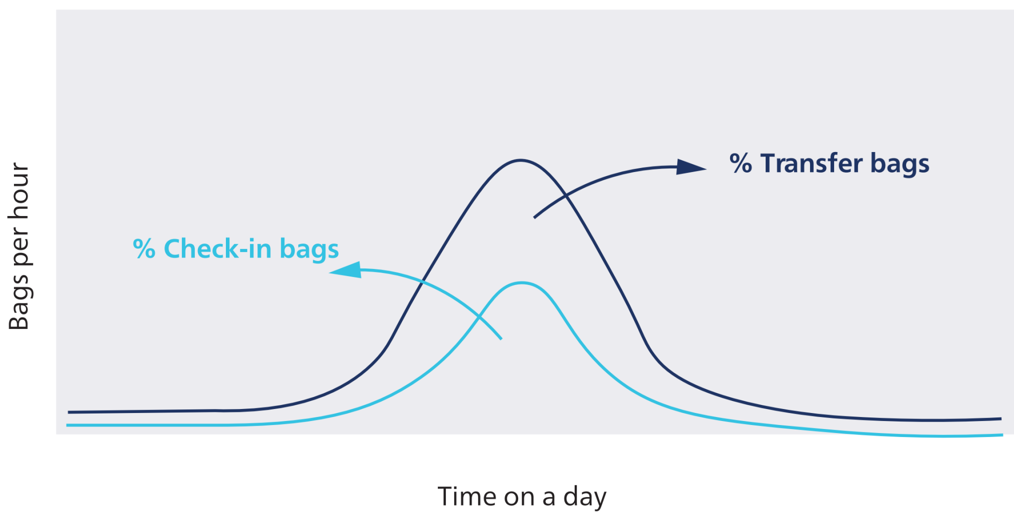


Figure 1.3: Peak shaving can be realized by processing CI baggage before peak hours and TRF baggage after peak hours.

1.2.2 Peak Shaving by Buffering Cold Transfer Baggage

Cold transfer baggage, which has longer layover times, offers an opportunity to reduce peak load on the BHS by delaying its processing. By placing this baggage in external buffers and reintroducing it into the system during off-peak hours, the workload on the BHS can be smoothed out throughout the day. AAS has already demonstrated that basic buffering strategies can achieve peak shaving, but further refinement is needed to optimize this approach.

Several aspects are critical when designing an effective cold buffering system:

- **Number of Buffers**: Research indicates that using multiple decentralized buffers is more efficient than a single centralized one, reducing transportation times and streamlining re-entry into the BHS at AAS (van der Grift, 2023).
- **Buffer Organization**: The layout of the buffer also plays a role in system efficiency. For example, baggage can be organized by departure time or destination to ensure it is quickly retrieved when needed. A well-structured buffer reduces handling times and minimizes disruptions when reintroducing baggage into the system.
- **Distribution Strategies**: This research specifically focuses on how to distribute the buffered baggage most effectively. The key challenge is determining the optimal timing and strategy for releasing baggage from the buffer. By dynamically adjusting the reintroduction of baggage based on occupancy levels at the infeed points, peak shaving can be maximized. This requires a sophisticated scheduling system that takes into account the varying loads on the BHS throughout the day.

In summary, while some aspects of cold transfer baggage buffering, such as the number of buffers and their internal organization, have been explored, this research focuses on developing strategies for efficient distribution of buffered baggage. By improving the timing and method of reintroducing cold baggage, airports can significantly enhance the efficiency of the BHS, reduce bottlenecks, and improve overall operational performance.

1.3 Literature

BHS are critical to airport operations, ensuring the efficient transfer of baggage from check-in to the aircraft, aircraft to aircraft, and from the aircraft to reclaim area. The increasing complexity of airport operations, rising passenger volumes, and heightened security requirements necessitate continuous optimization of these systems to enhance efficiency, reliability, and passenger satisfaction. Also, specific attention has been paid to studies that are relevant to optimization in the BHS of airports. Various studies have proposed methods and models to optimize BHS at airports. An overview of these studies is given in Table 1.1.

Author	Title	Year
Clausen, Kissinger	Dynamic Routing of Short Transfer Baggage	2010
Barth et al.	A Model for Transfer Baggage Handling at Airports	2013
Frey	Models and Methods for Optimizing Baggage Handling at Airports	2014
Huang et al.	Optimal Assignment of Airport Baggage Unloading Zones to Outgoing Flights	2016
Frey et al.	Optimizing Inbound Baggage Handling at Airports	2017
Huang, Liu & Lin	Robust Model for the Assignment of Outgoing Flights on Airport Baggage Unloading Areas	2018
Ekeocha & Ushe	Application of Queuing Process in the Optimization of Baggage Handling System of Murtala Muhammed International Airport	2018
Barth et al.	Optimization of Transfer Baggage Handling in a Major Transit Airport	2021

Table 1.1: Overview of Studies on Baggage Handling Systems Optimization

The research by Clausen and Pisinger (2010) on "Dynamic Routing of Short Transfer Baggage" addresses the need for more efficient baggage handling processes at airports. The study highlights the challenges associated with routing short transfer baggage, which requires timely and accurate delivery to ensure smooth operations and passenger satisfaction. The authors propose a dynamic routing algorithm that adapts in real-time to the varying conditions within the BHS. By leveraging real-time data, the proposed algorithm optimizes routing decisions, thereby reducing delays and improving the reliability of baggage transfers. This research is particularly relevant for large, busy airports where the volume of transfer baggage is high and the need for efficient routing solutions is critical.

Frey (2014)'s work presents significant advancements in the planning and scheduling of inbound baggage handling systems. Recognizing the complexity and critical nature of inbound baggage handling, Frey (2014) proposes a comprehensive optimization approach using hybrid heuristics. This hybrid heuristic combines a Greedy Randomized Adaptive Search Procedure (GRASP) with a guided fast local search and path-relinking. Implemented at Munich's Franz Josef Strauss Airport, the proposed algorithm demonstrated impressive results, reducing baggage peaks at carousels by 38% and passenger waiting times by 11%.
Building on this foundation, Frey et al. (2017) further elaborate on these methods by focusing on the planning and scheduling of inbound baggage with the aim of optimizing the entire process through the use of extensive simulations and real-world data. The research underscores the importance of combining heuristic techniques with mathematical modeling to tackle complex optimization challenges in airports. This structured approach ensures that the solutions are both theoretically sound and practically viable.

Barth and Larsen (2013) introduce a dynamic model using fuzzy logic to optimize transfer baggage handling in their paper "A Model for Transfer Baggage Handling at Airports". This model simulates human decision-making in baggage operations, enhancing adaptability to real-time changes. Validated with a case study at a major international airport, the model shows significant improvements in resource allocation, reduction of baggage mishandling, and overall system efficiency. By integrating various subsystems like passenger arrival, check-in, and sorting, the model offers a comprehensive approach to identifying and resolving bottlenecks. The use of fuzzy logic enables the model to handle uncertainties, such as flight delays and fluctuating passenger volumes, making it robust and reliable.
A follow-up study "Optimization of Transfer Baggage Handling in a Major Transit Airport" by Barth et al. (2021) present a model for scheduling and assigning outbound baggage handling facilities to flights. This study focuses on improving the efficiency and robustness of baggage handling processes, particularly in major transit hubs where high volumes of transfer baggage pose significant challenges. The authors propose a dynamic, near real-time scheduling model that considers multiple criteria such as quality, efficiency, and robustness. By incorporating these factors into a multi-criteria objective function, the model optimizes the assignment of handling facilities and determines the appropriate start times for baggage handling tasks. Using real-world data from Frankfurt Airport, the study validates the model, demonstrating that the proposed heuristic and decomposition methods significantly enhance the efficiency and reliability of baggage handling operations.

Huang et al. (2016) address the complexities of assigning unloading zones to outgoing flights in their paper "Optimal Assignment of Airport Baggage Unloading Zones to Outgoing Flights". The study highlights the inherent uncertainties in airport operations, such as flight delays and varying numbers of bags, which necessitate robust optimization approaches. The authors model the chute assignment problem as a Stochastic Vector Assignment Problem (SVAP), incorporating multiple extensions to address various design needs of airports. This model allows for a more flexible and dynamic assignment process, adapting to real-time changes and uncertainties. Results show that the proposed models significantly improve the efficiency and reliability of the baggage handling system compared to traditional methods. The robust optimization model reduces the total number of manually handled baggage by 58%, illustrating its effectiveness in managing the complexities and uncertainties of real-world airport operations.
Huang et al. (2018) build upon their earlier research in "Robust Model for the Assignment of Outgoing Flights on Airport Baggage Unloading Areas". This paper focuses on developing a robust optimization model to improve the efficiency and reliability of BHS at airports, addressing the inherent uncertainties in airport operations such as flight delays and fluctuating baggage volumes. The robust optimization (RO) model proposed by the authors aims to maintain performance stability under various uncertainty conditions. This model creates a robust assignment plan for outgoing flights to baggage unloading areas, ensuring consistent performance even when unexpected disruptions occur. Constructing a simulation model of the BHS, the study compares the performance of the robust optimization model against traditional assignment methods. Results demonstrate that the robust model significantly outperforms conventional methods, reducing the total number of manually handled baggage by 58%. This reduction not only improves efficiency but also enhances the reliability of the baggage handling process, leading to fewer delays and increased passenger satisfaction.

The paper 'Application of Queuing Process in the Optimization of Baggage Handling System of Murtala Muhammed International Airport' by Ekeocha and Ushe (2018) reviews various BHS models and applies a queuing process to optimize the system, especially in managing the flow and imbalances during peak hours at Nigeria's largest airport. The suggested algorithms aim to enhance efficiency and reduce waiting times for passengers.

The reviewed studies highlight several methodologies and models for optimizing BHS at airports, each building on previous research to address the challenges of modern airport operations. Clausen and Pisinger (2010) focus on dynamic routing algorithms that adapt to real-time conditions, while Frey et al. (2017) introduces hybrid heuristics for efficient inbound baggage handling and further elaborates on their implementation and success in real-world scenarios. Barth et al. (2021) utilize fuzzy logic for adaptability in transfer baggage handling and propose dynamic scheduling models for major transit airports. Huang et al. (2018) emphasize robust optimization models to handle uncertainties in baggage unloading zones and outgoing flights assignments.

Recent advancements in the field, such as condition-based maintenance, comprehensive simulation models, and sustainable optimization techniques, further contribute to the ongoing improvement of BHS. These varied approaches underscore the importance of flexibility, adaptability, and robustness in optimizing baggage handling systems. By integrating advanced algorithms, real-time data, and robust optimization techniques, these studies enhance the efficiency, reliability, and performance of airport BHS, ultimately improving passenger satisfaction and operational effectiveness.

1.3.1 Conclusion

In the literature, multiple studies can be found where peak shaving occurs in various sectors. This is a hot topic because optimization leads to lower costs and more efficient systems. It is seen that optimization modeling is commonly used for comparable problems. Optimization modeling is the process of finding the best solution from a set of feasible solutions, typically by maximizing or minimizing an objective function subject to various constraints. It involves mathematical techniques such as linear programming, integer programming, and heuristic methods to solve complex decision-making problems efficiently.

1.4 Research Gap

While various studies address optimization in airport BHS, including improvements in routing, scheduling, and resource allocation, there is a noticeable gap in the literature concerning the role of distribution strategies in achieving peak shaving at transfer infeed points. Existing research predominantly focuses on static or heuristic approaches to managing baggage flow, often without considering the dynamic and variable nature of transfer baggage inflow, which is subject to fluctuating peaks and troughs throughout the day.

To date, limited attention has been given to the potential of buffering strategies that are specifically tailored for cold transfer baggage—baggage with sufficient layover time that can be temporarily stored without jeopardizing its outbound schedule. Although previous studies have explored general peak shaving techniques in other systems, their application to the unique constraints of transfer baggage at transfer infeed points remains underexplored. The impact of targeted buffering, combined with forecasting and dynamic reintroduction based on real-time occupancy levels, is an area where few studies have provided empirical or simulation-based insights.

This thesis addresses this knowledge gap by developing a simulation model that mirrors the transfer baggage journey and incorporates a predictive forecasting model to determine optimal moments for releasing buffered baggage. By examining various distribution strategies in controlled scenarios, this research aims to understand the conditions under which different strategies can most effectively mitigate peaks at transfer infeed points. Additionally, it seeks to establish the viability and robustness of these strategies under varying operational conditions, contributing a nuanced understanding of how buffering can be leveraged to enhance overall BHS efficiency and reliability.

Ultimately, this study provides new insights into distribution strategies for buffered baggage, with the goal of achieving peak shaving at the transfer infeed points. By addressing the current lack of research on these strategies, this thesis advances the field of airport baggage handling and supports future research on adaptable, data-driven optimization solutions within BHS operations.

1.5 Research Objectives

The primary objective of this research is to explore and evaluate distribution strategies for buffering cold transfer baggage to achieve peak shaving at transfer infeed points within the BHS at AAS. Specifically, this research aims to develop a simulation model that incorporates a predictive forecasting system to dynamically manage the timing of when to buffer and release baggage, helping to mitigate peak occupancy levels. The research objectives are as follows:

- **Objective 1:** Develop a comprehensive simulation model of the transfer baggage journey, from arrival to the transfer infeed point, capturing the dynamic nature of baggage flows and the impact of fluctuating occupancy levels on the BHS.
- **Objective 2:** Design and implement a predictive forecasting model to determine optimal release times for buffered baggage based on real-time occupancy data at transfer infeed points, supporting dynamic peak shaving efforts.
- **Objective 3:** Develop and evaluate multiple distribution strategies for cold transfer baggage, with a focus on optimizing timing and allocation methods to enhance peak shaving and balance baggage flows at transfer infeed points. These strategies will be informed by an in-depth review of relevant literature, analysis of the system's operational environment, and insights gathered from interviews with experts at AAS.
- **Objective 4:** Investigate and compare the designed distribution strategies, analyzing their effectiveness in reducing peak occupancy and robustness under different operational scenarios.
- **Objective 5:** Provide actionable insights and recommendations on how airports can use distribution strategies to enhance operational efficiency, manage peak periods, and optimize resource allocation.

These objectives aim to address the current gap in research on peak shaving through distribution strategies for cold transfer baggage, contributing to a deeper understanding of how targeted buffering can improve airport baggage handling efficiency.

1.6 Research Contribution

This thesis provides meaningful contributions to both academic research and the practical operations at AAS. From a scientific perspective, it advances the understanding of dynamic buffering and distribution strategies for cold transfer baggage, focusing on their role in achieving peak shaving at transfer infeed points. Unlike prior studies that primarily rely on static or heuristic methods, this research integrates simulation modeling with predictive time series forecasting to explore how baggage flows can be dynamically managed under fluctuating conditions. By systematically comparing multiple distribution strategies and integrating simulation with forecasting models, this research introduces a flexible framework for dynamic optimization. The approach demonstrated in this thesis provides a foundation for future studies and highlights the potential of adaptive methods in managing complex operational systems within logistics and existing aviation infrastructure.

From a practical standpoint, this research is directly relevant to AAS. The simulation model developed in this thesis is tailored to the specific characteristics of the AAS baggage handling environment, enabling the airport to evaluate and optimize distribution strategies for cold transfer baggage under realistic conditions. By utilizing AAS's operational environment as a basis for simulation, the findings provide actionable insights that can be implemented directly to alleviate peak occupancy levels and improve system efficiency. Additionally, the research includes calculations and evaluations of transport capacity, such as the coordination of Automated Guided Vehicles (AGVs), ensuring that available resources are effectively allocated. These contributions align closely with AAS's strategic objectives of enhancing operational efficiency, improving passenger satisfaction, and minimizing unnecessary expansions or costs.

By bridging the gap between theoretical research and practical application, this thesis demonstrates how dynamic buffering strategies, guided by a forecasting model, can optimize resource use and reduce operational inefficiencies. Furthermore, it lays the groundwork for future studies by illustrating the potential of adaptive, data-driven approaches in addressing variability in complex systems.

1.7 Research Scope

The scope of this research focuses on the development, testing, and evaluation of distribution strategies for buffering super cold transfer baggage within the BHS at AAS. This study is confined to the following areas:

- **Simulation Model:** The simulation model developed in this research is designed to capture the flow of transfer baggage from arrival at AAS to the transfer infeed points, where baggage is fed to the BHS. The model includes variations in baggage inflow and dynamically manages occupancy rates through strategic buffering.
- **Super Cold Transfer Baggage:** This research focuses specifically on super cold transfer baggage, which have layover times long enough to allow temporary storage. Other types of baggage, such as tail-to-tail and ShoCon baggage, are excluded from the buffering strategies because their layover times are insufficient for effective buffering. Although cold transfer baggage could also be buffered, this study limits its scope to super cold transfer baggage. This decision was informed by discussions with various stakeholders, who highlighted that buffering cold transfer baggage involves tighter safety margins. As a result, super cold transfer baggage was chosen as the focus to ensure operational feasibility and reliability.
- **Distribution Strategies:** This research involves designing and evaluating distribution strategies for releasing buffered baggage. The strategies will be based on timing and occupancy level predictions to achieve peak shaving at the transfer infeed points. These strategies are intended to help balance the workload across the BHS by redistributing baggage flows during high-demand periods.
- **Operational Scenarios:** To test the robustness of the distribution strategies, this study examines different operational scenarios, including variations in baggage volumes, peak occupancy times, and potential system disruptions. This allows for an assessment of each strategy's effectiveness and adaptability under varying conditions.
- **Exclusions:** This research does not extensively address the operational logistics of baggage transportation outside of the BHS. A brief analysis is conducted on the movement of baggage on the airside, but no in-depth research is performed. Furthermore, this study does not consider financial constraints or cost analyses related to implementing the proposed strategies.

In summary, the scope of this research is narrowly focused on exploring and optimizing the buffering and distribution of super cold transfer baggage to alleviate peak occupancy levels at transfer infeed points at AAS. The study's findings aim to provide insights into how strategic buffering can contribute to smoother BHS operations and enhanced resource allocation at large hub airports.

1.8 Research Question

This research aims to explore how the distribution of buffered super cold transfer baggage can be optimized to achieve peak shaving, thereby improving the overall efficiency of the BHS. To guide this study, the following main research question is formulated:

How to support Amsterdam Airport Schiphol to identify the best distribution strategy for super cold transfer baggage through a simulation model?

To address this overarching question, the following sub-questions will be explored:

1. **What is the current state of baggage handling processes at the transfer infeed points at AAS?**
   This question is essential to understand the baseline of current operations. By analyzing the existing baggage handling processes, challenges, and bottlenecks, this research can identify the specific areas where peak shaving through buffering could have the most impact. Understanding the current state also helps clarify the potential benefits and limitations of introducing new distribution strategies.
2. **How does peak shaving work in the context of an airport?**
   Peak shaving is a key strategy to manage high baggage volumes, particularly during peak times. This question explores the theory behind peak shaving, how it applies specifically to airport baggage systems, and why it is relevant to managing transfer baggage. Addressing this question provides a foundation for understanding the value of buffering and its role in reducing congestion at the transfer infeed points.
3. **What is needed to optimize the distribution of super cold transfer baggage?**
   This question looks at what is needed to successfully set up an effective distribution strategy. Specifically, it focuses on how forecasting can help find the best timing for releasing buffered containers. By accurately predicting changes in baggage volumes, the forecasting model can guide decisions on when to return baggage from the buffer, helping to keep peak occupancy low without disrupting airport operations.

4. **Which distribution strategies are suitable for buffering super cold transfer baggage, and how do these strategies impact peak shaving under different scenarios?**
   This question investigates the types of distribution strategies that can be applied to super cold transfer baggage and their potential effects on peak shaving. By exploring different strategies and scenarios, such as varying inflow rates or capacity limitations, this research can evaluate how each approach performs under different operational conditions, providing insight into the robustness and adaptability of each strategy.

5. **What operational considerations and resource implications arise from implementing the simulated distribution strategies, and how feasible are they for reducing peak occupancy at AAS transfer infeed points?**
   This question explores the practicality of implementing each distribution strategy, focusing on resource requirements, operational adjustments, and overall feasibility. By assessing these factors, the research aims to determine the viability of integrating these strategies into AAS's baggage handling operations.

1.9 Approach

In this thesis, the research methodology is presented in greater detail in the subsequent chapter 2. It begins with an overview of the baggage journey at AAS, provided in chapter 3. This offers a foundational understanding of the system under study. Next, the concept of peak shaving and its significance for managing the BHS is explained in chapter 4. The data used for the study is described in chapter 5, alongside any necessary data transformations. Following this, the process of developing the simulation model is elaborated in chapter 6, where the simulation logic is detailed and the validation process is explained. Chapter 7 introduces the time series forecasting model, which is built upon the simulation output and validated using historical trends. In chapter 8, the various distribution strategies for buffered super cold transfer baggage are designed and explained, followed by the scenarios. The performance of these strategies under different operational scenarios is shown in chapter 9, and discussed in chapter 10, where results are analyzed, and the effectiveness of each strategy is assessed. Then the conclusions of the research are presented in chapter 11. Finally, the recommendations are given in chapter 12.

2 Methodology

This chapter outlines the methods used to explore how the distribution of buffered super cold transfer baggage can be optimized to achieve peak shaving at transfer infeed points. The research focuses on developing a simulation model based on an analysis of the current state of the BHS at AAS and using 10 weeks of historical baggage data. By designing and testing multiple distribution strategies and evaluating their effectiveness across different scenarios, this research aims to identify the most efficient approach to managing baggage flows.

2.1 Current State Analysis

To establish a solid foundation for the simulation model, a detailed current state analysis of the BHS at AAS was conducted. This analysis provided critical insights into the existing operations and challenges within the system, offering a baseline for designing the simulation.

The current state analysis involved the following steps:

- **Field Observations:** Firsthand observations of the BHS systems were made by visiting the airside of AAS. This enabled a close-up view of the baggage handling processes, including infeed, storage, sorting, and make-up. Observing the handling of both bulk and containerized baggage helped capture the complexities of the system.
- **Stakeholder Discussions:** Meetings were held with baggage handling experts at AAS and with the KLM Ground Handlers, including those involved in airside operations and the design and management of the BHS. These discussions provided valuable context, including operational constraints, the rationale behind current processes, and the potential for optimization.
- **Process Mapping:** Using the information gathered, a detailed process map of the baggage journey was created to identify key touchpoints and bottlenecks in the system. This map served as a visual representation of how baggage moves through the system and the interactions between subsystems.
- **Key Performance Indicator Review:** Existing KPIs, such as system capacity, on-time delivery rate, and buffer efficiency, were reviewed to understand current performance levels and areas requiring improvement.

The current state analysis was instrumental in shaping the simulation model by providing a clear understanding of the processes and constraints that needed to be incorporated. Additionally, insights gained from field visits and discussions helped identify practical considerations for implementing distribution strategies.

2.2 Peak Shaving: Concept and Relevance

Peak shaving is a concept widely used across various sectors, including energy distribution and supply chain management, to reduce peak demand and balance resource utilization. In this research, the principles of peak shaving are adapted to the context of BHSs at airports, specifically targeting the occupancy levels at transfer infeed points.

Chapter 4 provides a detailed exploration of the peak shaving concept, including its application in other industries and its adaptation to the unique challenges of baggage handling at a hub airport. By explaining the underlying mechanisms and objectives, the chapter sets the foundation for understanding how peak shaving can be achieved in this research.

This section serves as an essential link between the analysis of the current state and the subsequent development of simulation models and distribution strategies. Incorporating peak shaving into the methodology ensures that the research focuses on addressing one of the most critical challenges in transfer baggage handling: mitigating peak demand at infeed points.

2.3 Data Collection

Following the current state analysis, historical baggage data from AAS was collected to form the basis of the simulation. This dataset focused on the arrival times, quantities, and entry points of transfer baggage over a 10-week period, capturing realistic inflow patterns. Additionally, parameters and boundary conditions essential for the simulation were gathered through interviews with key stakeholders, ensuring that the model reflected operational constraints and priorities.

The key elements of data collection included:

- **Baggage Arrival Times:** Arrival timestamps for transfer baggage were used to simulate the inflow to the BHS accurately. This data provided a clear view of peak and off-peak periods throughout the day.
- **Baggage Volumes:** Quantities of baggage arriving at various times helped identify trends in inflow and provided input for simulating realistic peaks and troughs in occupancy levels.
- **Entry Points:** Information on where baggage entered the system, such as specific transfer infeed points, was used to map the distribution of baggage flows across the BHS.

In addition to historical data, essential parameters and constraints for the simulation were derived from interviews with:

- **KLM Ground Handlers Management:** Discussions with ground handlers offered practical insights into the operational challenges and decision-making processes involved in managing baggage flows, particularly at critical transfer infeed points.
- **Baggage Operations Experts at AAS:** Stakeholders from the baggage handling team at AAS provided detailed information about the functionality of the BHS, including constraints on storage capacity, transport systems, and timing requirements for handling cold and super cold transfer baggage.

The insights from these interviews were instrumental in defining the simulation's boundary conditions, such as the maximum allowable buffer size, constraints on processing times, and the availability of resources like AGVs. The combination of real-world data and expert input ensured that the simulation model captured both the complexity and the practical realities of baggage handling operations at AAS.

By integrating historical data with qualitative insights from stakeholders, this research established a robust foundation for the simulation, enabling the testing of distribution strategies under realistic and operationally relevant conditions.

2.4 Simulation Model

The simulation model focuses on the occupancy levels at the transfer infeed points and models how baggage is handled from the moment it arrives at the airport, until it is fed into the BHS. Historical data is used to accurately reflect baggage arrival times and volumes, but from this point onward, the routing of baggage to the BHS is simulated. This includes:

- **Baggage Assignment to Containers or Carts**: Simulating how baggage is assigned to specific containers or carts for transport.
- **Transportation and Routing**: Simulating the movement of baggage from its arrival gate to a designated transfer infeed point, accounting for potential delays or routing changes.

The simulation will form the basis for testing different distribution strategies and evaluating how effectively they achieve peak shaving. More details on the simulation process will be provided in the simulation chapter 6.5.

The simulation incorporates the use of AGVs and buffer operations to evaluate the feasibility and resource implications of the proposed distribution strategies. AGVs are modeled to handle two main tasks: transporting containers to the buffer and delivering them to transfer infeed points. The buffer is simulated as a dynamic system to assess occupancy levels over time, providing insights into capacity needs and the impact of different strategies.

By including AGV requirements and buffer utilization in the simulation, this research ensures that operational constraints and resource demands are realistically captured, supporting the evaluation of each strategy's practical applicability and efficiency.

2.4.1 Simulation Validation

To ensure that the simulation accurately reflects real-world conditions, a validation process will be carried out. The simulation outputs will be compared to historical data, focusing on the accuracy of occupancy levels at transfer infeed points. Other KPIs, such as baggage volumes, will also be considered

Additionally, a sensitivity analysis will be conducted to test how small variations in input parameters (e.g., changes in baggage arrival times or volumes) affect the results. This will help ensure the model's robustness under different conditions.

2.5 Time Series Forecasting Model

Once the simulation model is validated, its output over the 10-week period will be used to create a time series forecasting model. This model will analyze the simulated data to predict future occupancy levels at the transfer infeed points, offering insights into when peaks and troughs are likely to occur. This is necessary to support the distribution strategies in determining the optimal timing for feeding buffered baggage into the BHS.

The time series model will be based on patterns observed in the output of the simulation, enabling proactive distribution strategies by informing the optimal timing for reintroducing buffered cold transfer baggage into the system.

2.6 Distribution Strategy Design

The core of this research involves the design and testing of some distribution strategies for buffered cold transfer baggage. These strategies will focus on determining the most effective timing and method for reintroducing baggage into the BHS, aiming to reduce peaks in occupancy levels at infeed points.

Rule-based strategies and algorithms will be developed to govern how and when baggage is released from buffers. Each strategy will be simulated to evaluate its effectiveness in achieving peak shaving while ensuring that all baggage reaches its destination on time.

2.7 Scenario Testing

To evaluate the effectiveness and robustness of each distribution strategy, three operational scenarios have been tested. These scenarios reflect realistic challenges that the BHS may encounter, allowing for a comprehensive assessment of each strategy's performance. The scenarios are as follows:

- **Scenario 1: Normal Situation**
  This scenario represents typical operational conditions at AAS, where all transfer infeed points are functioning normally, and baggage flows are at expected levels. This baseline scenario serves as a reference point for comparing the performance of the distribution strategies under regular circumstances.
- **Scenario 2: Reduced Capacity**
  In this scenario, one of the four transfer infeed points is out of service, creating a bottleneck in the system. This condition tests the distribution strategies' ability to manage baggage flow effectively under constrained resources, emphasizing the importance of efficient buffering and distribution during peak times.
- **Scenario 3: Increased Inflow of Transfer Baggage**
  This scenario simulates a surge in transfer baggage inflow, exceeding normal operational levels. The goal is to assess how well the distribution strategies cope with high demand and whether they can maintain effective peak shaving despite the increased pressure on the BHS.

Each scenario aims to test the adaptability and efficiency of the distribution strategies, offering insights into how they perform across a range of real-world operational conditions. This scenario testing approach provides a nuanced understanding of each strategy's resilience and potential weaknesses, guiding the selection of the most effective peak shaving methods.

2.8 Analysis

Following the scenario testing, a detailed analysis will be conducted to evaluate the performance of each distribution strategy. The primary metrics for analysis include:

- **Peak Occupancy Level Reduction:** Assessing the extent to which each strategy successfully reduces peak occupancy levels at the transfer infeed points.
- **Robustness Across Scenarios:** Analyzing the consistency of each strategy's performance across different scenarios with multiple KPIs to determine its robustness and adaptability.
- **Feasibility and Practicality:** Considering the feasibility of implementing each strategy in real airport operations, including any logistical or resource limitations. By looking at the size of the buffer needed and an insight in the number of tugs/Autonomous Guided Vehicles (AGV) needed for the operation.

The analysis will provide a comparative overview of the strategies, highlighting their strengths and potential trade-offs. By examining the results of scenario testing and the forecasting data, this analysis aims to identify the distribution strategy that offers the most significant improvement in peak shaving while maintaining operational reliability. The findings will support recommendations for practical implementations at AAS and provide a foundation for future research in airport BHS optimization.

3 The Baggage Journey

This chapter will explain the baggage journey. Figure 3.1 provides a clear overview of the various steps a suitcase can go through at a hub airport. These steps cover the entire process, from checking in baggage, processing it within the BHS, and finally unloading or transferring it to the correct destination. Each step in this process is essential for the efficient operation of an airport, especially at hub airports, where thousands of suitcases are processed daily. A well-functioning BHS system allows the airport to handle baggage quickly and safely, minimizing the processing time for passengers and their baggage. This chapter offers a detailed explanation of the infeed, the operation of the BHS, and the outfeed of baggage. The combination of human staff, automated systems, and technology makes it possible to run this complex process smoothly. This chapter gives answers to subquestion 1: *What is the current state of baggage handling processes at the transfer infeed points at AAS?*

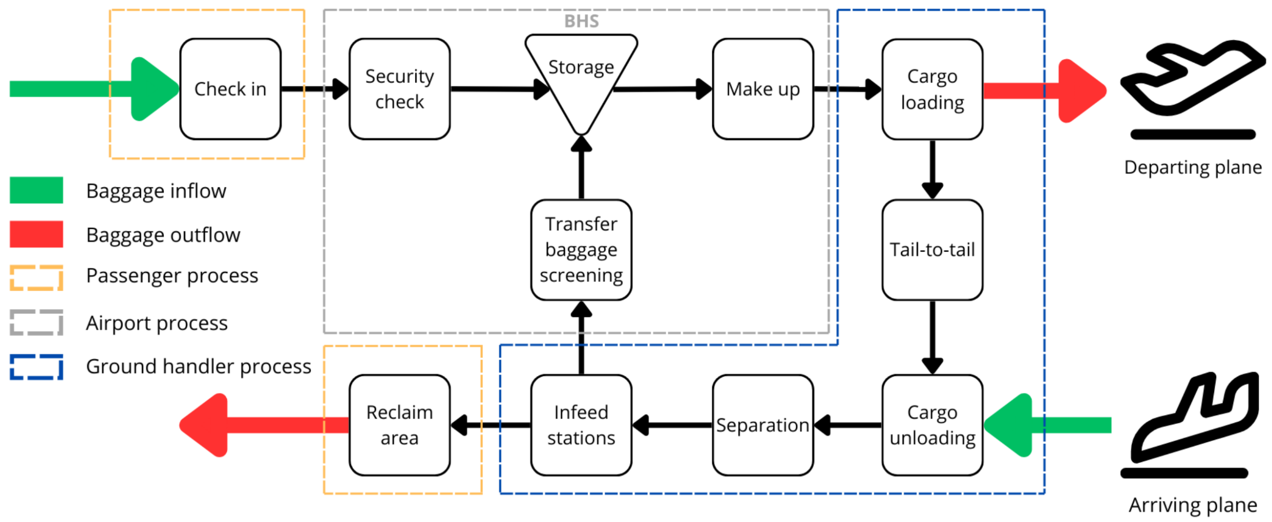


Figure 3.1: Visualization of the baggage journey

3.1 Infeed of Baggage

The infeed of baggage, or the intake of baggage into the BHS, can occur in two ways: via landside and airside. These two routes are essential for baggage processing, as they mark the beginning of the journey that baggage takes through an airport. Landside refers to the baggage checked in by passengers at the counter or self-service kiosks. Airside involves the baggage arriving on incoming flights, which is immediately handled for further processing or transfer to other flights. Coordinating these two streams is crucial to avoid system bottlenecks and ensure that baggage arrives at the correct destination on time. Since landside and airside have different characteristics and challenges, each type of infeed requires a different approach and handling. For example, baggage entered landside has a direct relationship with the passenger and their travel plan, while airside baggage usually comes from transfer passengers and, therefore, needs stricter time handling.

3.1.1 Landside Infeed

Landside infeed refers to baggage checked in by passengers at airport check-in counters. This process begins once the passenger offers their baggage at a check-in counter or self-service kiosk. The baggage is tagged with a label that includes the final destination and flight information and then fed into the system. Once entered, the baggage is processed and screened by the BHS. At many modern airports, this process is largely automated, with baggage transported on conveyor belts and automatically scanned for content and destination. This automation ensures that baggage is handled quickly and efficiently. Landside infeed is a crucial part of the baggage process, as it ensures that baggage enters the system before the passenger proceeds to the gate. It is also important for security, as every checked bag is thoroughly inspected before it is allowed on the plane.

3.1.2 Airside Infeed

Airside infeed refers to baggage arriving with incoming flights, either as transfer baggage or as baggage for its final destination. This baggage can be transported in various ways, such as in containers (e.g., AKH or AKE containers, see Figures 3.2c and 3.2b) or as bulk (loose suitcases, see Figure 3.2a). Once the plane has landed, the baggage is unloaded and sorted depending on its type: some bags remain at the airport for transfer to other flights, while others go to the reclaim area for passengers arriving at their final destination. Transfer baggage is further categorized. Transfer baggage with a short connection time is processed urgently to ensure it is reloaded on time for the next flight. Transfer baggage with longer connection times is given lower priority. This process requires close coordination between ground handling teams and the BHS, as timely transfer is essential to prevent delays.

Additionally, the type of aircraft used to transport the baggage significantly impacts how the baggage is processed. The larger Wide Body (WIBO) aircraft, typically used on long-haul flights, often use multiple AKE containers (see Figure 3.2b) to transport baggage. In a WIBO aircraft, two AKE containers can be placed side by side in the cargo hold, allowing for efficient storage and rapid processing of baggage. The large capacity of these aircraft enables more baggage to be processed per flight, which is especially important for busy international hubs. The use of containers in WIBO aircraft speeds up the loading and unloading process and protects baggage from damage.

On the other hand, most Narrow Body (NABO) aircraft use bulk baggage (see Figure 3.2a), where suitcases are placed loose in the cargo hold without containers. NABO aircraft are often used for short- and medium-haul flights and have a smaller cargo hold compared to WIBOs. Using bulk baggage in NABOs means that the suitcases must be loaded and unloaded individually, which can take more time and be more labor-intensive for ground staff. However, some NABO aircraft use AKH containers (see Figure 3.2c). These smaller containers fit into the cargo hold of NABOs and offer similar protection and efficiency as AKE containers in WIBO aircraft but on a smaller scale. This is particularly useful at airports with high baggage volumes, where speed and efficiency are crucial for timely departures.

Therefore, airside infeed places high demands on sorting and storage processes, especially when baggage arrives in bulk, meaning that suitcases must be processed individually. Moreover, baggage must be screened quickly and directed to the appropriate infeed stations for further processing. A well-organized infeed process contributes to the overall efficiency of baggage handling and minimizes the risk of delays for passengers and flights.

(a) Bulk baggage ("Bagagekarretje Met Koffers", n.d.)

(b) AKE container ("Baggage Container", n.d.)

(c) AKH container ("Baggage Container", n.d.)

Figure 3.2: Different types of baggage transportation

3.2 Baggage Handling System

The BHS is a crucial infrastructure that enables the complete processing of baggage at an airport. The system consists of multiple components working together to safely and efficiently transport baggage from one location to another. Screening, storage, sorting, and make-up are some of the key elements within this system. Automated conveyor belts, scanners, and sorting systems ensure that each suitcase is directed to the correct destination without delays or errors. Each part of the BHS fulfills a specific role, from checking baggage for security risks to temporarily storing suitcases waiting for their next flight. Thanks to technological advances, the BHS has become increasingly sophisticated, leading to faster processing times and higher levels of safety and reliability. This section explains how each component of the BHS works and contributes to overall baggage handling.

3.2.1 Screening

Screening is one of the first steps in the BHS process and is critical for the safety of the airport and flights. All baggage entering the BHS via landside or airside must be screened for prohibited or dangerous items. This process usually involves advanced X-ray and scanning technologies that quickly and accurately analyze the contents of the baggage. Suspicious baggage can be separated for further inspection or manually inspected by security staff. Thanks to automation and technological developments, most baggage can be screened within seconds without slowing down the system's throughput. The screening process is an essential part of overall airport security, ensuring that passengers can travel safely without the risk of dangerous materials on board. Besides the security aspects, the speed of this process also plays a significant role in the efficiency of baggage handling. A well-functioning screening system can minimize delays and reduce processing time.

3.2.2 Storage

Storage within the BHS is intended for baggage that cannot be processed immediately. This can occur for various reasons, such as when there is a waiting period between the arrival and departure of the next flight or when baggage is checked in earlier than needed. The storage facilities within the BHS are often automated systems that can hold baggage temporarily without disrupting the flow of other suitcases. Airport baggage storage can have a centralized or decentralized architecture. In a centralized architecture, there is one central storage, whereas decentralized storage consists of several storage locations (Lin et al., 2015). The storage system uses computer-controlled algorithms to ensure that the baggage is released for further processing at the right time. This helps streamline handling, particularly for transfer baggage that has a longer waiting period. The use of storage within the BHS helps manage peaks in baggage processing and optimize the occupancy of infeed stations and other critical points. In cases where baggage needs to be stored for an extended period, monitoring temperature and other environmental factors ensures the baggage remains in optimal condition.

Figure 3.3: Storage in baggage handling hall (Duell, n.d.)

3.2.3 Sorter System

The sorting system in the BHS is responsible for separating baggage and directing it to the correct destination within the airport. This is based on information such as flight details, transfer plans, and final destination of the baggage. Modern sorting systems use automated technologies such as barcode readers and RFID systems to quickly and accurately identify each suitcase. The sorting system ensures that the baggage is directed to the correct gate or transfer location, so it is ready for the next phase of the process. This can range from transferring baggage to another flight to sending suitcases to the reclaim area for passengers who have completed their journey. The speed and efficiency of the sorting system are crucial for the overall performance of the BHS, especially at busy airports where thousands of suitcases need to be processed per hour. The sorting system also uses redundancy and error handling to ensure that misrouted baggage can be corrected quickly without disrupting the entire process.

3.2.4 Make Up

The make-up phase in the BHS is the final step before the baggage is handed over to ground staff for loading onto the plane. In this phase, the baggage is gathered and organized based on flight information and loading priorities. The baggage is consolidated into containers or carts, which are then transported to the aircraft for loading. This process is carefully planned to ensure that the baggage is loaded in a way that optimizes space on the aircraft while also ensuring a quick and efficient unloading at the next destination. The make-up process often requires close collaboration between the BHS and ground staff, as timing is critical to prevent delays. It is also important that the baggage is loaded in the correct order, especially for transfer flights where some baggage needs to be unloaded before others. Through advanced planning and coordination, the make-up phase runs smoothly without disruptions.

To assist ground staff with lifting and packing baggage into carts or containers, various tools are used. Two of these tools are the 'Baggage Manipulator' (see Figure 3.4a) and the 'Baggage Loader' (see Figure 3.4b). The Baggage Manipulator helps staff move and position suitcases, reducing the need to lift heavy baggage and making the process more efficient. This is especially helpful when filling containers, where precise placement is needed to optimize space. The Baggage Loader ensures that suitcases are raised to an ergonomic height, so staff do not have to constantly bend and lift, contributing to safer and faster baggage handling. The use of these tools reduces the physical strain on staff and increases the efficiency of the baggage handling process, which is essential in a busy airport environment.

(a) Baggage manipulator

(b) Baggage loader

Figure 3.4: Tools to support the baggage handling process ("Baggage Make-up and Unloading Systems | Beumer Group", n.d.)

3.3 Outfeed of Baggage

The outfeed of baggage refers to the baggage leaving the airport's baggage system: the flow of baggage either to passengers in the reclaim area or to planes for departing flights. This process is just as important as the infeed and the BHS, as errors or delays in this phase directly affect the passenger experience and flight efficiency. Properly handling the outfeed ensures that passengers can quickly retrieve their baggage upon arrival and that departing flights leave on time with the correct baggage on board. Like the infeed, the outfeed can occur landside or airside, depending on the destination of the baggage.

3.3.1 Landside Outfeed

The landside outfeed refers to the process of delivering baggage to the reclaim area, where passengers can retrieve their suitcases after arrival. Once the plane has landed, the baggage is unloaded and transported via the BHS to the reclaim area. Here, the suitcases are placed on conveyor belts where passengers wait to collect their baggage. The reclaim process is one of the final steps in the passenger journey and is, therefore, crucial for the overall passenger experience. Quick and efficient handling in the reclaim area ensures satisfied passengers and smooth airport operations. This system often uses sensors and automation to ensure that baggage arrives in the correct order and without delays. The reclaim area is often equipped with multiple baggage belts to handle peak times and large numbers of passengers efficiently. Coordination between the BHS and ground staff ensures that the baggage is transferred to passengers quickly and without errors.

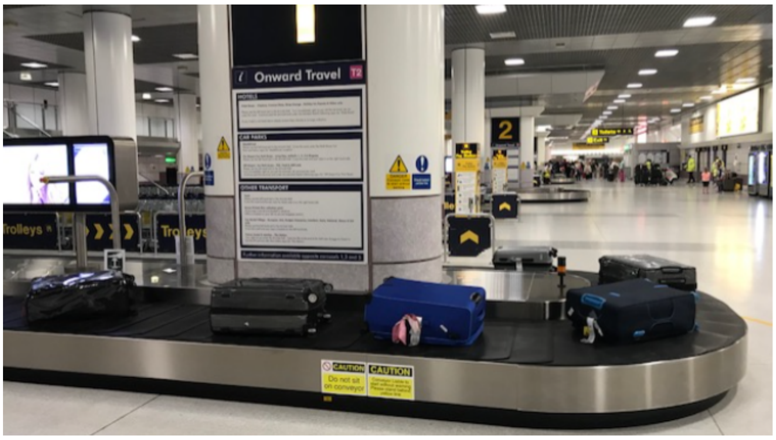


Figure 3.5: Lateral at reclaim area (Pickering, n.d.)

3.3.2 Airside Outfeed

The airside outfeed involves loading baggage onto planes for departing flights. After the make-up phase, in which the baggage is sorted and prepared, it is transported to the plane and carefully loaded. This process requires precise planning to ensure that the baggage is placed correctly, both to save space and to ensure that the baggage can be unloaded in the right order upon arrival. Loading baggage is a time-critical operation, especially at busy airports where multiple flights are being prepared simultaneously. Ground staff work closely with the BHS to ensure that all baggage is on board in time and that no delays occur due to missing or incorrectly labeled suitcases. An error-free airside outfeed contributes to the on-time departure of the flight and ensures that passengers can trust their baggage will arrive at the correct destination.

3.4 Key Performance Indicators

Key Performance Indicators (KPIs) are widely used in airport BHS to measure and optimize performance. These indicators provide valuable insights into system efficiency, reliability, and potential bottlenecks, enabling operators to identify areas for improvement. Below are some common KPIs used to assess BHS operations:

- **System capacity (bags/hour):** This KPI measures the maximum number of bags the BHS can process per hour. A higher capacity indicates that the system can handle large volumes of baggage efficiently, which is critical for busy international airports.
- **Mishandled baggage (total number of bags mishandled):** This KPI tracks the number of bags that are lost, delayed, or misrouted. A low mishandling rate is vital for maintaining passenger satisfaction and minimizing operational disruptions.
- **Infeed capacity:** This KPI evaluates the maximum baggage intake rate at landside and airside entry points, ensuring that the system can handle peak inflow periods without congestion.
- **Buffer efficiency:** This indicator measures how effectively the buffer is utilized, focusing on how well baggage is stored and retrieved without causing delays or disruptions in the system.
- **On-time delivery rate:** This KPI reflects the percentage of baggage delivered on time to its destination, whether to the reclaim area for passengers or to the aircraft for departing flights. A high on-time delivery rate is crucial for both operational success and passenger satisfaction.

While these KPIs are commonly used to monitor and improve BHS performance, this research focuses on a subset of KPIs that are directly relevant to the objectives of optimizing transfer infeed processes and evaluating the impact of distribution strategies for buffered baggage. The KPIs selected for this study include:

- **Peak occupancy at transfer infeed points:** This KPI measures the maximum number of bags present at transfer infeed points during peak periods. The primary goal of this research is to reduce these peaks through effective buffering strategies, thereby minimizing operational stress on the system.
- **Peak occupancy in the buffer:** This KPI evaluates the maximum load on the buffer over time, ensuring that buffering strategies do not create new bottlenecks while aiming to optimize system-wide efficiency.

- **Required number of AGVs:** The number of AGVs needed is calculated based on the transport demand created by buffering operations. This KPI provides insights into the logistical implications and feasibility of each distribution strategy.

By integrating both commonly used KPIs and those tailored to the objectives of this research, the study provides a comprehensive evaluation framework. The traditional KPIs ensure that the simulation adheres to established performance standards for BHS, while the study-specific KPIs offer a focused analysis of buffering and transfer infeed strategies.

In the *Results* chapter (9), these KPIs are used to compare the performance of different distribution strategies, highlighting trade-offs between reducing peak occupancy, optimizing buffer usage, and minimizing the logistical demands on AGVs. This dual-layered approach ensures that the findings are both operationally relevant and aligned with the overarching goals of the research.

4 Peak Shaving

This chapter discusses the concept of peak shaving as it applies to BHSs at airports. We will explore how peak shaving strategies can be implemented to balance the inflow of baggage and reduce congestion at critical points in the system. The chapter outlines various approaches to mitigating high occupancy rates at the infeed points, including strategies for managing check-in baggage and cold buffering of transfer baggage. This chapter gives answers to subquestion 2: *How does peak shaving work in the context of an airport?*

4.1 Literature on Peak Shaving

Peak shaving is a phenomenon that applies to many sectors. For instance in the energy sector, it is a strategy used to reduce the amount of energy consumed during periods of high demand. While commonly associated with energy management, peak shaving also applies to managing occupancy levels in various systems, such as buildings, transportation, and telecommunication networks. The goal is to optimize resource use, enhance efficiency, and prevent overloads by spreading the demand more evenly over time.

In buildings, peak shaving on energy consumption can be achieved by distributing occupancy more evenly throughout the day. Strategies include flexible working hours, remote work options, and scheduling software to prevent overbooking. These methods can significantly reduce energy consumption and improve occupant comfort and productivity. For example, research on energy storage systems in buildings demonstrates the effectiveness of adaptive control algorithms in optimizing peak shaving and improving energy efficiency (Chua et al., 2016).

In public transportation, peak shaving aims to alleviate congestion by encouraging off-peak travel through dynamic pricing and increased service frequency during peak times. These measures can reduce overcrowding and improve overall system efficiency. A study on integrating liquid air energy storage with power plants for bidirectional peak shaving highlights the importance of such strategies in enhancing grid performance and reducing peak loads (Gao et al., 2021).

Peak shaving in telecommunication networks involves managing data traffic to prevent congestion. Techniques such as traffic shaping, bandwidth throttling, and promoting off-peak usage help maintain service quality and reduce latency during high-demand periods. Research indicates that these methods lead to more stable and reliable network performance (Tziovani et al., 2021).

Peak shaving of occupancy levels is an effective approach to optimizing the use of resources and enhancing system efficiency across various domains. By implementing strategies to spread demand more evenly, organizations can prevent overloads, improve service quality, and reduce costs. Adopting peak shaving measures can lead to significant benefits in building management, transportation systems, telecommunication networks, and other systems.

4.2 Peak Shaving on Baggage Occupancy Levels at an Airport

In the previous chapter, we outlined the baggage journey and discussed some important KPIs, including the infeed capacity of the BHS. A significant issue at many airports, particularly at hub airports like AAS, is that the number of incoming and outgoing flights is not evenly distributed throughout the day. This results in fluctuations in the inflow of transfer baggage, with peaks and troughs that put pressure on the efficiency of the BHS. One of the strategies to mitigate these peaks is peak shaving.

4.2.1 Check-in Baggage and Peak Shaving

One way to achieve peak shaving is by processing check-in baggage earlier. By encouraging passengers to check in their baggage earlier, for example, the evening before the flight or well before peak hours, the load on the system during peak times can be reduced. The idea is that by shifting the load of check-in baggage forward, more capacity is available for processing transfer baggage during the busy periods. This process is illustrated in Figure 1.3 in Chapter 1.2, where the peak of check-in baggage is shifted to a quieter time of day, while the peak of transfer baggage is moved to a later time.

4.2.2 Cold Buffering of Transfer Baggage

As discussed earlier, transfer baggage can be classified into different temperature categories based on the layover time, where "warm" baggage is prioritized for processing. "Cold" transfer baggage, with longer layover times, can be temporarily stored (buffered) and reintroduced into the system later during quieter periods. This is an effective way to reduce peaks in the occupancy levels of the infeed points, without delaying the baggage for its departing flight.

Figure 4.1 shows how buffering cold transfer baggage adds an extra layer to the baggage journey. In this visualization, cold baggage is kept separate from warm baggage and can be temporarily stored. After being buffered, the now "warmed-up" baggage can be reintroduced into the BHS later, leading to a more even distribution of the occupancy levels throughout the day.

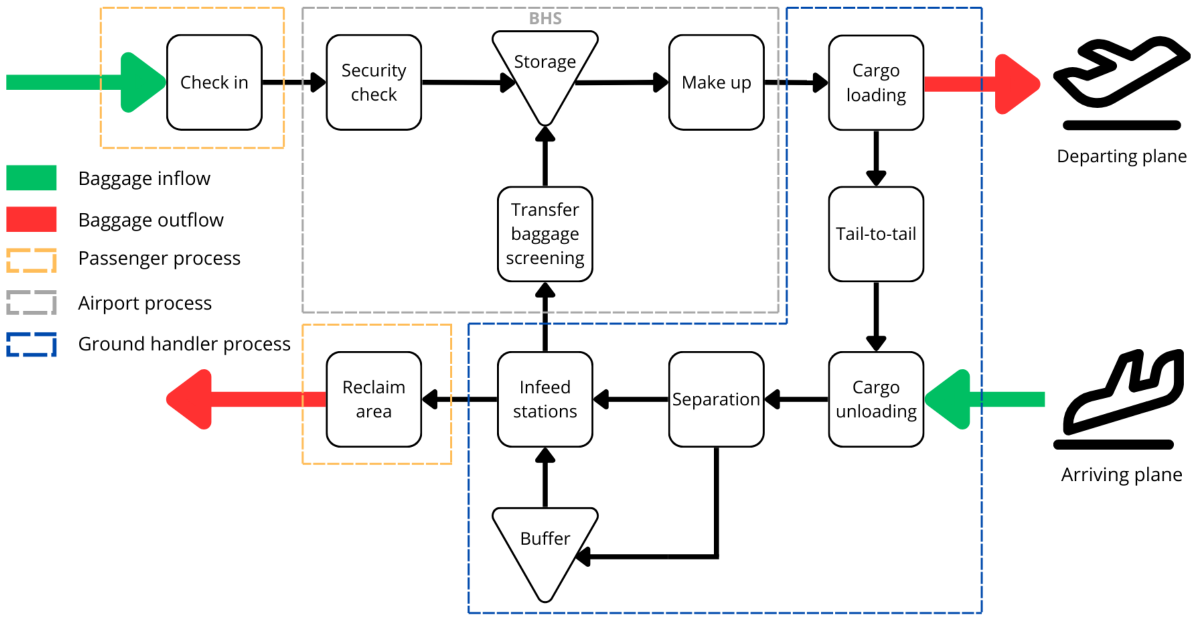


Figure 4.1: Visualization of the baggage journey with buffering.

4.2.3 Design Options for Cold Buffering

There are various options and strategies for how cold baggage can be buffered and then distributed. One of the questions is how to determine whether baggage should be classified as warm or cold, as this affects the priority of processing. There can also be differences in how carts or containers are packed, which impacts the distribution of each piece of luggage. Additionally, there is the question of how the baggage should be transported on the airside. There are increasing innovations that further optimize transport processes. Buffer configuration and buffer placement are also aspects to consider.

Baggage Classification Baggage can be split into different categories. For example, there is transfer baggage classified as tail-to-tail, where the baggage has a shorter layover time than the minimum processing time through the BHS. This baggage needs to be taken directly to its next flight. Then there is ShoCon baggage, which has enough time to go through the BHS but must be processed immediately. There is also reclaim baggage on board, which does not transfer but is transported via the BHS to the reclaim areas on landside for passengers to collect. Finally, there is cold transfer baggage. This baggage has a longer layover time and can be buffered to alleviate the BHS from peaks. Table 4.1 provides a rough estimate of which layover times correspond to each classification. The layover time is considered as the difference between the Actual In Block Time (AIBT), the moment when the aircraft comes to a stop on the apron, and the Scheduled Off Block Time (SIBT), the moment the departing aircraft begins to move.

Baggage Temperature	Description
Tail-to-tail	Connection time <45 minutes
ShoCon	45 <Connection time <90
Cold transfer	90 <Connection time

Table 4.1: Example of possible baggage classification

Container Packing The use of containers or carts offers variety in packing, where choices can be made based on temperature or destination. One option could be to place baggage for the same baggage hall in the same container, even if the departure times of the baggage slightly differ. This could make transportation more efficient, as one container can be brought directly to the correct infeed location rather than making multiple trips along the same route. Alternatively, containers can be packed based on baggage temperature. This will be further discussed in Chapter 6.5.2.

Transport on Airside: Autonomous Guided Vehicles There are several ways to transport baggage on the airside. Currently, the most common method is the tug, which is a vehicle driven by a driver that pulls a train of carts carrying baggage. AAS, along with many other airports, aims to become more sustainable and autonomous (Schiphol, 2024). An emerging innovation in airport logistics is the use of autonomous guided vehicles (AGVs) for the transportation of baggage. The problem addressed by this research is the need to relieve the BHS during peak times. Relieving the BHS is not necessary throughout the entire day, but can be only during specific periods. As mentioned before, it is challenging to employ staff for short periods, and this is where AGVs could play a significant role.

Figure 4.2: Aurrigo AV with baggage ULD (Aviation Week Network, 2024)

AGVs are self-driving vehicles that are designed to transport baggage directly from the baggage handling area to the aircraft and vice versa. In Figure 4.2, a picture of Aurrigo's AGV capable of transporting a ULD (AKE- or AKH-container) is shown, also with the ability to tow carts that carry more ULDs. The implementation of AGVs at airports could revolutionize the efficiency and reliability of baggage transfer. AGVs are equipped with advanced navigation systems that allow them to maneuver safely and efficiently through the complex environment of an airport. By automating the transport of ULDs, airports can optimize traffic on the airside. AGVs could be used to facilitate the transport of ULDs to and from external buffers, partially resolving the personnel issue caused by fluctuating transfer baggage flows. The potential use of AGVs represents an exciting development in the ongoing effort to innovate and improve airport logistics.

Buffer Configuration A buffer can be configured in different ways, depending on the specific needs of the airport and the nature of the baggage flows. A commonly used method is to configure the buffer based on flight departure times, so that baggage that needs to depart first is placed at the front of the buffer. This minimizes the time required to retrieve baggage from the buffer when it needs to be reintroduced into the BHS. Another option is to sort baggage based on destination or gate, so that all baggage for the same destination is placed in the same section of the buffer. This speeds up the transportation of baggage to the correct infeed stations.

Although buffer configuration offers important advantages for efficiency, this study only looks at the timing of baggage release from the buffer, without addressing the internal configuration of the buffer.

Buffer Placement Buffers can be placed in different locations at the airport, and the choice between a centralized or decentralized system can have major implications for the efficiency of baggage processing. A centralized buffer is located in one place close to the BHS, offering advantages in terms of control and logistics, as all baggage can be managed from a single location. However, the downside of a centralized buffer is that transporting baggage to distant infeed stations can take longer, especially during peak hours when airside traffic increases.

A decentralized system, on the other hand, distributes buffers across different locations, usually closer to the gates or specific infeed stations. This reduces transport time and spreads the load of traffic over a larger area, which can reduce congestion. Previous research by van der Grift (2023) showed that having multiple decentralized buffers is more advantageous for AAS, as this not only reduces transport time but also increases operational flexibility. While buffer placement is an important consideration in optimizing baggage flows, it falls outside the scope of the simulation carried out in this study.

5 Data

This research uses approximately 10 weeks of baggage data from AAS, covering the period from April 15, 2024, to June 30, 2024. The dataset contains several key pieces of information on each baggage item's journey through the BHS, including timestamps, locations, and flight details. These data points come from Baggage Source Messages (BSM), combined with scans made at different processing points at AAS.

The BSM is an electronic message that provides crucial information about a piece of baggage. It is widely used in aviation to manage and track baggage flow. The BSM includes details such as flight numbers, departure times, destinations, and other relevant information about the baggage.

5.1 Details of Dataset

Table 5.1 summarizes the most important columns included in the dataset, along with their descriptions.

Column name	**Description**
DATETIME OF ENTRY	Timestamp when the baggage entered the BHS
ENTRY AREA	The area in the airport where the baggage entered the BHS
ENTRY FUNCTION	The function of the baggage at the entry point (TransferIn or Check-In)
SIBT	Scheduled In-Block Time, when the inbound flight was scheduled to park
AIBT	Actual In-Block Time, when the inbound flight parked at the gate
IN FLIGHT DESIGNATOR	The designator for the inbound flight associated with the baggage
IN RAMP	The ramp location where the inbound flight was processed
IN AIRPORT	The airport from which the inbound flight originated
IN AC IATA	The IATA aircraft code for the inbound flight
DATETIME OF EXIT	The timestamp indicating when the baggage exited the BHS system
EXIT NAME	The name of the exit point for the baggage
EXIT AREA	The area in the airport where the baggage exited the BHS
SOBT	Scheduled Off-Block Time, when outbound flight is scheduled to leave the gate
AOBT	Actual Off-Block Time, when the outbound flight actually left the gate
OUT FLIGHT DESIGNATOR	The designator for the outbound flight associated with the baggage
OUT RAMP	The ramp location where the outbound flight was processed
OUT AIRPORT	The airport to which the outbound flight is heading
OUT AC IATA	The IATA aircraft code for the outbound flight

Table 5.1: Dataset columns

The initial dataset contained 4,778,334 rows, where each row stands for a single piece of baggage. However, after applying several filtering and cleaning steps, the final dataset used for analysis was reduced to 2,094,561 rows. The next sections explain the various data-related manipulations.

5.2 Data cleaning and filtering

The dataset was cleaned in Python using multiple packages, including the Pandas library. Since large datasets can sometimes have malformed rows, a custom handler was implemented to count and skip such rows without crashing the loading process. Notably, there were no bad lines in this dataset.

To ensure the integrity of the data, the column names were stripped of unnecessary whitespace.

Several filtering steps were applied to clean the dataset:

- Rows with `ENTRY_FUNCTION` equal to `TransferIn` were retained. This is because this research focuses on occupancy rates at transfer infeed points, and thus reclaim baggage is excluded from the analysis.
- Rows with missing values in crucial time fields (`SIBT`, `AIBT`, `SOBT`, and `AOBT`) were removed.

After applying these filters, the dataset was reduced to 2,094,561 rows, or bags.

Conversion of time columns
The columns related to scheduled and actual times, such as `SIBT`, `AIBT`, `SOBT`, and `AOBT`, were converted into a

proper datetime format to ensure consistency during analysis.

Conversion of columns to appropriate datatypes
To ensure that the dataset is correctly structured for further analysis, specific columns were converted to their appropriate types. Time-related columns were converted to datetime, and others, such as categorical columns, were cast to object types. This conversion was essential to facilitate filtering and group operations in subsequent steps of the analysis.

Adding the `ODD_SIZED` column
A new column, `ODD_SIZED`, was added to label baggage as either "Odd sized" or "Normal". This was based on specific keywords found in the `EXIT_NAME` column. Baggage that passed through certain specific sorter bands was labeled as "Odd sized", while the rest were considered "Normal." Since odd-sized baggage represents a small part of the total dataset (1.28%), it is excluded from the scope, partly due to its need for special transport.

Adding Bag Identifier
A unique identifier was added for each row, representing an individual baggage item. This identifier allows tracking of each bag's journey through the simulation, enabling detailed analysis at the baggage level.

Splitting datetime columns into date and time
For some parts of the analysis, it was necessary to separate the date and time components of datetime fields such as `DATETIME_OF_ENTRY` and `DATETIME_OF_EXIT`. This was achieved by splitting these columns into two separate fields: one for the date and another for the time. This facilitated easier time-based analysis in the simulation phase.

Table 5.2 shows the number of rows, and thus bags, before and after cleaning the dataset.

	Number of rows in dataset
Original dataset	4,778,334
Dataset after cleaning	2,094,561

Table 5.2: Dataset size before and after cleaning

The processed data forms the foundation for the simulation model described in the next chapter, enabling the evaluation of different strategies for optimizing baggage handling operations at AAS.

6 Simulation at Amsterdam Airport Schiphol

This chapter outlines the simulation framework developed to model baggage handling at AAS. The goal of the simulation is to compare different distribution strategies for handling super cold transfer baggage, using real-world data and logical assumptions regarding the operational steps involved. This section will provide an overview of the simulation's structure, including the journey of transfer baggage through the system, modifications to the dataset for modeling purposes, and the incorporation of key variables such as flight type, baggage temperature, and ramp clusters.

6.1 Main Goal

The main objective of the simulation is to create a realistic representation of baggage flow at AAS, particularly focusing on transfer baggage. The simulation models the distributing and buffering of transfer baggage, aiming to identify strategies that optimize resource utilization, reduce peak loads, and minimize delays.

6.2 The Transfer Baggage Journey

The journey of a transfer bag begins as soon as the inbound flight arrives at the gate (AIBT). After the flight parks, the baggage is unloaded and transported to its destination within the airport. The process is divided into several time segments:

- **T1:** The time from the AIBT to when the first bag or container is unloaded from the aircraft.
- **T2:** The duration required to unload all bags or containers from the aircraft.
- **T3:** The transportation time from the aircraft's parking position to the infeed point for ShoCon baggage.
- **T4:** If the baggage is cold, it may be temporarily buffered, with T4 representing the buffering time.
- **T5:** The time taken to place the baggage on the unloading quay upon arrival at the infeed point.

This process is visualized in Figure 6.1. The blue lines represent the cold process, where the baggage is stored temporarily, before becoming hot (red). The red line represents the hot process, where baggage is moved directly to the infeed point without buffering.

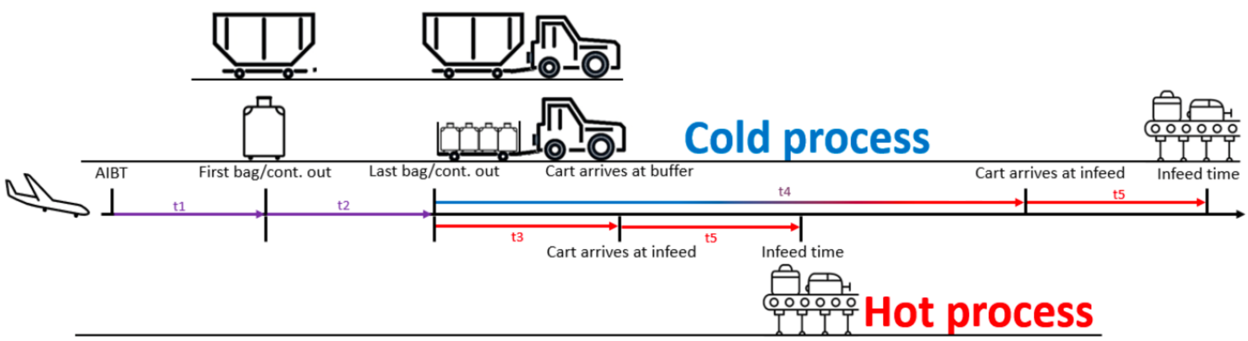


Figure 6.1: Transfer baggage journey from arrival to transfer infeed point (van der Grift, 2023)

6.3 Modifications to the Dataset

To create a realistic simulation, several modifications were applied to the cleaned dataset provided by AAS. These adjustments were necessary to simplify the simulation and improve computational efficiency while maintaining accuracy.

6.3.1 Flight Type

A new column for flight type (`FLIGHT_TYPE`) was added to the dataset to distinguish between Schengen and non-Schengen flights. Schengen flights involve shorter security checks, while non-Schengen flights undergo more stringent screening due to international regulations. The classification is based on the airports where the flights arrive from or depart to.

Flights arriving from or departing to airports listed in the Schengen zone were categorized as Schengen flights, while all other flights were categorized as non-Schengen. The dataset was modified accordingly using a Python function that checks the inbound and outbound airports to determine the flight type.

6.3.2 Baggage Temperature

Each piece of baggage was assigned a temperature classification based on the connection time between the AIBT of the inbound flight and the SOBT of the outbound flight. This connection time was used to calculate whether a bag could be classified as Tail-to-Tail, ShoCon, Cold Transfer, or Super Cold Transfer. Tail-to-Tail baggage, which does not pass through the BHS, was excluded from the simulation. Baggage temperatures were calculated based on the logic shown in Table 6.1.

Baggage type	**Interval (minutes)**	
	Schengen	**non-Schengen**
Tail-to-Tail	$30 < \text{connection time} \leq 50$	
ShoCon	$30 < \text{connection time} \leq 80$	$50 < \text{connection time} \leq 90$
Cold Transfer	$80 < \text{connection time} \leq 180$	$90 < \text{connection time} \leq 180$
Super Cold Transfer	$180 < \text{connection time}$	

Table 6.1: Baggage temperature classification based on connection time (Bunnik, 2018)

For each cold and super cold transfer bag, a time is added: `Latest Time Bag`. This time represents the latest moment a bag must arrive at one of the infeed points in order not to miss its outbound flight. According to Bunnik (2018), the BHS requires at least 25 minutes to transport a bag from an infeed point to its flight. The `Latest Time Bag` is calculated as: SOBT - 25 minutes.

6.3.3 Entry and Exit Area Cleaning

To standardize the dataset, adjustments were made to the `ENTRY_AREA` and `EXIT_AREA` columns. Specifically, instances of 'D' in the `ENTRY_AREA` were replaced with 'TSD', and the reverse was done for the `EXIT_AREA`. This was necessary to maintain consistency across the dataset. 'D' and 'TSD' are at the same location, baggage hall D, but one is for the entry and the other for the exit of baggage.

6.3.4 Baggage Type

The simulation required knowledge of the type of baggage being handled, based on the aircraft model for both inbound and outbound flights. The baggage type was categorized as either AKE (38 bags), AKH (28 bags), or bulk baggage, based on the aircraft model's IATA code. Bulk baggage is unloaded from the aircraft onto a cart, with each cart holding up to 30 bags. In the simulation, it is assumed that each cart can carry a maximum of 30 bulk bags.

Different aircraft can carry different types of baggage. For example, narrow-body aircraft like the A320 or 737 carry bulk baggage. Wide-body aircraft, including the 747 and 777 models, were classified as carrying AKE containers. The A320 specifically was categorized as carrying AKH containers. The baggage types for inbound and outbound flights were derived from the aircraft IATA codes and added to the dataset.

In the simulation, it is assumed that the capacities presented in Table 6.2 are used. Additionally, it is assumed that each cart or container can be filled to its full capacity. However, in reality, this is not always the case due to variations in the volume of each bag. For instance, a cart might be fully loaded with 25 bags on one occasion, while on another, it could hold up to 35 bags, depending on the size and shape of the baggage.

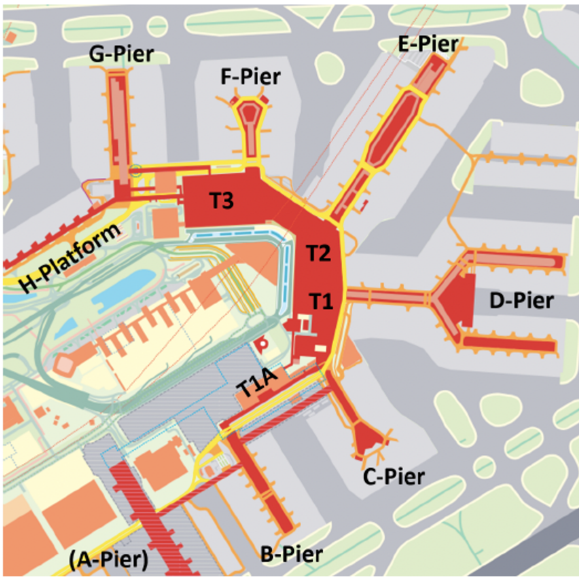
	Bulk (Cart)	**AKE-Container**	**AKH-Container**
Capacity	30	38	28

Table 6.2: Baggage type capacities

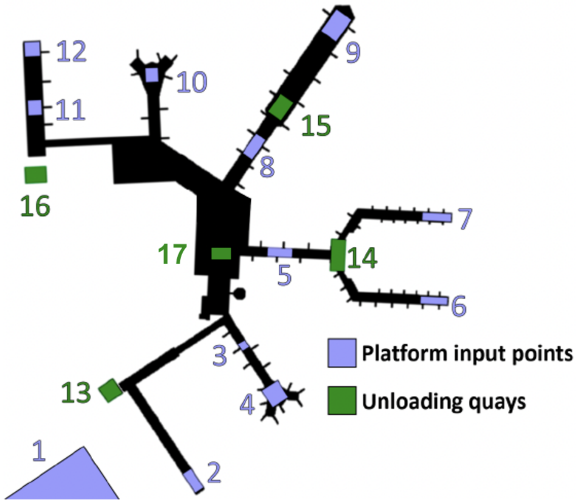
6.3.5 Ramp Clusters

When an aircraft arrives, it is assigned a ramp where passengers and baggage are unloaded. To reduce the complexity of the simulation, ramp positions were clustered. Instead of simulating all 150 stands individually, they were grouped into clusters. Each piece of baggage was assigned a ramp cluster based on the aircraft's parking position.

Figure 6.2 shows the layout of AAS, including both the piers and terminals, as well as the unloading quays and infeed points used in the simulation.



(a) Piers and terminals at AAS



(b) Platform input points and unloading quays

Figure 6.2: Layout of Amsterdam Airport Schiphol

Table 6.3 outlines the clusters and their descriptions, providing a simplified representation of the airport's layout for the simulation.

Nr.	Name	Description
1	A-pier + B-platform	A00 to A9 and B51 to B91
2	B-pier	B13 to B36
3	C-stem	C04 to C09
4	C-head	C10 to C18
5	D-stem	D02 to D08
6	D-fork south	D10 to D31
7	D-fork north	D41 to D99
8	E-stem	E02 to E07
9	E-head	E08 to E99
10	F-pier	F01 to F99
11	G-stem	G02 to G06
12	G-head	G07 to G80, H1 to H7, and M1 to M7
13	Unloading area South	4 unloading quays
14	Unloading area D	5 unloading quays
15	Unloading area E	2 unloading quays
16	Unloading area West	1 unloading quay
17	Unloading area Terminal 2	1 unloading quay

Table 6.3: Description of the platform input points shown in Figure 6.2b

6.4 Infeed points and unloading quays

Figure 6.2 shows a schematic representation of AAS. The locations of the different piers and terminals are shown, and the locations of the infeed points are indicated in green. The infeed points are actually located at level -1, which is underground.

Each infeed point consists of one or more unloading quays. As mentioned earlier, baggage arrives at the infeed points in bulk or in ULDs. These are then unpacked by ground handlers and placed individually on the unloading quay. From here, the automated BHS takes over the transportation of the baggage. Table 6.4 shows the different capacities of the infeed points.

Each infeed point has its own processing capacity, as shown in Table 6.4. The capacity varies between different infeed points, influencing how quickly they can process incoming baggage trains. For instance, some infeed points, like Infeed Point D, can handle higher volumes of baggage, while others, such as Infeed Point West, have a much lower capacity.

Nr.	Name	Capacity (per hour)
1	Infeed point South	2000
2	Infeed point D	4500
3	Infeed point E	3600
4	Infeed point West	900
5	Infeed point Terminal 2	500

Table 6.4: Capacity of infeed points (Schiphol, 2023)

The infeed point at Terminal 2 is rarely used and is mainly reserved for special baggage that requires reprocessing due to specific handling issues or irregularities.

6.5 Simulation

The simulation of baggage handling at AAS begins after applying the necessary modifications to the dataset, as described earlier. The simulation itself was built using the SimPy library, which allows for the modeling of complex processes with event-based simulations.

Through discussions with employees from both KLM and Schiphol, it became evident that the assignment of transfer baggage to infeed points is currently managed by a KLM team. Interestingly, different members of this team often follow varying practices based on their experience and intuition, resulting in an absence of clear distribution rules. However, the aim for KLM in the coming years is to establish fixed rules for this process, with a view toward automation on the airside. The simulation in this research adopts the rules that KLM and the ground handlers are planning to implement, which will be explained later in this chapter.

The choice of distribution types for various processes in the simulation is based on empirical data collected by AAS. According to Bunnik (2018), measurements taken during baggage handling operations at AAS were used to establish realistic ranges for certain activities, such as unloading times and transit times. Uniform and triangular distributions were chosen because they closely reflect the observed variability in these processes.

It is important to note that Tail-to-Tail baggage, which bypasses the BHS, is excluded from the simulation.

6.5.1 T1: Arrival of the Flight

When a flight arrives at its allocated stand (AIBT), the number of bags on board is extracted from the dataset. The type of baggage—whether bulk or containerized—is also considered. Initially, the time between AIBT and the unloading of the first bag or container was modeled as a uniform distribution between 120 and 180 seconds, based on measurements taken at AAS (Bunnik, 2018).

However, after analyzing the output of the simulation, it became evident that the simulated baggage handling times did not perfectly align with the actual data. To better synchronize the simulated peaks with the actual observed peaks, the time range for T1 was adjusted to a uniform distribution between 1320 and 1380 seconds. This will be further explained in 6.7.

The uniform distribution is used for processes like the time between flight arrival (AIBT) and the unloading of the first bag (T1), as this process was found to have a consistent range of variability with no clear central tendency. The uniform distribution assumes that any time within the specified range is equally likely, which is appropriate for certain random operational delays that don't have a bias toward any particular value.

6.5.2 T2: Unloading of Baggage

Once the first bag or container is unloaded, T2 begins, as shown in Figure 6.1. The time to unload each bag is modeled using a uniform distribution between 6 and 7.5 seconds (Bunnik, 2018). For the process of unloading of containers (T2), a triangular distribution was selected to model a more predictable pattern with a clear mode, representing the most common value observed in the data. This distribution captures the fact that while most unloading times fall around a certain value (e.g., 80 seconds), there are occasional variations within the lower and upper limits (e.g., 60 and 100 seconds) (Bunnik, 2018). This distribution is ideal for modeling operations where there is a most likely value but still some variation due to operational differences.

Carts and Container Loading During the unloading process of bulk baggage, bags are placed in carts. Each cart can hold a maximum of 30 bags in this simulation. As mentioned earlier, AKE containers can hold up to 38 bags, while AKH containers can hold 28 bags. At outstations, the baggage that is packed into containers, is categorized by temperature (e.g., ShoCon, Cold Transfer), and these containers are loaded onto the aircraft in a particular order to ensure efficient handling upon arrival at AAS. If a ShoCon container cannot be fully filled with ShoCon baggage, Cold Transfer bags are "upgraded" to fill the remaining space. This same process applies to Cold Transfer containers being filled with upgraded Super Cold Transfer bags, and also for bulk baggage in carts.

The different baggage temperature categories are unloaded in a specific order: Tail-to-Tail baggage first, followed by ShoCon transfer baggage, then reclaim baggage, followed by Cold Transfer baggage, and finally Super Cold Transfer baggage. To simplify the simulation, odd-sized baggage, which requires special handling, is excluded from the scope.

For each cart or container, the simulation keeps track of the specific bags loaded inside. Since the bags in a single cart or container might be destined for different outbound flights, each cart or container is assigned an "infeed preference." For ShoCon carts or containers, this preference is determined by the bag with the shortest connection time. For Cold Transfer and Super Cold Transfer carts or containers, the most common infeed preference within the cart or container is used (Table 6.5). After this step, each cart or container has a defined infeed point preference.

	Container infeed point preference
ShoCon	Warmest bag in cart/container
Cold Transfer	Most frequent bag in cart/container
Super Cold Transfer	Most frequent bag in cart/container

Table 6.5: Container infeed point preference

In addition to the infeed point preference, each cold and super cold cart or container is assigned a `Latest Time ULD`. This time represents the latest moment a cart or container must arrive at one of the infeed points to ensure that none of the bags miss their outbound flight. The `Latest Time ULD` is equal to the earliest `Latest Time Bag` present in the carts or container. The carts or containers are grouped by temperature behind a tug to form a train, with each train consisting of up to 6 trailers.

6.5.3 T3: Transportation Time to Infeed Points

Once the baggage has been unloaded from the aircraft and loaded onto the carts or containers, the next critical step is transporting the ShoCon baggage directly to the BHS. T3 represents the time it takes for a tug (or baggage train) to travel from the ramp cluster where the aircraft is parked to the designated transfer infeed point. These travel times vary depending on the distance between the ramp cluster and the specific infeed point.

Table 6.6 presents the travel times (in seconds) between different ramp clusters and infeed points. These times were derived from measurements taken at AAS and reflect the average duration for a tug to travel between these locations.

Ramp Cluster	Terminal 2	D	West	South	E
A-pier + B-platform	451	441	549	271	461
B-pier	336	325	433	155	345
C-stem	241	198	338	136	250
C-head	273	189	370	168	282
D-stem	190	93	287	182	200
D-fork south	257	76	354	248	266
D-fork north	262	84	360	254	213
E-stem	124	202	221	207	88
E-head	229	248	326	312	19
F-pier	177	299	191	301	227
G-stem	199	322	136	324	250
G-head	273	395	210	397	324

Table 6.6: Travel times between ramp clusters and infeed points in seconds (Bunnik, 2018)

This table serves as a reference for determining the transportation times for ShoCon baggage trains, ensuring that the simulation accounts for the varying distances between different parts of the airport. The travel time plays a critical role in the overall timing and efficiency of the baggage handling process. In the base simulation scenario, no baggage is buffered. All baggage types, once the baggage is ready for departure, are immediately transported to their infeed point of preference.

6.5.4 T4: Buffering Time

As previously mentioned, no buffering occurs in the base scenarios, but it is introduced in alternative distribution strategies. This research considers only Super Cold Transfer baggage as eligible for buffering; ShoCon and Cold Transfer baggage are not to be buffered. T3 applies to ShoCon and Cold Transfer baggage, while Super Cold Transfer baggage is directed to a buffer after T2 and later transported to an infeed point. T4 is defined as the time interval between T2 and arrival at the infeed point (T5). The specific locations and capacities of these buffers are outside the scope of this research, and it is assumed that the buffer has unlimited capacity for containers. Additionally, it is assumed that baggage is buffered by container, with containers remaining intact without unpacking or re-sorting within the buffer—each container is returned in the same configuration as it entered. Further research could explore the detailed workings of the buffer. For the different distribution strategies discussed later, T4 will be specified for each container or cart.

6.5.5 T5: Arrival at Infeed Point

T5 begins the moment a baggage train arrives at an infeed point. For ShoCon and Cold Transfer baggage, this occurs directly after transportation from the ramp cluster. For Super Cold Transfer baggage, T5 begins once the baggage has been released from the buffer and transported to the infeed point. The timing of T5 is critical, as it determines when the bags can be unloaded onto the infeed quays and processed by the BHS.

In cases where too much baggage is brought to an infeed point, and the processing capacity is exceeded, the most recently arrived baggage is placed in a waiting queue. This introduces a delay in processing, as the system prioritizes the baggage that arrived earlier. If the total baggage volume exceeds the infeed point's capacity, the excess baggage must wait until space becomes available. This delay can have downstream effects, especially for bags with tight connections or those nearing their `Latest Time` window for outbound flights.

In essence, T5 represents the final phase of this simulation of the baggage journey through the infeed system. Ensuring that this step is managed efficiently is crucial for avoiding delays that can cascade through the system, potentially affecting multiple flights. The capacity at each infeed point, combined with effective baggage distribution, determines how smoothly this phase operates.

6.6 Buffer and Autonomous Guided Vehicles

In this simulation, the buffer is conceptualized as a black box with theoretically unlimited storage capacity for baggage containers. Although no strict limits are imposed on buffer space, the simulation tracks buffer occupancy over time. This provides insights into peak demands on the buffer and informs a realistic estimate of required capacity to ensure smooth handling of container flows.

To facilitate the buffer process, AGVs are deployed to perform two primary tasks: transporting containers from the apron to the buffer and moving containers from the buffer to the appropriate transfer infeed points. The simulation dynamically calculates the required number of AGVs based on the demand for these tasks, with specific operational rules designed to optimize vehicle use and minimize total AGV requirements.

As said, each AGV performs one of two actions:

- Action 1: Picking up containers from the apron and delivering them to the buffer
- Action 2: Retrieving containers from the buffer and transporting them to designated transfer infeed points

Some containers bypass the buffer and are instead transported directly to the infeed point. For these containers, the total travel time is always less than 20 minutes; however, the exact timing of arrival at the infeed point depends on each container's designated infeed time. To ensure that the container reaches the infeed point precisely when needed, the AGV pauses strategically along its route, thereby synchronizing its arrival with the required infeed schedule. This "buffered by transport" approach reduces physical buffer occupancy and allows the AGV itself to function as a mobile buffer for containers with tight schedules. Once these containers are delivered, the AGV enters a cooldown period before it can be reassigned.

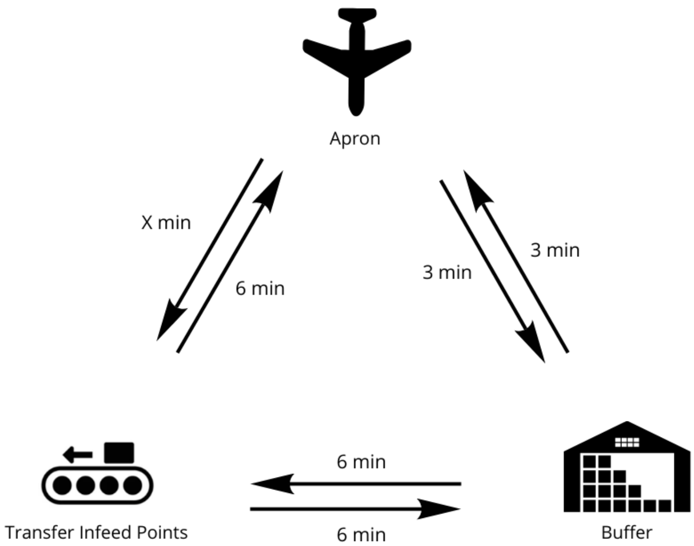


Figure 6.3: AGV Travel Times

The simulation incorporates specific handling and travel times essential for accurately modeling AGV operations. For travel times, transporting containers from the apron to the buffer (Action 1) requires 3 minutes, while moving containers from the buffer to the transfer infeed point (Action 2) takes 6 minutes (Dijkhuizen, 2024). Each container requires 2 minutes to connect or disconnect from the AGV during loading and unloading (Dijkhuizen, 2024). An AGV can form a "train" with up to 6 containers, so fully loading or unloading an AGV at capacity takes 12 minutes. When transporting containers from the buffer to an infeed point, the AGV must carry containers bound for the same destination to ensure operational efficiency.

After completing an action, each AGV undergoes a 6-minute cooldown period. This cooldown accounts not only for realignment and preparation time but also for the journey back to the start location of its next task, whether at the apron or buffer. By incorporating this cooldown, the simulation ensures that AGVs are strategically positioned and ready for new assignments, minimizing downtime and optimizing the workflow.

To further minimize the total number of AGVs required, the simulation prioritizes reusing available vehicles. Before allocating a new AGV, the simulation checks if any existing AGVs are available at the required time. Only when no AGVs are free is a new vehicle assigned. This approach optimizes vehicle availability and ensures that AGV demand is met with the fewest vehicles possible. The simulation dynamically tracks each AGV's availability based on its assigned tasks and cooldown periods, allowing each AGV to re-enter the pool of available vehicles once its cooldown is complete. This system provides a realistic view of AGV requirements and utilization over time, adapting efficiently to fluctuations in container demand and ensuring a responsive and resource-efficient operation.

The simulation generates a comprehensive log of AGV activities, detailing each trip performed. Each entry in the log records the AGV ID, type of action (to buffer or to infeed point), pickup and drop-off locations, start and end times, number of containers transported, and container identifiers. This log could serve as the foundation for a future scheduling system to manage AGVs, although further optimization would be needed to achieve this.

Analyzing the simulation output allows for precise determination of the daily AGV requirements based on peak demand, as well as insights into AGV utilization rates and average trip times. By assessing buffer occupancy and vehicle usage, the simulation supports strategic planning for AGV fleet sizing, scheduling, and capacity allocation, ensuring an efficient response to varying operational demands.

6.7 Simulation Verification and Validation

For the simulation verification and validation, the output is presented in 15-minute intervals, providing a detailed look at the occupancy rate across all infeed points. Each piece of baggage is assigned to a specific 15-minute window, allowing for the aggregation of baggage volumes into these time slots. This approach enables the monitoring of the occupancy rate at each infeed point, helping identify periods of peak activity and potential bottlenecks in the system.

The graphs in Figures 6.4 and 6.5 display the comparison between simulated and actual occupancy data for the average daily values, as well as a specific day (June 30, 2024). The 15-minute interval data is crucial for assessing the overall performance of the baggage system in handling uneven baggage flows and identifying areas for improvement in distribution strategies.

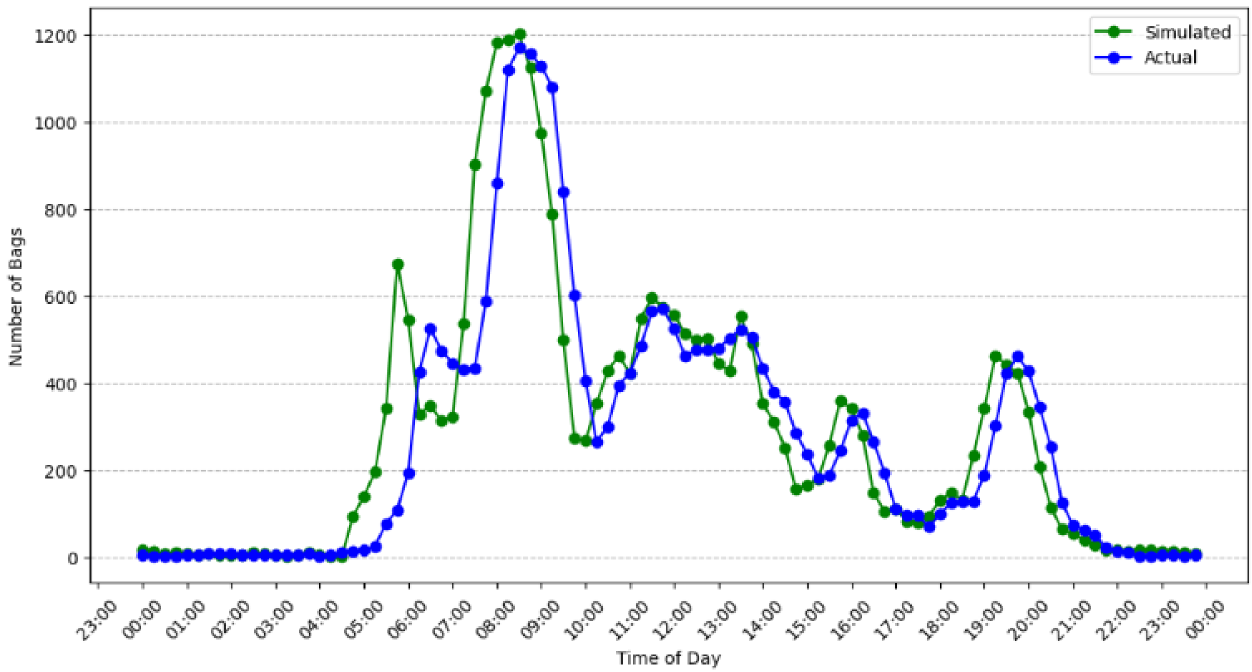


Figure 6.4: Simulated and Actual Average Baggage Occupancy per 15-Minute Interval

In Figure 6.4, the average simulated occupancy values for each 15-minute window are shown alongside the actual historical average occupancy rates for the same period. The simulated values closely match the actual values, though some deviations are observed during peak times. These deviations can be attributed to small differences in processing assumptions or variability in the real-world data that was not fully captured in the simulation model. Despite this, the simulation successfully replicates the general trends and patterns of baggage flow, confirming its validity for testing alternative distribution strategies.

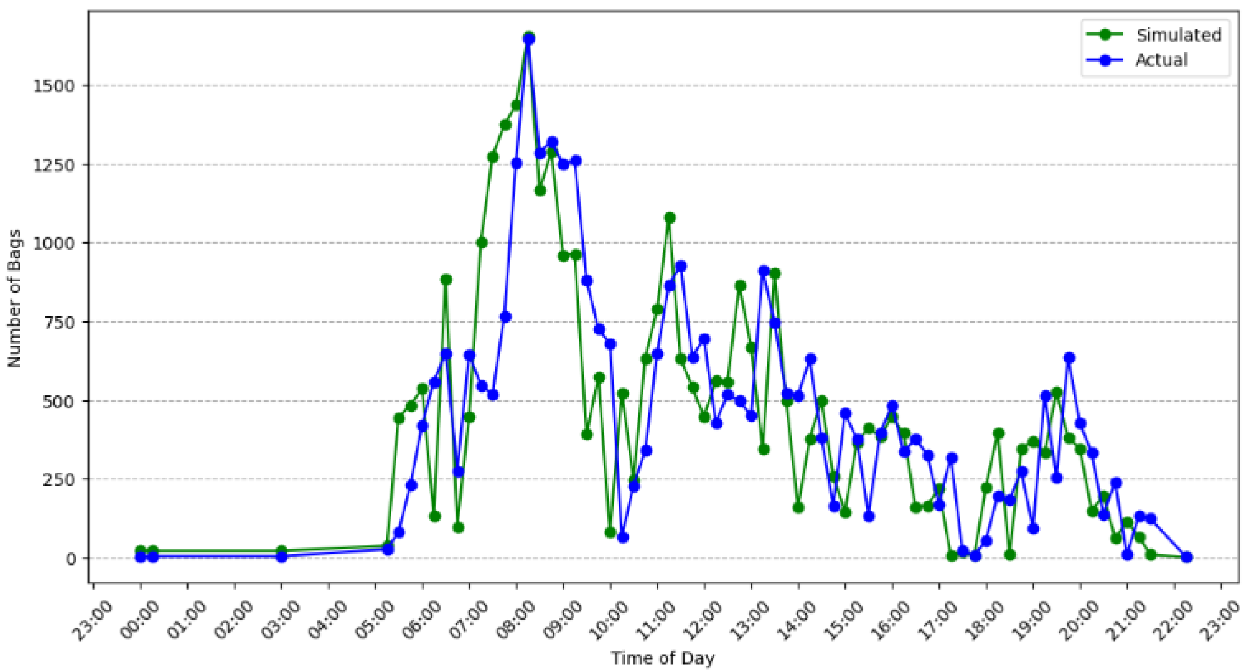


Figure 6.5: Simulated and Actual Baggage Occupancy on 30-06-2024

Figure 6.5 shows the detailed simulation output for a single day: June 30, 2024. This specific day was chosen for analysis due to its heavy baggage volume and operational challenges, and because it is the last day of the dataset.

The graph provides a comparison between the actual and simulated baggage occupancy rates throughout the day.

6.8 Verification

To ensure the accuracy of the simulation model, a series of verification checks were conducted. These checks involved modifying key input parameters and observing whether the resulting changes in the simulation matched the expected outcomes. The purpose of these tests was to verify that the model behaves logically when faced with extreme or altered conditions.

Table 6.7 outlines the three primary verification tests that were performed. Each test involved changing a specific input, such as the time to unload baggage or the travel times for baggage trains. The expected effect was that increasing these times would delay the infeed times of baggage, causing the simulated occupancy graph to shift to the right, representing delayed baggage processing.

In all three tests, the expected effects were observed. For example, multiplying the unloading times by a factor of 100 resulted in a significant delay in baggage processing, as seen by the shift in the graph's peak times to later intervals. Similarly, increasing the travel times and the time required to unload the first bag also led to delayed infeed times, with the simulation output graph showing the anticipated shift to the right. These successful outcomes indicate that the model is functioning as intended under different conditions.

Changed input	Change	Expected effect	Effect	Check
Time to unload	x100	Delayed infeed times	The simulated graph shifts to the right	✓
Unloading times	x100	Delayed infeed times	The simulated graph shifts to the right	✓
Traveltimes	x100	Delayed infeed times	The simulated graph shifts to the right	✓

Table 6.7: Verification checks

The results from these verification checks confirm that the simulation accurately responds to changes in input parameters in a predictable manner. The fact that the simulated infeed times consistently shifted to the right when processing times were increased demonstrates that the model's logic is sound and robust. With these successful verifications, confidence in the model's reliability is strengthened, and it is prepared for more complex scenario testing.

6.9 Validation

Validation is an essential step to ensure that the simulation model accurately represents real-world processes. For this validation, both visual and statistical comparisons between the simulated and actual baggage occupancy levels at the transfer infeed points were conducted. The visual validation is presented through line plots comparing the simulated and actual data, while statistical validation uses key error metrics, such as Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), and Mean Absolute Error (MAE). These metrics provide an objective measure of the model's accuracy.

6.9.1 Visual Validation

The visual validation was performed using line plots that compare the simulated and actual baggage occupancy levels. Two scenarios were examined:

- The averaged daily baggage flow across the entire dataset.
- A specific day, 30-06-2024.

Figures 6.4 and 6.5 (found in section 6.7) depict the original simulation results compared to actual occupancy data. Figure 6.4 shows the comparison of the average daily baggage occupancy over all days in the dataset, while Figure 6.5 displays the results for a specific day, 30-06-2024.

In both cases, the original simulation exhibited a consistent time misalignment, with peaks in the simulated baggage occupancy occurring earlier than the actual observed peaks. After discussions with several simulation experts at AAS, it became clear that such discrepancies are not unusual; similar inexplicable delays have been observed in previously developed simulations.

6.9.2 Statistical Validation

In addition to visual comparisons, statistical metrics were calculated to quantify the difference between the simulated and actual data. These metrics include:

- RMSE: Measures the average magnitude of the error between the simulated and actual values.
- MAPE: Indicates the percentage error between simulated and actual values.
- MAE: Reflects the average absolute difference between simulated and actual values.

Several configurations with varying delays were tested, ranging from 100 seconds to 1700 seconds, and the results are presented in Table 6.8.

	RMSE	**MAPE [%]**	**MAE**
Original simulation	10,682.28	174.44	6079.62
Simulation with 100 seconds delay	10,132.37	170.89	5739.54
Simulation with 300 seconds delay	9,228.26	166.88	5059.01
Simulation with 500 seconds delay	8041.19	161.51	4391.00
Simulation with 700 seconds delay	7185.85	163.17	3,892.74
Simulation with 900 seconds delay	6,416.73	163.60	3,551.44
Simulation with 1100 seconds delay	5,658.91	163.24	3132.06
Simulation with 1300 seconds delay	5,123.59	166.33	3,094.15
Simulation with 1500 seconds delay	4,803.56	171.64	3,184.74
Simulation with 1700 seconds delay	5,094.81	176.85	3,445.02
Simulation with 2000 seconds delay	5,543.99	184.14	3,883.41

Table 6.8: Statistical validation for different delays

The introduction of delays significantly impacted the model's accuracy. Based on the RMSE, MAE, and MAPE, the following insights were drawn:

- The 1500 seconds delay resulted in the lowest RMSE (4,803.56) and a relatively low MAE (3,184.74), making it the best configuration based on overall error reduction.
- Although the 900 seconds delay showed improved accuracy compared to the original simulation, the 1500 seconds delay consistently provided better results across multiple metrics.

6.9.3 Conclusion of Validation

Both the visual and statistical validation results confirm that the simulation model provides a reasonable representation of the actual baggage handling processes at AAS, particularly after introducing a delay. The 1500 seconds delay offers the best overall performance based on RMSE and MAE, making it the most accurate choice for future simulations. This adjustment allows for a more accurate reflection of the operational dynamics observed during the busiest times at the airport. While this modification may not be directly explained by empirical data, it serves to ensure that the simulation outcomes more closely mirror real-world baggage flow patterns, enhancing the overall validity of the model. For further testing and scenario analysis, the 1500 seconds delay will be applied due to its optimal balance across error metrics.

7 Time Series Forecasting

This chapter provides answers to subquestion 3: *What is needed to optimize the distribution of cold transfer baggage?* A prediction model is essential to identify more favorable times to introduce baggage to the BHS, helping to manage occupancy levels efficiently and avoid unnecessary peaks. By forecasting occupancy, we can strategically time the release of buffered baggage to moments when the system load is lower, optimizing the flow through the BHS.

To gain a better understanding of the occupancy levels of the baggage system over time, two Time Series Forecasting models were developed. These models are based on 10 weeks of historical baggage data and aim to predict occupancy levels, which can then inform distribution strategies to effectively manage peak loads in the BHS.

7.1 Literature on Time Series Forecasting Models

There are multiple models that can be applied for Time Series Forecasting. SARIMA (Seasonal Autoregressive Integrated Moving Average) is well-regarded for its ability to capture regular seasonal patterns effectively. One study ranks SARIMA as the top choice for short-term seasonal forecasting, emphasizing its simplicity and accuracy in handling datasets with a consistent seasonal structure, such as daily fluctuations in the baggage system (Naim et al., 2020). The model excels in breaking down the time series into distinct components: autoregressive, differencing, and moving average. This decomposition makes SARIMA particularly suitable for applications where one dominant seasonal period is present, such as the daily cycle in the baggage data (Shchelkalin, 2014). Given these characteristics, SARIMA was selected as it is well-suited for time series data with a single, consistent seasonal pattern. Its simplicity and effectiveness in capturing regular seasonal trends make it an appropriate choice for modeling daily variations in baggage flows at the airport.

On the other hand, TBATS (Trigonometric, Box-Cox transformation, ARMA errors, Trend and Seasonal components) provides greater flexibility, especially when dealing with multiple overlapping seasonal patterns, such as daily and weekly cycles. A comparative study of SARIMA and TBATS revealed that TBATS excels in handling complex seasonalities, outperforming SARIMA in scenarios where multi-seasonal phenomena occur, such as ATM withdrawal forecasting and tracking disease incidence patterns (Gurgul et al., 2023; Yu et al., 2021). This flexibility makes TBATS the ideal model for systems where weekly patterns interact with daily cycles, as observed in the baggage data. TBATS was therefore looked at for its ability to model multiple seasonalities, including both daily and weekly patterns. This makes it particularly well-suited for forecasting in environments where system occupancy fluctuates across different time scales, as in airport baggage handling systems, where the interaction between weekly peaks and daily fluctuations is critical to ensuring efficient operations.

The goal of this chapter is to evaluate both methods and determine which model provides the most accurate and reliable predictions for occupancy levels at the infeed points. By comparing the performance of SARIMA and TBATS, the aim is to identify the best approach to support future distribution strategies in the BHS.

7.2 Time Series Forecasting with SARIMA

SARIMA is a statistical model used for time series forecasting, especially when the data exhibits seasonal patterns. SARIMA extends the ARIMA model by adding a seasonal component, making it suitable for datasets with strong seasonal cycles. In this research, SARIMA was initially chosen because the baggage data exhibits a clear daily pattern, with repeated peaks every 24 hours (96 quarters).

The components of a SARIMA model are:

- **AR (Autoregressive) component**: This models the relationship between a current observation and a number of lagged observations (previous time steps). A higher AR value allows the model to incorporate a larger history of past values to predict the next one.
- **I (Integrated) component**: This removes the trend from the time series by applying one or more differencing steps. The differencing step helps make the data stationary, which is essential for accurate forecasting.
- **MA (Moving Average) component**: This models the relationship between a current observation and a number of lagged error terms. This allows the model to correct for past forecasting errors.

- **Seasonal components (SAR, SI, SMA)**: These extend the AR, I, and MA components to capture seasonal patterns over time, where SAR is the Seasonal Autoregressive term, SI is the Seasonal Differencing term, and SMA is the Seasonal Moving Average term.
- **s (Seasonal Period)**: This refers to the number of time steps in one seasonal cycle. For this data, $s = 96$, as the data is collected in 15-minute intervals and there are 96 quarters in a 24-hour period.

The SARIMA model is particularly useful for this research because the baggage data exhibits a strong daily pattern. However, one limitation of SARIMA is that it can only capture one seasonal period at a time, in this case, the daily pattern (96 quarters in a day).

7.2.1 Model Construction and Parameter Selection for SARIMA

The construction of the SARIMA model began with preparing and analyzing the dataset, followed by splitting the data into training and test sets. A grid search was used to determine the optimal parameters for the SARIMA model.

The SARIMA model with the best parameters yielded the following:

- **p (Autoregressive term)**: 1
- **d (Differencing term)**: 1
- **q (Moving Average term)**: 1
- **P (Seasonal Autoregressive term)**: 1
- **D (Seasonal Differencing term)**: 1
- **Q (Seasonal Moving Average term)**: 1
- **s (Seasonal Period)**: 96 (quarters in a day)

These parameters were chosen based on the RMSE on the training data. The SARIMA model yielded an RMSE of 144.37, which suggests that the model effectively captured the daily patterns in the baggage data.

7.2.2 Validation of SARIMA

Before diving into the results, it's important to understand the metrics used for validation:

- **RMSE**: This metric measures the average magnitude of the error between predicted and actual values. It is particularly sensitive to large errors because it squares the differences before averaging them. A lower RMSE indicates a better fit to the actual data.
- **SMAPE (Symmetric Mean Absolute Percentage Error)**: This is a variation of the MAPE, which measures the percentage difference between the forecasted and actual values. SMAPE is symmetric and more robust to zero or near-zero values, making it suitable when small or zero values are present in the dataset, which is the case.
- **MAE**: MAE represents the average absolute difference between the predicted and actual values. Unlike RMSE, it does not square the errors, making it less sensitive to large outliers.

The SARIMA model was validated using the forecasted occupancy levels for 30-06-2024, compared with the actual test data (15-04-2024 to 29-06-2024). The metrics in table 7.1 were used to assess model accuracy.

	RMSE	**SMAPE [%]**	**MAE**
SARIMA	144.37	75.39	102.63

Table 7.1: Validation Metrics SARIMA

These metrics indicate that SARIMA performs reasonably well for forecasting baggage occupancy levels, but its limitation lies in its inability to capture weekly patterns. Figure 7.1 shows the comparison between the training data, test data, and SARIMA predictions for 30-06-2024.

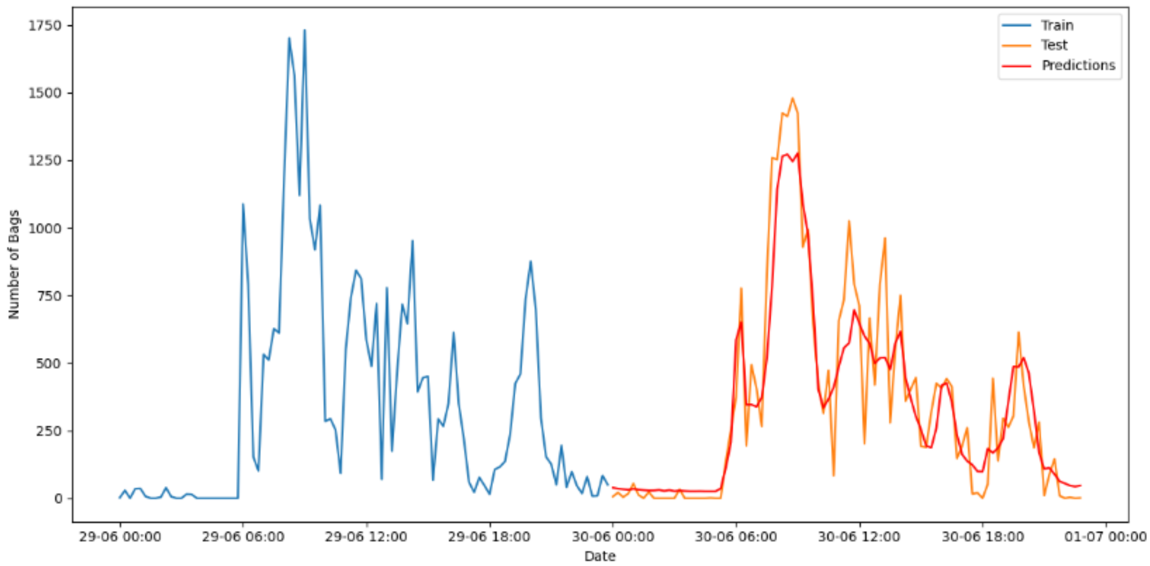


Figure 7.1: Train, Test, and Predictions using SARIMA Model for 30-06-2024

7.3 Time Series Forecasting with TBATS

TBATS is a more flexible time series forecasting model that can handle complex seasonal patterns, including both daily and weekly cycles. Unlike SARIMA, TBATS can model multiple seasonal periods, making it particularly useful for datasets like the baggage data, where occupancy levels fluctuate on both daily and weekly bases.

The components of TBATS include:

- **Trigonometric terms**: These capture complex seasonal patterns, such as weekly and daily cycles.
- **Box-Cox transformation**: This stabilizes variance in the data.
- **ARMA errors**: Autoregressive Moving Average terms model residual correlations.
- **Trend component**: This captures any long-term trends in the data.
- **Seasonal components**: TBATS can model multiple seasonalities, such as both daily and weekly patterns.

TBATS was chosen for its ability to model both daily (96 quarters) and weekly (672 quarters) patterns in the baggage data.

7.3.1 Model Construction and Parameter Selection for TBATS

The TBATS model was trained on the same training dataset as SARIMA and used the following seasonal periods:

- **Daily seasonality**: 96 quarters (24 hours)
- **Weekly seasonality**: 672 quarters (7 days)

The model automatically selects the best parameters, including the Box-Cox transformation and ARMA errors. The RMSE for TBATS was calculated as 151.02, which is slightly higher than SARIMA, but TBATS captured both daily and weekly seasonal patterns.

7.3.2 Validation of TBATS

Before diving into the results, the same metrics as used for SARIMA validation will be employed:

- **RMSE (Root Mean Squared Error)**: This measures the average error magnitude, with an emphasis on large errors.
- **SMAPE (Symmetric Mean Absolute Percentage Error)**: SMAPE is robust to zero values and calculates the percentage difference between forecasted and actual values.

- **MAE (Mean Absolute Error)**: MAE measures the average absolute error, giving a direct interpretation of how much the predictions differ from actual values on average.

TBATS was validated using forecasted occupancy levels for 30-06-2024. The following metrics were used to assess the model's accuracy:

The metrics in table 7.1 were used to assess model accuracy.

	RMSE	SMAPE [%]	MAE
TBATS	151.02	78.73	115.71

Table 7.2: Validation Metrics TBATS

TBATS produced slightly higher errors compared to SARIMA, but its ability to model weekly seasonality gives it an advantage. Figure 7.2 shows the comparison between the training data, actual test data, and TBATS predictions for 30-06-2024.

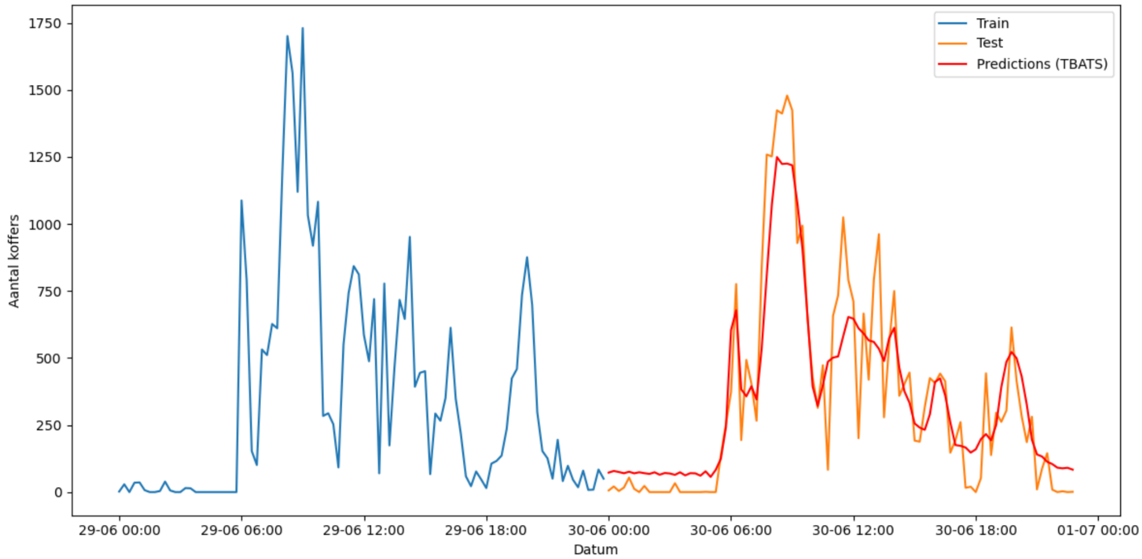


Figure 7.2: Train, Test, and Predictions using TBATS Model for 30-06-2024

7.4 Conclusion and Model Comparison

Both SARIMA and TBATS provide valuable insights into the seasonal patterns present in the baggage occupancy data. SARIMA performs well for daily forecasting, with an RMSE of 144.37 and MAE of 102.63. However, it is limited by its inability to capture weekly patterns.

TBATS is slightly less accurate in terms of RMSE (151.02) and MAE (115.71), but its ability to model both daily and weekly cycles makes it more suitable for long-term forecasts. Additionally, TBATS has greater potential for more accurate predictions with a larger dataset. Given the importance of capturing weekly patterns in baggage handling processes, TBATS will be used for further analysis and for developing distribution strategies to manage peak loads in the BHS.

8 Distribution Strategies and Scenario's

In this chapter, we explore different strategies for the buffering of super cold transfer baggage at AAS. The goal is to minimize peak occupancy at the infeed points of the BHS, ensuring smoother operations and reducing workload imbalances. This chapter sets up the answers to subquestion 4: *Which distribution strategies are suitable for buffering cold transfer baggage, and how do these strategies impact peak shaving under different scenarios?*

Two primary tasks are addressed: determining when to buffer baggage and deciding when it is appropriate to reintroduce it into the system. Both steps—buffering and reintroduction—are essential for effectively managing peak loads and ensuring the BHS operates efficiently.

8.1 Strategic Timing for Baggage Buffering

A crucial part of distribution management is deciding when to buffer super cold baggage. Er wordt in dit onderzoek alleen gekeken naar de mogelijk First, the literature is reviewed. Thene, three strategy options for this part of the distribution are chosen.

8.1.1 Literature on the Timing for Buffering

The use of a fixed occupancy target as a buffering strategy is a simple yet robust approach. This strategy is often effective in environments with predictable peak loads and can be implemented without the need for complex forecasting. Literature shows that this method is suitable for systems with limited variability, as it helps prevent system overloads. In logistical environments, such as baggage handling systems, the simplicity of fixed occupancy thresholds makes management more straightforward and execution more reliable (Darwazeh et al., 2022). The robustness of this method lies in the consistency of the rules, leading to a more efficient process without requiring intensive computational models.

However, studies indicate that fixed targets are less effective in dynamic environments with high fluctuations, such as airport systems. Setting a static threshold may result in suboptimal performance during unpredictable peak moments, leading to system overloads or unnecessary buffering (Van Kampen et al., 2010).

Dynamic models, such as those based on polynomial functions, offer the flexibility to better adapt to varying system demands. In the literature, such methods are praised for their ability to account for fluctuations in demand, especially in systems with varying seasonality, such as the aviation sector. The use of advanced forecasting techniques aids in refining processes like peak load management and buffering (Syntetos et al., 2009).

In time series forecasting, finding the right balance between model complexity and overfitting is essential. Literature suggests that higher-order polynomials, such as degree 9, provide flexibility without overfitting to noise in the data, leading to more stable forecasts that are effective in managing peak loads (Fildes et al., 2006).

8.1.2 Option 1: Fixed Target

As discussed in earlier chapters, uneven inflow of transfer baggage can create pressure on other parts of the BHS and lead to imbalanced workloads for the ground staff. Ideally, the occupancy levels at the infeed points would be stable throughout the day. By setting a fixed target occupancy, we can buffer super cold baggage whenever the occupancy exceeds this threshold.

For instance, a target value of 600 bags per 15-minute interval can be used as the threshold for buffering (Figure 8.1). When the occupancy level surpasses this value, baggage is temporarily held back. This approach is straightforward and easy to implement, offering a clear rule for when to buffer baggage.

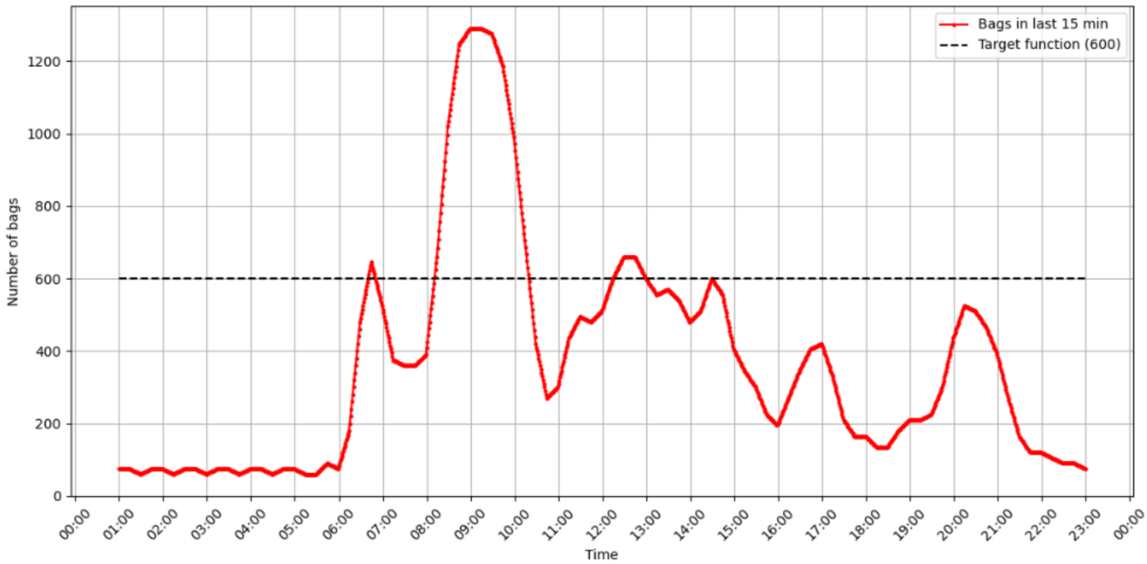


Figure 8.1: Target Value on Time Series Forecasting TBATS for 30-06-2024

While this method offers simplicity, it does not account for the natural variability in baggage flows throughout the day. Peaks in baggage arrival, driven by the flight schedule, make it difficult to maintain a constant occupancy target. This can result in unnecessary buffering during high-demand periods, where some variability is expected.

8.1.3 Option 2: Polynomial Function

A more adaptive method is to use a polynomial function derived from time series forecasting to determine when to buffer baggage. This approach acknowledges that occupancy levels will naturally fluctuate throughout the day based on flight schedules. Instead of setting a rigid target, a smoothed curve—based on a polynomial function—can act as a dynamic occupancy target, allowing for more realistic and flexible buffering.

In this strategy, a polynomial function of degree 9 is applied to the predicted occupancy levels generated by the TBATS model (Figure 8.2). A polynomial function of degree 9 is chosen because it provides a balance between flexibility and overfitting when modeling the overall trend in baggage occupancy. With a lower degree, the function may be too rigid to capture the nuances in the fluctuating baggage flow, missing important peaks and troughs. However, a degree higher than 9 could lead to overfitting, where the polynomial starts modeling random noise in the data rather than the underlying trend. Degree 9 offers enough complexity to smooth out short-term fluctuations while maintaining a realistic long-term pattern that can guide buffering decisions effectively. To prevent the polynomial from dropping too low during times of low predicted occupancy, a lower boundary of 150 is introduced: if the polynomial value falls below 150, it is capped at this minimum level. This constraint helps maintain a baseline occupancy target, ensuring that buffering decisions remain consistent even in low-demand periods. This polynomial curve represents an optimal level of occupancy, smoothing out peaks and filling in troughs. When the actual occupancy exceeds this smoothed target, super cold baggage is buffered. This approach allows for more sophisticated peak shaving and helps avoid unnecessary buffering during natural peaks in baggage arrival.

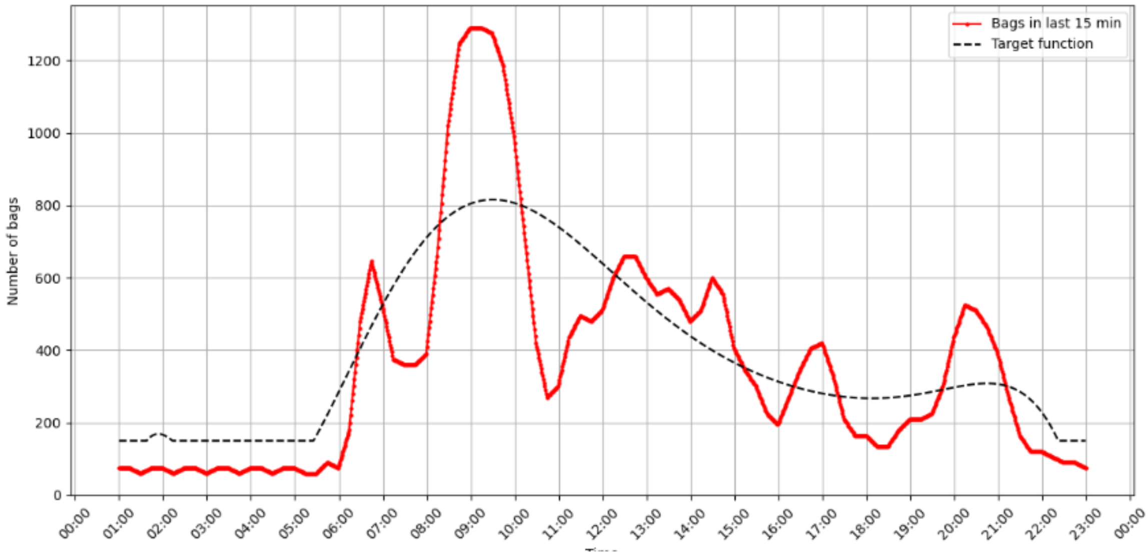


Figure 8.2: Polynomial Function on Time Series Forecasting TBATS for 30-06-2024

The polynomial-based approach is more flexible and realistic compared to the fixed target strategy. It adapts to the natural ebbs and flows in baggage arrival, ensuring that buffering occurs when the system is genuinely under pressure, rather than whenever it crosses an arbitrary threshold. This method is expected to support peak shaving more realistically, smoothing out operational pressures on the BHS.

8.1.4 Option 3: Combination Fixed and Polynomial Function

To further enhance the buffering strategy, a combination of the fixed target and the polynomial function approach is proposed. This hybrid method leverages the simplicity of the fixed target strategy while incorporating the flexibility of the polynomial function to account for natural fluctuations. The fixed target serves as a baseline occupancy level, while the polynomial function dynamically adjusts this target based on forecasted baggage arrival patterns.

In this combined strategy, the same polynomial function of degree 9 is used to provide a smoothed curve, which is then adjusted with a fixed baseline occupancy target (Figure 8.3). This combination allows for an adaptive threshold that remains stable during low-variability periods but flexibly increases when higher baggage flows are expected. This approach optimizes the balance between stability and adaptability, reducing the chances of system overload while minimizing unnecessary buffering.

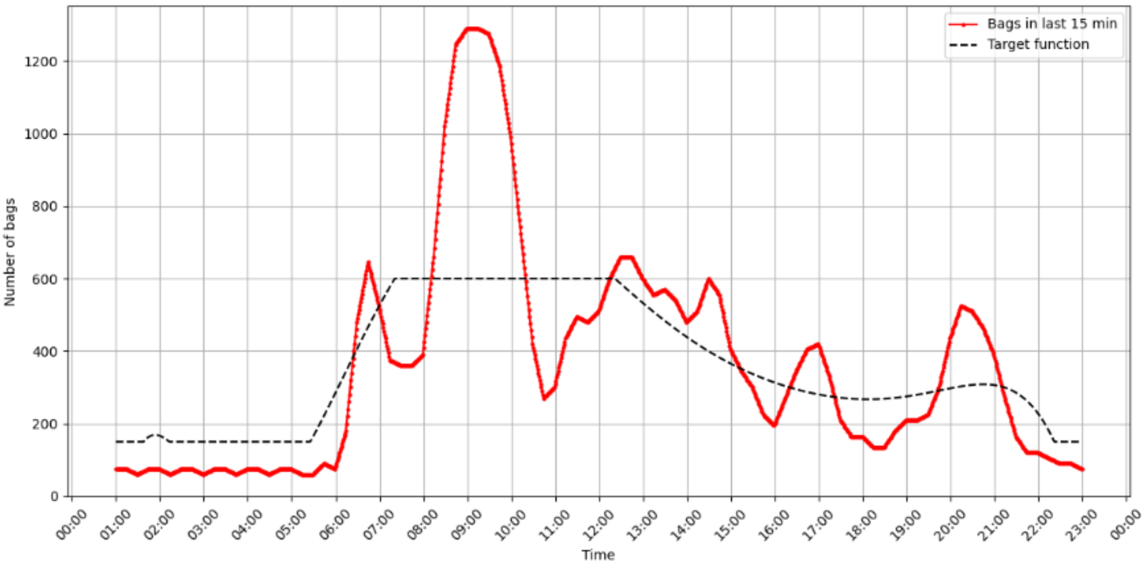


Figure 8.3: Target Value on Time Series Forecasting TBATS for 30-06-2024 with Fixed and Polynomial Function

8.2 Strategic Release of Buffered Baggage

The timing of reintroducing buffered baggage plays a crucial role in determining the effectiveness of peak shaving. First, relevant literature on this topic is reviewed, followed by the design of two distinct alternatives for releasing buffered baggage.

8.2.1 Literature on the Timing of Returning from a Buffer

Forecasting based on future occupancy levels offers a methodology for proactively managing infeed points, preventing peaks. This approach is widely used in inventory management and logistics systems, where precise timing is crucial to avoid system overloads (Disney et al., 2004). The literature also emphasizes the importance of real-time monitoring of buffered items. In dynamic environments, such as baggage handling systems, continuous control over the scheduling and status of reintroductions is essential to prevent new peaks from arising due to poorly timed re-feeding into the system (Monczka et al., 2002).

Strategic use of system slack by reintroducing baggage when occupancy is at its lowest maximizes system capacity. In logistics literature, this optimized reintroduction process is often cited as a method for further minimizing peak loads by making optimal use of available buffers within the system (Silver et al., 2016). Capacity optimization strategies that focus on reintroducing buffered items during periods of low system occupancy significantly enhance the efficiency of logistical processes. These methods help avoid peak loads and reduce critical bottlenecks, such as infeed points in baggage handling systems. Research demonstrates that dynamic capacity optimization improves system performance in variable environments, especially where demand is unpredictable (Wang et al., 2020).

After super cold baggage has been buffered, it must eventually be reintroduced into the system to ensure it reaches its connecting flight on time. Each baggage container has a specific `Latest Time Train`, which represents the latest possible time the baggage can be fed back into the BHS to meet the flight's schedule. The decision of when to reintroduce buffered baggage must strike a balance between avoiding overloading the infeed points and ensuring that no baggage misses its connection. If the infeed points are busy and a buffered container needs to be reintroduced to prevent missed connections, this baggage must be given priority at the infeed point. Two strategies for reintroducing buffered baggage are explored in this section.

8.2.2 Alternative 1: Early Release

The first strategy uses the TBATS Time Series Forecasting model to determine when buffered baggage should be reintroduced. In the simulation (or later in real-time), when it is decided that the baggage is to be buffered, the forecasting model is run to identify the next available time when predicted occupancy drops below the target value—either the fixed threshold or the polynomial curve. The baggage is then scheduled for reintroduction at the first time this happens.

For example, if at 09:00 (green line) it is decided that a suitcase should be buffered, the Time Series Forecasting model will be consulted to find the first time in the available interval (between green and blue line) when occupancy drops below the target, say at 10:15 (purple dot). The suitcase will be scheduled for reintroduction at this time. If there is no moment within the interval between the green and blue lines when the forecast is lower than the target value, the container is assigned to the minute with the lowest prediction within the interval. In the following interval (e.g., 09:15), when another container is buffered, the forecasting model will be run again, with the 09:00 baggage already accounted for in the new forecast. It is crucial to continuously monitor which buffered baggage is scheduled for reintroduction at specific times, ensuring this is updated every minute. This ensures that reintroducing multiple pieces of baggage doesn't create a new peak in occupancy.

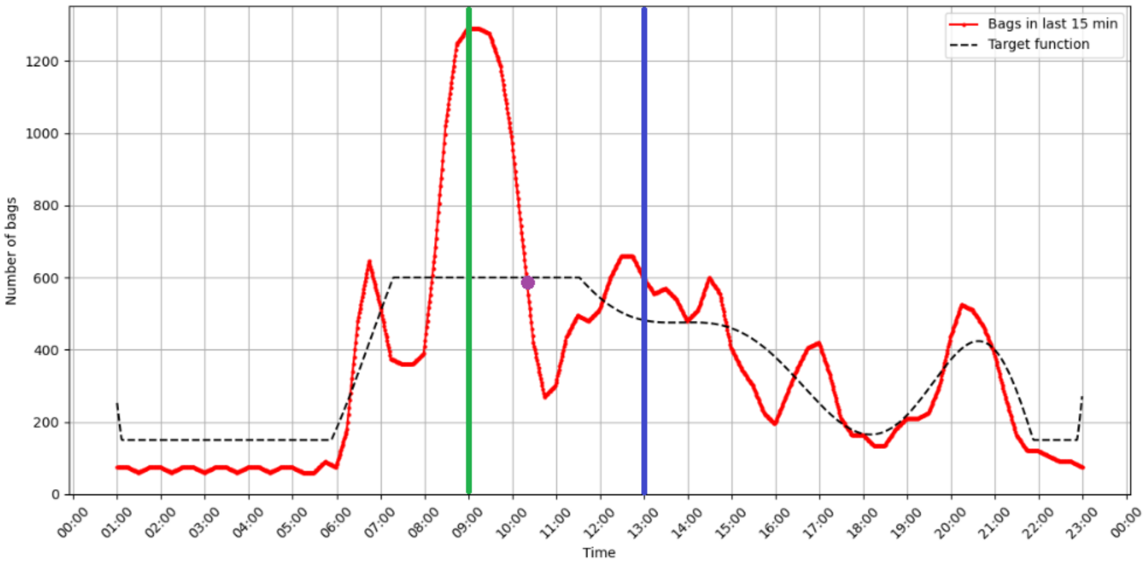


Figure 8.4: Early Reintroduction Strategy

The theory is that this proactive method allows for the smooth reintroduction of baggage, balancing the system load and ensuring that all baggage meets its flight deadlines without overwhelming the system.

8.2.3 Alternative 2: Optimized Release

In this strategy, rather than reintroducing the baggage as soon as the occupancy level falls below the target value, the baggage is held and only reintroduced at the point where the predicted occupancy has the maximum gap below the target value. This approach takes a broader view of the available reintroduction windows. The buffered baggage is only reintroduced when the occupancy levels are at their lowest, within the allowable time-frame based on the `Latest Time Train`.

For instance, when the forecasting model is run at 09:00 for a buffered container, it will look for the time where the gap between the predicted occupancy and the target value is the largest. If this maximum gap occurs at 10:45, the baggage is scheduled for reintroduction at that time, even if the occupancy drops below the target at an earlier moment like 10:15. This maximizes the opportunity to reintroduce baggage without creating a new peak in occupancy.

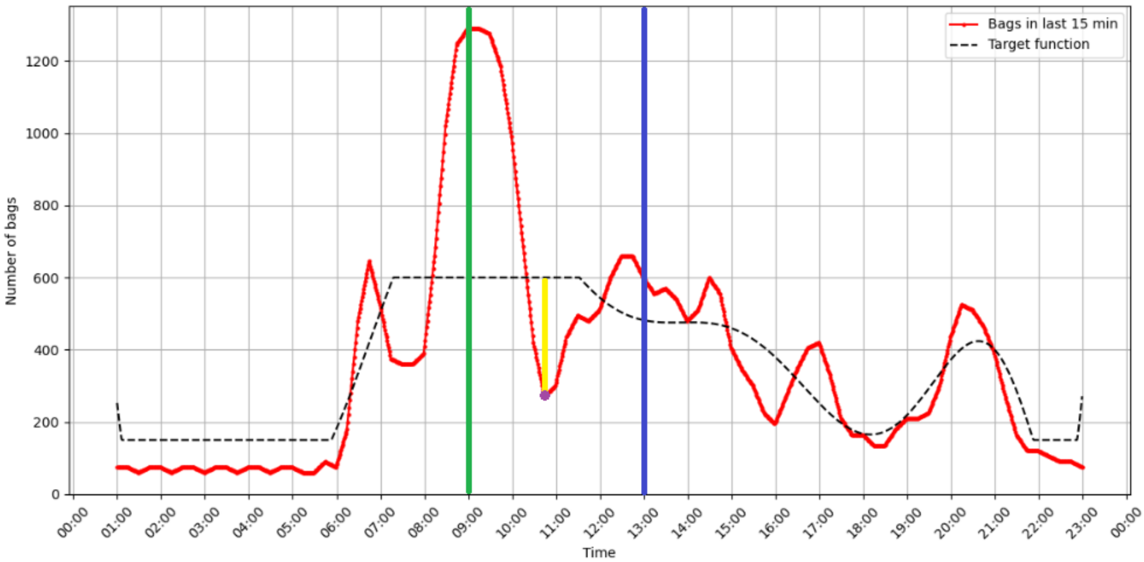


Figure 8.5: Optimized Reintroduction Strategy

This method aims to better optimize the system's capacity by selecting the most opportune times to reintroduce baggage, providing more flexibility and reducing the likelihood of overloading the system at infeed points.

8.3 Scenarios

In this section, we will explore various scenarios that combine the three Strategic Timing Options for Baggage Buffering and the two Strategic Releasing Alternatives for Baggage Buffering. These strategies will be tested across different operational conditions to assess their performance under varying degrees of system stress and external constraints. The goal is to evaluate how robust these strategies are in maintaining optimal occupancy levels and avoiding system overloads in the BHS.

8.4 Scenario 1: Normal Situation

In this scenario, the BHS operates under standard conditions with all infeed points and halls fully functional. This represents the baseline operational environment where the inflow of baggage follows the expected patterns based on historical data. The buffering and reintroduction strategies will be tested in this controlled setting to establish a benchmark for their performance under optimal conditions. The objective is to maintain a balanced and steady occupancy level at the infeed points throughout the day, ensuring efficient operation without exceeding capacity limits.

8.4.1 Scenario 2: Reduced Capacity due to Hall Unavailability

As previously discussed, this study considers four transfer infeed points used at AAS. The distribution of containers across these four transfer infeed points is shown in Table 8.1. In this scenario, baggage hall Z is rendered unavailable for use due to operational constraints such as maintenance work or unforeseen breakdowns. This situation simulates a real-world event where the BHS's capacity is significantly reduced, placing extra pressure on the remaining halls. All baggage that would normally be routed to infeed point Z (8.8%) will now be redirected to one of the other infeed points. This redistribution increases the load on the remaining infeed points, requiring the system to manage its resources efficiently to avoid bottlenecks. Buffering and reintroduction strategies will be critical in this scenario to ensure that the loss of one hall does not result in excessive peaks in occupancy at the remaining infeed points.

Full dataset		
Infeed point	**# of bags**	**Percentage [%]**
D	589,339	28.5
E	1,253,354	60.6
W	24,058	1.2
Z	201,021	9.7
Total	2,067,772	100

30-06-2024		
Infeed point	**# of bags**	**Percentage [%]**
D	9,341	28.9
E	19,522	60.4
W	618	1.9
Z	2,833	8.8
Total	32,314	100

Table 8.1: Distribution of the number of bags across the infeed points

8.4.2 Scenario 3: Increased Baggage Inflow (+20%)

In this third scenario, the inflow of baggage into the system is increased by 20% (Table 8.2). This simulates a situation where flight schedules are unusually dense, or where unforeseen events, such as flight delays or seasonal surges, lead to a higher-than-expected volume of baggage arriving at the airport. This scenario tests the system's resilience in managing increased pressure on the infeed points and BHS as a whole. The performance of different buffering and reintroduction strategies will be examined to determine whether they can still maintain balanced occupancy levels under heightened demand.

30-06-2024		
Infeed point	**# of bags**	**# of bags + 20%**
D	9.341	11,209
E	19.522	23,426
W	618	742
Z	2833	3,400
Total	32.314	38,777

Table 8.2: Distribution of the number of bags across the infeed points + 20 percent

8.5 Conclusion

By combining the three Strategic Timing Options for Baggage Buffering with the two Strategic Releasing Alternatives, we create 6 unique strategy combinations. These strategies are tested across the three scenarios, resulting in 18 distinct configurations. Additionally, baseline scenarios without buffering are included for comparison to assess the effectiveness of the buffering strategies.

Configuration	Option	Alternative	Scenario
Base 1	No buffer	No buffer	1: Normal
Base 2	No buffer	No buffer	2: Reduced capacity
Base 3	No buffer	No buffer	3: Increased inflow
1	1: Fixed target	1: Early release	1: Normal
2	1: Fixed target	1: Early release	2: Reduced capacity
3	1: Fixed target	1: Early release	3: Increased inflow
4	1: Fixed target	2: Optimized release	1: Normal
5	1: Fixed target	2: Optimized release	2: Reduced capacity
6	1: Fixed target	2: Optimized release	3: Increased inflow
7	2: Polynomial target	1: Early release	1: Normal
8	2: Polynomial target	1: Early release	2: Reduced capacity
9	2: Polynomial target	1: Early release	3: Increased inflow
10	2: Polynomial target	2: Optimized release	1: Normal
11	2: Polynomial target	2: Optimized release	2: Reduced capacity
12	2: Polynomial target	2: Optimized release	3: Increased inflow
13	3: Combined target	1: Early release	1: Normal
14	3: Combined target	1: Early release	2: Reduced capacity
15	3: Combined target	1: Early release	3: Increased inflow
16	3: Combined target	2: Optimized release	1: Normal
17	3: Combined target	2: Optimized release	2: Reduced capacity
18	3: Combined target	2: Optimized release	3: Increased inflow

Table 8.3: Configuration Options and Scenarios

The scenarios simulate the impact of varying external conditions on the strategies, providing insights into their effectiveness and robustness. The results will be analyzed and compared in the next chapter to evaluate their relative strengths, weaknesses, and overall performance under different conditions.

9 Results

This section presents the outcomes of the simulated strategies for buffering super cold transfer baggage on a single day (30-06-2024), with the objective of achieving peak shaving at the transfer infeed points. Table 9.1 summarizes the results across different configurations, enabling comparison of the strategic options and their performance across various scenarios. First, the KPIs used to assess these results will be discussed. Secondly, the configurations will be analyzed by comparing the three different options within each scenario. Thirdly, the configurations will be examined to evaluate the performance of the two alternatives relative to each other. At last, a comparison of the results of all configurations is given.

Configuration	Peak	STD	MAC	CoV	Peak in buffer	Buffered containers	# of AGVs
Base 1	1958	404.83	28.82	1.20	0	0	0
Base 2	1958	404.15	29.4	1.0	0	0	0
Base 3	2283	485.38	33.88	1.20	0	0	0
1	1462	349.90	26.25	1.04	113	199	27
2	1454	348.81	26.99	1.04	114	205	26
3	1734	411.03	29.64	1.02	138	288	34
4	1454	347.95	26.94	1.03	116	199	28
5	1464	350.19	27.23	1.04	117	205	28
6	1736	414.84	30.49	1.03	142	288	35
7	1492	346.89	25.82	1.03	104	257	33
8	1484	345.25	26.57	1.03	105	259	32
9	1819	413.07	30.15	1.02	127	311	40
10	1433	350.48	26.55	1.04	118	251	31
11	1459	349.25	26.77	1.04	115	251	31
12	1663	417.77	30.98	1.03	139	300	39
13	1492	346.89	25.82	1.03	104	257	33
14	1484	345.25	26.57	1.03	105	259	32
15	1819	413.07	30.15	1.02	127	311	40
16	1433	350.48	26.55	1.04	118	251	31
17	1459	349.25	26.77	1.04	115	251	31
18	1663	417.77	30.98	1.03	139	300	39

Table 9.1: Full results of all configurations

9.1 Key Performance Indicators

To evaluate the effectiveness of the different strategies, several KPIs were used. The primary focus is on the performance of the strategies in directly influencing the occupancy levels at the transfer infeed points.

- **Peak**: The maximum observed occupancy at the infeed points during the simulated day. Lower peak values indicate better peak shaving performance.
- **STD (Standard Deviation)**: The standard deviation of occupancy levels over the day, which reflects the variability in the occupancy. A lower standard deviation suggests a more consistent flow of baggage.
- **MAC (Mean Absolute Change)**: The average absolute change in occupancy levels between consecutive minutes, indicating the stability of the occupancy. Smaller values suggest smoother transitions.
- **CoV (Coefficient of Variation)**: The coefficient of variation (standard deviation divided by mean) of occupancy, providing a normalized measure of variability relative to the mean occupancy.

Additionally, other operational aspects relevant to airside logistics were considered. These include the peak number of containers held in the buffer at any given time and an estimate of the number of AGVs required to support the buffering process.

- **Peak in Buffer**: The maximum number of containers held in the buffer at any given moment. This metric shows the extent of buffering required to achieve the desired occupancy levels.
- **Buffered Containers**: The total number of containers buffered throughout the day. Higher values reflect greater reliance on the buffer to reduce peaks at the infeed points.

- **# of AGVs**: The number of automated vehicles needed to transport containers between the apron, buffer, and infeed points. This reflects the logistical demand of each configuration.

9.2 Analysis of Configurations

The results illustrate the impact of various buffering strategies and scenarios on both occupancy levels and resource requirements. Strategic Options and Alternatives are compared within each scenario. All results can be found in Appendix 13. One remark before diving deeper into the results, Option Polynomial and Option Combination show the same values.

9.2.1 Strategic Options

In this section, configurations are analyzed with a focus on the 'Option' variable as the only differing factor. By examining each of the 'Options' under both 'Alternatives' and across all 'Scenarios', we can assess the specific impact of each 'Option' strategy.

Early Release Alternative
Firstly, we fix the Early Release alternative. By comparing configurations 1, 7, 13 and 2, 8, 14, and 3, 9, 15, we can observe the effects. Table B.2 presents the results here, but in Appendix 13, Tables B.2, B.3, and B.4 are displayed together to give a clear overview of these configuration ordered on 'Peak in buffer' from small to large.

We observe that the Fixed option, with the Fixed Early Release alternative, yields the most significant peak shaving, achieving a 25.3% reduction in peak occupancy. The Polynomial and the Combination achieve a reduction of around 23.8%. The Polynomial and Combination options exhibit slightly more favorable values for standard deviation, MAC, and CoV, indicating a smoother and more stable occupancy profile, even though the difference is not significant. The tables clearly display these effects, with all tables sorted by 'Peak of bags' from smallest to largest.

Conf.	Option	Alternative	Scenario	Peak	STD	MAC	CoV	Peak buff	Buff. containers	AGVs
1	1: Fixed	1: Early release	1: Normal	1462	349.90	26.25	1.04	113	199	27
7	2: Polynomial	1: Early release	1: Normal	1492	346.89	25.82	1.03	104	257	33
13	3: Combination	1: Early release	1: Normal	1492	346.89	25.82	1.03	104	257	33
Base 1	0. No buffer	0. No buffer	1: Normal	1958	404.83	28.82	1.2	0	0	0

Table 9.2: Results on Strategic Options with Alternative 1 and Scenario 1

The Fixed option results in a slightly higher peak in the buffer (113 vs. 104) compared to the other options. While the total number of buffered containers during the day is lower for the Fixed option, it requires fewer AGVs (27 vs. 33). This trade-off between peak shaving effectiveness and operational demands highlights the importance of choosing an option aligned with logistical capabilities.

From Table B.2, we see that Configuration 1 achieves the lowest peak, with 1462 bags in the last quarter-hour. Figure B.1 shows the occupancy levels at the transfer infeed points for Configuration 1 and Base 1. Each minute, the total number of bags arriving in the last 15 minutes at the transfer infeed points is aggregated and plotted.

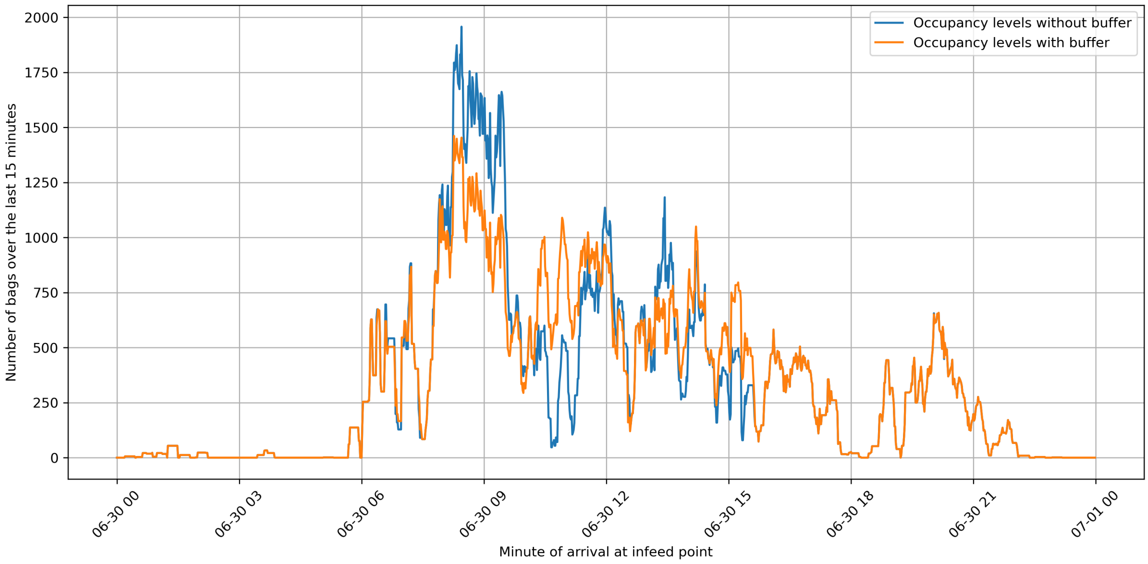


Figure 9.1: Transfer infeed point occupancy levels for Configuration 1 and Base 1

Figure B.2 shows the occupancy levels in the buffer. The y-axis represents the number of containers, and the x-axis shows the time.

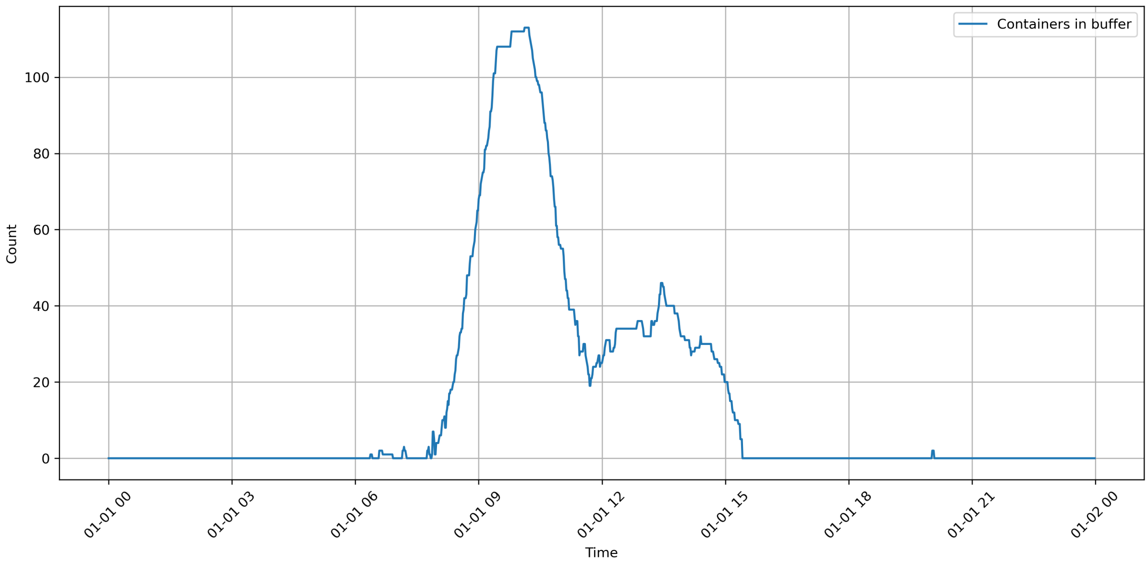


Figure 9.2: Buffer occupancy levels for Configuration 1

Optimized Release Alternative

Then, we fix the Optimized Release alternative. By comparing Configurations 4, 10, 16 and 5, 11, 17, and 6, 12, 18, we can observe the effects. Table 9.3 presents the results here, but in Appendix 13, Tables 9.3, B.6, and B.7 are displayed together.

We observe that the Polynomial and Combination options yield the most significant peak shaving in all scenarios, achieving a 26.8% reduction in peak occupancy. The Fixed option achieves a reduction of around 25.7%, so the difference is 1.1%. The Polynomial and Combination options also exhibit slightly more favorable values for standard deviation and MAC, but not for CoV, indicating a somewhat smoother and more stable occupancy profile, even though the difference is not significant. The tables clearly display these effects, with all tables sorted by 'Peak of bags' from smallest to largest.

Conf.	Option	Alternative	Scenario	Peak	STD	MAC	CoV	Peak buff	Buff. containers	AGVs
10	2: Polynomial	2: Optimized release	1: Normal	1433	350.48	26.55	1.04	118	251	31
16	3: Combination	2: Optimized release	1: Normal	1433	350.48	26.55	1.04	118	251	31
4	1: Fixed	2: Optimized release	1: Normal	1454	347.95	26.94	1.03	116	199	28
Base 1	0. No buffer	0. No buffer	1: Normal	1958	404.83	28.82	1.2	0	0	0

Table 9.3: Results on Strategic Options with Alternative 2 and Scenario 1

The Polynomial and Combination options yield the same results, achieving more peak shaving than the Fixed option. The KPIs for standard deviation, MAC, CoV, and peak values in the buffer are comparable to each other. However, it is observed that the Fixed option requires a lower number of AGVs to handle container transport. With the Optimized Release alternative fixed, configurations 10 and 16 appear to be the best-performing configurations.

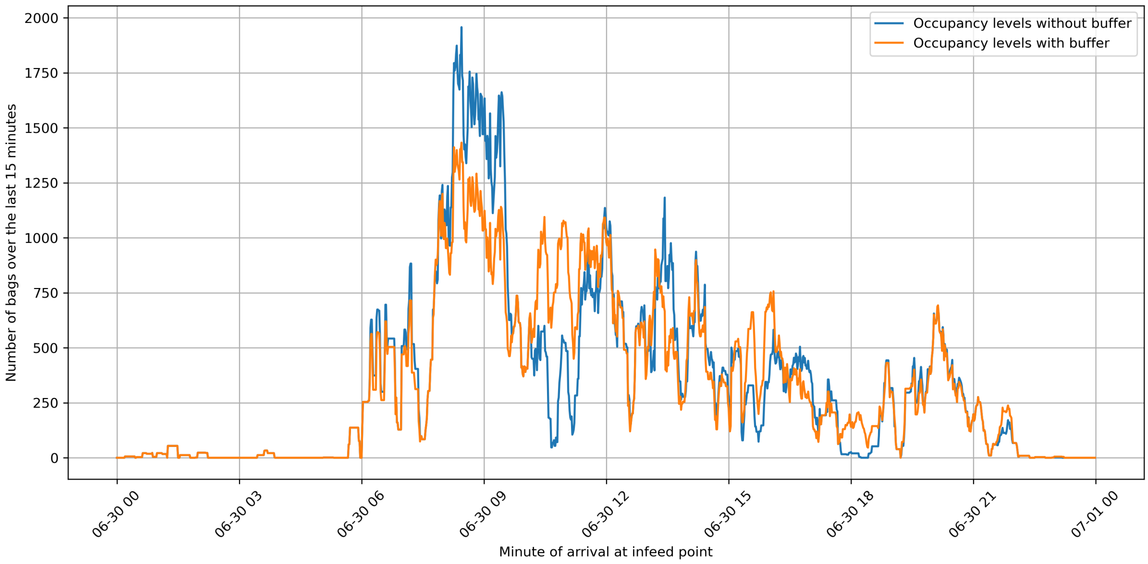


Figure 9.3: Transfer infeed point occupancy levels for Configuration 10 and Base 1

Figure B.19 shows the occupancy levels for Configuration 10 and Base 1, followed by Figure B.20, which shows the buffer occupancy levels in Configuration 10.

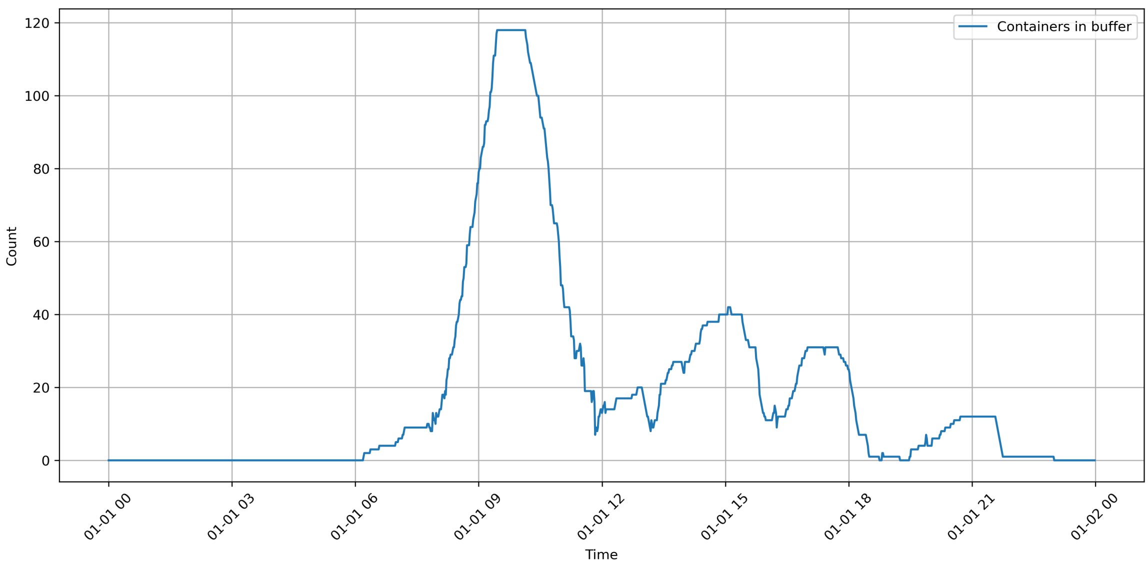


Figure 9.4: Buffer occupancy levels for Configuration 10

9.2.2 Strategic Alternatives

In this section, configurations are analyzed with a focus on the 'Alternative' variable as the only differing factor. By examining each of the 'Alternatives' under all three 'Options' and across all 'Scenarios', we can assess the specific impact of each 'Alternative' strategy on the performance metrics.

Fixed Target Option
When only looking at the Fixed option, comparisons can be made between the Early Release and Optimized Release alternatives across the scenarios. Here, configurations 1 and 4, 2 and 5, and 3 and 6 were analyzed. The corresponding tables B.8, B.9, and B.10 can be found in Appendix 13.

In scenario 1, the Optimized Release alternative achieves better peak shaving than Early Release, with 1454 bags versus 1462. However, in scenario 2 (1464 versus 1454) and scenario 3 (1736 versus 1734), Early Release performs better. The differences, however, are very small. The other KPIs are also very similar, with no significant differences observed.

Polynomial and Combination Target Option
Then, looking at the Polynomial and Combination options together—since they have the same results for these configurations—a comparison can be made between the Early Release and Optimized Release alternatives across the scenarios. Configurations 7 and 10, 8 and 11, and 9 and 12 were analyzed here. The corresponding tables B.11, B.12, and B.13 can be found in Appendix 13.

In all scenarios, the Optimized Release achieves better peak shaving than Early Release. In scenario 1, the peak is 1433 bags for Optimized Release versus 1492 for Early Release. Scenario 2 shows a difference between 1459 and 1484, while in scenario 3 the largest difference is observed, with a peak of 1663 bags for Optimized Release versus a peak of 1819 for Early Release. The differences in peaks are larger with the Polynomial and Combination options than with the Fixed Option. Although Optimized Release performs less well on STD, MAC, CoV, Peak of the buffer, and the number of buffered containers, it does perform slightly better in terms of the number of AGVs required.

9.3 Overview of Results

Same values from Table 9.1 are displayed below in Table 9.4, now sorted by 'Peak' in ascending order. As previously noted, the values for each configuration with a 'Polynomial target' or 'Combination Fixed and Polynomial target' are the same.

It is evident that Configurations 10 and 16 achieve the most peak shaving, at 26.8%. In these configurations, the Polynomial Option is combined with the Optimized Release Alternative. However, the other configurations are not far behind. Configuration 4, which uses the Optimized Release Option with the Early Release alternative, achieves a peak shaving of 25.7% and also has lower values in Standard Deviation, MAC, CoV, and AGVs, although it has a higher peak in buffer occupancy. Configuration 1 also has lower AGV requirements and has a peak shaving of 25.3%.

Looking at Scenarios 1 and 2, the STDs are all between 345.25 and 350.48. This close range suggests a high consistency in fluctuations across configurations within these scenarios, indicating that the volatility in occupancy rates is fairly uniform regardless of the configuration used.

Similarly, the MAC values are close to each other, all between 25.82 and 27.23. This narrow range suggests that the MAC remains stable across the different configurations, ensuring that peak demand is effectively managed without significant variability.

The differences in CoV are also minimal: a variation of only 0.01, which indicates that the relative variability in occupancy is nearly the same across configurations.

The peak in buffer shows slightly more variation. However, for the top 8 configurations, the peaks are quite close to each other, all between 113 and 118 containers. This similarity in buffer peaks across these configurations suggests a balanced capacity in buffering, avoiding excessive spikes and maintaining efficient flow through the BHS.

There are notable differences in the number of AGVs required. The Fixed Option requires significantly fewer AGVs (between 26 and 28) to operate compared to the Polynomial Option (between 31 and 33). This reduction in AGV requirements highlights the efficiency of the Fixed Target Option in maintaining performance with limited resources.

Conf.	**Option**	**Alternative**	**Scenario**	**Peak**	**STD**	**MAC**	**CoV**	**Peak buff**	**Buff. cons**	**AGVs**
10	2: Polynomial	2: Optimized release	1: Normal	1433	350.48	26.55	1.04	118	251	31
16	3: Combination	2: Optimized release	1: Normal	1433	350.48	26.55	1.04	118	251	31
2	1: Fixed target	1: Early release	2: Reduced cap.	1454	348.81	26.99	1.04	114	205	26
4	1: Fixed target	2: Optimized release	1: Normal	1454	347.95	26.94	1.03	116	199	28
11	2: Polynomial	2: Optimized release	2: Reduced cap.	1459	349.25	26.77	1.04	115	251	31
17	3: Combination	2: Optimized release	2: Reduced cap.	1459	349.25	26.77	1.04	115	251	31
1	1: Fixed target	1: Early release	1: Normal	1462	349.90	26.25	1.04	113	199	27
5	1: Fixed target	2: Optimized release	2: Reduced cap.	1464	350.19	27.23	1.04	117	205	28
8	2: Polynomial	1: Early release	2: Reduced cap.	1484	345.25	26.57	1.03	105	259	32
14	3: Combination	1: Early release	2: Reduced cap.	1484	345.25	26.57	1.03	105	259	32
7	2: Polynomial	1: Early release	1: Normal	1492	346.89	25.82	1.03	104	257	33
13	3: Combination	1: Early release	1: Normal	1492	346.89	25.82	1.03	104	257	33
12	2: Polynomial	2: Optimized release	3: Incr. inflow	1663	417.77	30.98	1.03	139	300	39
18	3: Combination	2: Optimized release	3: Incr. inflow	1663	417.77	30.98	1.03	139	300	39
3	1: Fixed target	1: Early release	3: Incr. inflow	1734	411.03	29.64	1.02	138	288	34
6	1: Fixed target	2: Optimized release	3: Incr. inflow	1736	414.84	30.49	1.03	142	288	35
9	2: Polynomial	1: Early release	3: Incr. inflow	1819	413.07	30.15	1.02	127	311	40
15	3: Combination	1: Early release	3: Incr. inflow	1819	413.07	30.15	1.02	127	311	40
Base 1	0. No buffer	0. No buffer	1: Normal	1958	404.83	28.82	1.20	0	0	0
Base 2	0. No buffer	0. No buffer	2: Reduced cap.	1958	404.15	29.4	1.20	0	0	0
Base 3	0. No buffer	0. No buffer	3: Incr. inflow	2283	485.38	33.88	1.20	0	0	0

Table 9.4: Full results of all configurations in peak shaving order

The comparative analysis across options and alternatives reveals variations in peak occupancy, buffer utilization, and logistical demands. While each configuration demonstrates different degrees of peak shaving and buffer efficiency, the specific metrics observed highlight the unique impact of each option and alternative combination. The results offer insight into the operational characteristics of each strategy, paving the way for an in-depth discussion in the following chapter.

10 Discussion

This chapter provides a comprehensive analysis of the findings from the experiments conducted in this study. Each subsection delves into a critical component of the research, examining the simulation setup, the forecasting model, the effectiveness of various distribution strategies, and the observed results. Assumptions and simplifications used in the simulation are discussed, as well as the limitations of the forecasting model and its implications on peak shaving. The discussion also explores the strengths and weaknesses of the chosen distribution strategies in managing peak baggage loads and how they perform under different scenarios. Finally, the chapter concludes by discussing the contribution of the research and reflecting on the results in light of these factors and identifying areas for future research that could further optimize peak shaving at transfer infeed points.

10.1 Simulation

The simulation developed for this study is based on a number of assumptions and simplifications necessary to focus on the core objective of this research: evaluating distribution strategies for buffering super cold transfer baggage at AAS. Each assumption has implications for the outcomes of the simulation and, consequently, for the insights derived.

Firstly, odd-sized baggage was excluded from the dataset, as it requires specialized handling outside of the typical baggage flow and often involves additional processes not directly related to peak occupancy at the infeed points. While this exclusion simplifies the model, it may lead to a slight underestimation of the overall baggage load during peak times. Nonetheless, this decision aligns with the practical focus of this study, as odd-sized baggage comprises a small proportion of total baggage and is typically processed separately from regular transfer baggage.

A notable assumption in this simulation is the inclusion of a 1500-second delay, which was introduced to improve the alignment of simulated baggage flows with observed real-world data. This adjustment mirrors delays seen in similar simulations at AAS and accounts for unmodeled complexities in real-world operations, such as handling delays or irregularities in the transportation process. However, the delay remains an approximation without a specific operational explanation, highlighting an area for further study to understand the sources of these recurring delays more precisely.

Only super cold transfer baggage was considered eligible for buffering in this research, with cold transfer baggage excluded despite its potential for additional peak shaving benefits. Although the inclusion of cold transfer baggage in buffering could lead to even greater reductions in peak occupancy, conversations with AAS and KLM suggested a cautious approach to minimize risks to operational timelines, particularly for shorter transfer windows. This preference reflects a conservative industry stance, where operational reliability is prioritized over potential gains in peak shaving. Future studies could explore the controlled inclusion of cold transfer baggage, especially if additional safety buffers in transport times are established.

In the clustering of ramp positions, individual ramp stands were grouped into clusters to streamline the simulation and enhance computational efficiency. This approach effectively reduces the complexity of the airport layout, making it more feasible to model overall transfer flows without detailing each stand individually. However, this simplification may slightly overlook variations in ramp-to-infeed travel times, as some individual ramp positions could be closer to or further from infeed points than others in the same cluster. The clustering approach, while simplifying the model, is expected to reasonably reflect the overall dynamics of baggage distribution.

The buffering system is treated as a black box, meaning that the internal processes within the buffer (e.g., sorting, holding capacity) are not explicitly modeled. This assumption aligns with the focus of this study on timing and strategy for buffering rather than on the specific mechanics of the buffering process itself. While this simplifies the model and enables clearer focus on peak shaving strategies, it also limits the insights into the buffer's operational constraints and its potential as an active management tool.

The calculation of the number of AGVs required for each configuration provides a realistic and accurate estimate. This calculation offers a basis for assessing AGV requirements across the different distribution strategies. However, it is worth noting that AGVs in this calculation do not follow optimized routes. They do not transport multiple containers with varying infeed destinations in a single trip, which could potentially save time by stopping at multiple infeed points in one route. Incorporating route optimization in further calculations could refine the estimate, but the current calculation already delivers a robust and realistic overview of AGV needs for each strategy.

Another assumption relates to the simplification of baggage handling processes by not explicitly modeling certain operational variables, such as shifts and breaks for operators and the availability of equipment like trains and tugs. In this simulation, the BHS is modeled as a seemingly continuous process, under the assumption that staff and equipment are sufficiently flexible to support uninterrupted operations. While in reality, staff work in shifts throughout the day, the assumption here is that these shifts overlap seamlessly, ensuring that the baggage handling process remains unaffected. This aligns with the operational goal of airports and airlines to maintain continuity in baggage flows.

A similar assumption is made for the availability of tugs. It is assumed that there are enough tugs available to support all operations without delay. In practice, however, the transport of buffered baggage would ideally be handled by AGVs to reduce reliance on manual tug drivers, particularly during peak hours when higher volumes of buffered baggage would require transport. Automating this process would alleviate the staffing challenges that arise during peak times. One of the outputs of this research is the estimation of the number of bags that will require buffering under various strategies and scenarios, which could inform further studies on the exact number of tugs or AGVs needed for implementation.

While the model does not include all potential operational disruptions—such as unexpected equipment downtime or temporary shortages of personnel—Scenario 2 in this study does simulate a Reduced Capacity condition. In this scenario, one of the four baggage halls is taken out of operation, with all baggage originally assigned to this hall rerouted to the remaining halls. This adjustment allows for preliminary insights into how the buffering strategies manage an increased load under constrained conditions. Including additional operational variables, such as personnel fluctuations or equipment breakdowns, in future simulations could further enhance the model's realism and test the robustness of these strategies under a wider range of day-to-day challenges.

A key potential of the simulation model lies in its ability to evolve into a real-time decision support system for transfer baggage handling. By integrating updated predictions from a continuously trained forecasting model, the simulation could dynamically adjust to changing conditions and make near-real-time decisions. For instance, periodic re-runs of the forecasting model with the latest baggage flow data would allow for accurate, context-sensitive predictions of occupancy levels at transfer infeed points. This enhanced accuracy could enable the proactive adjustment of buffering strategies to respond to fluctuating demands, ensuring peak shaving objectives are consistently met under varying operational conditions.

Moreover, aligning the simulation outputs with the operational deployment of AGVs at AAS could provide a practical framework for automating baggage transport. The distribution strategies developed in this research offer a foundation for implementing decision rules in AGV fleet management, optimizing their allocation in real time based on forecasted infeed occupancy. This integration could transform the simulation model from a theoretical tool into a scalable operational system, reducing bottlenecks, enhancing buffer utilization, and ensuring timely baggage delivery. Such an approach would not only improve system performance but also align with AAS's strategic objectives of leveraging automation to manage growing passenger and baggage volumes efficiently.

In summary, this simulation provides valuable insights into the effectiveness of distribution strategies for buffering super cold transfer baggage at AAS, while balancing necessary simplifications with operational relevance. Key assumptions—such as excluding odd-sized baggage, clustering ramp positions, and treating the buffer as a black box—streamlined the model to focus on peak shaving objectives. While idealized assumptions about continuous operations and AGV availability align with industry goals, they represent a simplified version of real-world conditions. Nonetheless, the simulation demonstrates the potential of leveraging predictive modeling and strategic buffering to mitigate peak occupancies effectively.

Future studies could enhance realism by incorporating factors such as operational disruptions, equipment constraints, and personnel shifts to further test the robustness of buffering strategies across a broader range of scenarios. Additionally, evolving the simulation into a real-time decision support system could enable dynamic adjustments based on updated predictions, offering a scalable and practical solution for peak shaving and resource optimization in airport BHSs.

10.2 Forecasting Model

The time series forecasting model in this study was developed to predict occupancy levels at the transfer infeed points, enabling proactive management of peak loads. For this purpose, a dataset covering 10 weeks of historical baggage data was used. This dataset, although representative of typical baggage flow patterns, presents limitations in terms of the breadth and variety of scenarios it can account for. A more extensive dataset, spanning several months or even years, would likely capture a broader range of operational conditions and seasonal fluctuations, contributing to a more robust and accurate forecasting model.

The selection of the SARIMA and TBATS models was based on their strengths in handling seasonal patterns, with SARIMA offering simplicity and effectiveness for daily cycles and TBATS providing flexibility for complex seasonal patterns, including weekly variations. However, other advanced forecasting techniques, such as machine learning-based methods (e.g., LSTM, Prophet), could potentially enhance accuracy by capturing non-linear patterns and adjusting dynamically to sudden changes in baggage flows. These models, particularly when paired with a larger dataset, could offer higher precision in predicting peak occupancy rates, which may result in even more optimized buffering decisions.

Another factor to consider is the model's dependency on the regularity and quality of historical data. Any gaps or inconsistencies in the data—due to operational disruptions, data logging issues, or changes in baggage handling procedures—could introduce bias in the predictions. A larger dataset might help smooth out these anomalies, but it also underscores the importance of rigorous data cleaning and validation processes for forecasting models in dynamic environments like airport baggage handling.

In summary, while the forecasting model developed in this study provided a useful basis for decision-making in peak shaving, its accuracy and reliability could be further improved with a larger, more diverse dataset and the exploration of alternative modeling approaches. Future research might focus on leveraging additional data sources or advanced methods to enhance the predictive power of the model and support even more effective distribution strategies.

10.3 Distribution Strategies

The distribution strategies in this study were developed to mitigate peak occupancy at the transfer infeed points through the buffering of super cold transfer baggage. One of the core strategies employed was the use of a Fixed Target Option of 600 bags per 15-minute interval. This threshold, though justified based on observed baggage flows and infeed capacities, was not mathematically optimized. A more rigorous investigation into the ideal Fixed Target value could provide a more accurate threshold, potentially enhancing the efficacy of peak shaving and impacting the overall conclusions of this study. Such research could explore various operational scenarios and sensitivities around different threshold levels to define an optimal fixed target that minimizes system congestion while accommodating the natural fluctuations in baggage inflow.

In addition to the fixed target strategy, a polynomial function with degree 9 was used as a dynamic target to account for daily variations in baggage inflow. This polynomial degree was selected after testing various polynomial functions, balancing flexibility with the risk of overfitting. While the degree 9 polynomial function provided a reasonable approximation of daily patterns, it is not necessarily the most optimal form. Future research could focus on refining the polynomial function, possibly through advanced curve-fitting techniques or machine learning models, to derive a target function that more precisely aligns with real-world inflow patterns. An improved target function could offer even more targeted peak shaving, allowing the buffer to dynamically adjust to high-occupancy periods without sacrificing processing efficiency during low-demand times.

Overall, while both the Fixed and Polynomial option strategies contributed to reducing peak occupancy, the methods used to set these targets could be further refined. A more detailed exploration of the target values and function forms would enhance our understanding of their impact on the transfer infeed system, potentially leading to a more adaptive and efficient distribution strategy. This would support not only improved peak shaving but also a better alignment with the operational capacities of the BHS at AAS.

10.4 Discussing Results

The results of this study provide a comprehensive view of how different configurations impact peak occupancy at the transfer infeed points. The configurations were ranked based on their ability to reduce peak loads, with key metrics such as STD, MAC, CoV, the peak in the buffer, and the number of AGVs required for each configuration. These results offer insights into the performance and trade-offs associated with each distribution strategy.

10.4.1 Top Strategies

The configurations utilizing the Polynomial and Combination target functions with the Optimized Release strategy in a normal scenario (e.g., Configurations 10 and 16) achieved the highest peak shaving, reducing peak occupancy to 1433 bags per quarter hour, or 26.7%. This outcome suggests that combining a Polynomial target with an Optimized Release mechanism is highly effective under typical conditions. However, this approach also requires a significant number of AGVs, which could impact operational costs and resource allocation.

Among the strategy options, the Polynomial and Combination approaches consistently achieve the most effective peak shaving across scenarios. The Fixed Target strategy, while resulting in slightly less peak shaving, demonstrates a clear advantage in requiring fewer AGVs, making it a more resource-efficient option in scenarios with limited AGV availability. For example, Configuration 3 utilizes a Fixed Target with Optimized Release, maintain a peak shaving of 25.7% while keeping AGV demands lower.

In terms of strategic release alternatives, the Optimized Release strategy outperforms Early Release in achieving peak shaving across almost all target options. This makes Optimized Release a preferable choice for configurations focused on minimizing peak occupancy. The trade-off between AGV requirements and peak shaving efficiency thus becomes central in selecting an approach, depending on an airport's operational resources and specific peak shaving goals.

10.4.2 Robustness of Different Strategies

The consistency of performance on STD, MAC, and CoV demonstrates the robustness of the different strategy combinations across the three scenarios. Although there are slight differences in these KPIs between strategies, the values for STD, MAC, and CoV remain remarkably close across the top-performing strategy combinations. This narrow range of KPI values suggests that each of these strategies performs with a comparable degree of reliability and stability, maintaining robust results despite the variability in conditions. The minimal variation across these metrics indicates that, while some strategies may have slight advantages in specific scenarios, the differences are not substantial, underscoring the overall resilience of these approaches in managing peak occupancy.

10.4.3 Peak in Buffer

The results indicate that the buffer peak varies across configurations but remains close within the top-performing strategies. For the best 3 configurations, buffer peaks range between 116 and 118 containers, showing a narrow spread and implying efficient buffering across these strategies. This narrow range suggests that these different strategies manage buffer capacity effectively without excessive peaks, maintaining a steady flow through the BHS.

Notably, the strategy alternative Early Release consistently shows the lowest buffer peaks across all scenarios, though it does not achieve the highest peak shaving overall. This suggests a clear relationship between buffer peak levels and the release strategy chosen: Early Release effectively reduces buffer occupancy but at the cost of slightly lower peak shaving performance. This trade-off highlights that while Early Release can be beneficial for minimizing buffer space requirements, it may be less optimal for airports prioritizing maximum peak shaving. The limited space on airside at airports like AAS makes buffer peak management a critical factor, as overly high buffer occupancy could lead to space constraints and operational bottlenecks. High buffer peaks not only require additional physical space but also influence the need for more AGVs to transport containers in and out of the buffer. In airports with restricted buffer areas, like AAS, the ability to keep buffer peaks within a narrow range is advantageous, as it minimizes the spatial footprint of the buffer while still achieving efficient peak shaving.

This consideration highlights why buffer peak is an important KPI that could weigh more heavily in decision-making, depending on the specific airport's layout and space constraints. At airports with tighter spatial limitations, selecting a strategy with lower or more consistent buffer peaks may be prioritized over other factors to avoid unnecessary congestion and ensure smooth baggage flow. Thus, while multiple strategies may perform similarly in peak shaving, the specific buffer demands of each strategy should be carefully evaluated according to the airport's physical capacity and operational constraints.

10.4.4 AGV Requirements

AGV requirements differ notably among configurations. The Fixed Target Option strategy requires significantly fewer AGVs than the Polynomial and Combination Options, which makes it advantageous in scenarios where resources are constrained. The Polynomial and Combination targets, while more effective in peak shaving, demand higher AGV allocations due to the dynamic nature of their buffering strategies. This insight underscores the need to balance peak shaving effectiveness with resource availability when selecting a distribution strategy.

The demand for AGVs is a particularly important consideration in the airside environment, where space and operational flow are already highly constrained. With multiple vehicles and equipment moving simultaneously, adding more AGVs to handle baggage buffering could increase congestion and operational complexity, potentially impacting safety and efficiency. Given these airside constraints, airports must carefully assess whether the additional peak shaving benefits offered by Polynomial and Combination strategies justify the higher AGV requirements, or if a less intensive Fixed Target approach might be more practical.

Ultimately, while Polynomial and Combination strategies provide stronger peak shaving results, their higher AGV demands could limit their feasibility in busy airport environments where AGV traffic must be minimized to maintain smooth operations. This reinforces the need for airports to consider both the peak shaving effectiveness and AGV resource impact of each strategy in the context of their specific operational capacity and airside conditions.

10.5 Cost Considerations

Although this research focuses on the operational impact of peak shaving and the evaluation of distribution strategies, it is important to reflect on the potential cost implications. One of the key motivations for peak shaving is to achieve a more evenly distributed inflow of baggage, not only at the transfer infeed points but also across other subsystems of the BHS. By reducing peaks, personnel costs can be minimized, as less or no additional staff would be required to handle baggage during peak hours. This has implications beyond the transfer infeed points, as uneven baggage inflows also create peaks in other BHS subsystems, such as screening, sorting, and make-up. A smoother flow reduces the need for additional resources, resulting in cost savings across the baggage handling process.

Furthermore, the choice of distribution strategy has significant cost implications. Strategies that rely on larger buffer capacities inherently require more investment in infrastructure. At AAS, for instance, a larger buffer would need to be constructed, potentially involving high capital costs. On the other hand, strategies that necessitate increased use of AGVs to manage baggage transport come with operational costs, including the purchase, maintenance, and management of additional AGVs. These costs must be carefully weighed against the operational benefits of the respective strategy.

While this study does not quantify these costs, the findings provide valuable input for future cost-benefit analyses. Such analyses could explore the trade-offs between infrastructure investments, resource allocation, and operational efficiency to determine the most economically viable approach for implementing peak shaving and distribution strategies at AAS.

11 Conclusion

In this chapter, the key findings of this research are summarized by addressing each sub-research question, followed by a final answer to the main research question. The conclusion highlights the current state of baggage handling at AAS, the role of peak shaving in managing transfer infeed points, requirements for optimizing cold transfer baggage distribution, and the effectiveness of the tested distribution strategies. This chapter concludes with insights into the implications of these findings for operational practices.

Sub-question 1: What is the current state of baggage handling processes at the transfer infeed points at AAS?
The current state of baggage handling at AAS reflects a complex but highly structured process, designed to manage a variety of baggage types with different requirements. However, analysis reveals that transfer infeed points experience significant fluctuations in occupancy levels throughout the day. These fluctuations are primarily due to peaks in the arrival of transfer baggage, which are influenced by the hub function of AAS and the timing of connecting flights. These peaks can create bottlenecks at the transfer infeed points, leading to potential delays and operational challenges. While the BHS has been optimized to a certain extent together with the limited space for expansion of the BHS, it faces strain during high-occupancy periods, demonstrating the need for further measures, such as the buffering process of cold transfer baggage to influence the unequal inflow of baggage on the BHS.

Sub-question 2: How does peak shaving work in the context of an airport?
Peak shaving in the context of an airport involves strategies to smoothen the fluctuating occupancy rates at various points within the BHS, such as the transfer infeed points. This approach aims to prevent the system from being overwhelmed during peak times by redistributing the workload more evenly over time. For peak shaving at the transfer infeed points at AAS, the focus is specifically on cold transfer baggage, which has longer layover times, allowing it to be buffered temporarily. By reintroducing this baggage during periods of lower demand, peak shaving helps maintain a balanced flow at the system and through the system, reduces bottlenecks, and enhances operational efficiency. This research has shown that peak shaving can lead to a more stable workload distribution, allowing resources to be allocated more effectively across different periods.

Sub-question 3: What is needed to optimize the distribution of cold transfer baggage?
Optimizing the distribution of cold transfer baggage requires a combination of accurate forecasting, flexible buffering capacity, and a strategic approach to decide when to buffer baggage, and when to reintroduce baggage into the BHS. Firstly, a time series forecasting model is essential for predicting future occupancy levels at transfer infeed points, allowing for proactive adjustments to buffering and release times. Secondly, a well-organized buffer system with sufficient capacity is necessary to accommodate fluctuations in baggage volumes. Finally, rule-based strategies for reintroducing baggage based on occupancy data are crucial to ensure that buffered baggage is released during off-peak times. Together, these elements help create a more responsive system that adjusts dynamically to occupancy demands, maximizing the efficiency of peak shaving.

Sub-question 4: Which distribution strategies are suitable for buffering cold transfer baggage, and how do these strategies impact peak shaving under different scenarios?

This research identified three main approaches to determine when baggage should be buffered:

- **Fixed Target:** Baggage is buffered when occupancy levels exceed a fixed threshold.
- **Polynomial Target Function:** Baggage is buffered based on a polynomial function that dynamically adjusts the target occupancy level, allowing for more flexible peak shaving during high inflow periods.
- **Combination:** Baggage is buffered if occupancy is exceeding a combined function of a Fixed Target and the Polynomial Function.

Additionally, two alternatives were tested to determine the optimal timing for reintroducing buffered baggage at the infeed points:

- **Early Release:** Buffered baggage is released earlier, aiming to keep buffer levels manageable and minimize congestion in buffer areas, though with potentially less peak shaving impact.
- **Optimized Release:** Buffered baggage is reintroduced to the BHS by filling the deepest troughs in the forecasted occupancy, maximizing peak shaving by strategically spreading out the inflow.

Combining these three buffering approaches with the two reintroduction alternatives results in six unique distribution strategies. These strategies were tested across three scenarios, each representing different operational conditions, to evaluate their effectiveness in reducing peak occupancy.

The combined strategies achieved peak shaving reductions ranging from 23.8% to 26.8% on a standard day. The strategy with the highest peak shaving efficiency involved a polynomial target function to determine when baggage should be buffered, along with an optimized release alternative that reintroduces baggage by filling the deepest troughs in the occupancy forecast. This combination achieved a peak shaving of 26.8%. However, this strategy required a relatively high number of AGVs to handle the transport, indicating a trade-off between peak shaving effectiveness and resource demand.

Strategies employing the Fixed Target option, while slightly less effective in peak reduction (achieving 25.7%), required fewer AGVs for transportation. In comparison to the highest peak shaving strategy, the Fixed Target option could reduce AGV requirements by up to 9.7% (28 AGVs compared to 31), making it a more resource-efficient choice on an averaged day.

While still achieving a peak shaving of 25.3%, the Early Release alternative consistently maintained the lowest peak occupancy levels within the buffer, reducing buffer peaks by as much as 11.9%. By releasing baggage from the buffer earlier, this strategy helps to prevent congestion within the buffer area, making it particularly advantageous in scenarios where maintaining buffer capacity is critical. This approach minimizes the risk of overcrowding in the buffer during peak inflow periods, thereby enhancing operational flexibility and reducing the chance of delays that might occur when buffer capacity is stretched to its limit. The Early Release alternative could be beneficial in operational situations where limited buffer space is available, or where there is a need for a high turnover rate in the buffer area to accommodate continuously incoming baggage. By focusing on a proactive release of buffered baggage, this strategy allows for a smoother flow back into the system, which could help airports better handle unexpected spikes in baggage volume or manage space constraints more effectively. Overall, this alternative presents a valuable trade-off by prioritizing buffer management over maximum peak shaving at the infeed points, offering operational benefits under specific conditions.

These findings illustrate how each distribution strategy impacts not only the occupancy levels at the transfer infeed points but also other operational aspects such as AGV requirements and buffer usage. Each strategy presents unique trade-offs, allowing AAS to select an approach that best aligns with its operational priorities and constraints.

Sub-question 5: What operational considerations and resource implications arise from implementing the simulated distribution strategies, and how feasible are they for reducing peak occupancy at AAS transfer infeed points?
Implementing the distribution strategies for buffering cold transfer baggage involves several operational considerations and resource implications, impacting the feasibility of each strategy in practice.

Firstly, a primary operational consideration is the availability and management of AGVs. Strategies that maximize peak shaving, particularly those using the Polynomial Target Function and Optimized Release, demand a higher number of AGVs due to the need for frequent and well-timed transport operations. While this approach achieves the highest peak shaving (up to 26.8%), it also increases AGV requirements. This raises a key resource implication: the cost and logistical challenge of deploying additional AGVs, which may not be feasible during all operational periods or under budget constraints. Moreover, increasing the number of AGVs leads to more traffic on the airside, which is not ideal for safe and efficient ground operations. Airports with limited AGV availability might therefore favor Fixed Target strategies, which achieve satisfactory peak shaving (around 25.7%) with fewer transport resources and reduced impact on airside traffic.

Secondly, managing buffer capacity is crucial to prevent congestion within the buffer areas, especially during high inflow periods. The Early Release alternative demonstrated a lower peak in buffer occupancy. This strategy is particularly advantageous when buffer space is limited, as it proactively releases baggage to maintain manageable buffer levels. However, the trade-off is slightly reduced peak shaving at the infeed points, highlighting a need to balance buffer capacity management with peak shaving objectives.

In conclusion, the effectiveness of each strategy in reducing peak occupancy must be weighed against resource and operational constraints. Airports like AAS must consider the AGV requirements and buffer space limitations when selecting the most suitable approach. Thus, while the strategies demonstrate promising potential for peak shaving, practical implementation will require balancing these factors to optimize both resource usage and operational efficiency at transfer infeed points.

Main research question: How to support Amsterdam Airport Schiphol to identify the best distribution strategy for super cold transfer baggage through a simulation model?
The simulation model developed in this research, combined with a predictive forecasting model, has provided AAS with valuable insights into the impact of various distribution strategies for buffering super cold transfer baggage. By dynamically testing and analyzing these strategies under different operational scenarios, the research has demonstrated how each approach can address specific objectives.

The results highlight that the choice of a distribution strategy depends on the operational priorities of AAS. For instance, strategies focusing on peak shaving at transfer infeed points, such as those using a polynomial target function and optimized release timing, are most effective in reducing peak occupancy. However, these strategies require more AGV resources, which may limit their feasibility during resource-constrained periods. Conversely, strategies employing fixed targets strike a balance between achieving substantial peak shaving and minimizing AGV demand, offering a more resource-efficient alternative.

Additionally, strategies like the Early Release alternative prioritize maintaining low buffer occupancy levels, which is particularly beneficial when buffer space is limited or a high turnover rate is required. This approach helps ensure smooth buffer operations, reducing the risk of congestion while still achieving reasonable levels of peak shaving at the infeed points.

In conclusion, the simulation model has enabled AAS to assess the trade-offs between different objectives, such as reducing peak occupancy, optimizing AGV usage, and managing buffer space. This research provides a data-driven framework to support AAS in selecting the most suitable distribution strategy based on its specific operational goals and constraints, enhancing both the efficiency and resilience of its BHS.

12 Recommendations

Further research could explore optimizing the criteria for which baggage should be buffered. This study exclusively focused on buffering super cold transfer baggage, defined as baggage with a layover time of more than three hours. However, investigating the potential of buffering cold transfer baggage with layover times of around two hours could yield new insights into peak shaving effectiveness. By examining how these shorter layover bags contribute to occupancy peaks, researchers could evaluate whether including them in buffering strategies would result in additional reductions in peak occupancy at transfer infeed points. Understanding the impact of different layover times on peak shaving could provide a more nuanced approach to buffering decisions, maximizing efficiency while ensuring baggage still meets its outbound flight schedules.

A recommendation for future research is to develop a more advanced forecasting model for occupancy rates at infeed points. In this study, a time series forecasting model was used, trained on a dataset of ten weeks. However, other methods, such as machine learning models or deep learning algorithms, could potentially provide more accurate predictions than traditional time series models. By using a larger dataset and applying these more advanced methods, the accuracy of the predictions could be further improved. This could not only contribute to more precise peak shaving at infeed points but also enhance the effectiveness of buffering strategies. Moreover, a more detailed model would allow for better forecasting of occupancy rates, supporting more efficient management of peak loads within the BHS system.

Another recommendation is to add greater flexibility to the infeed point assignment process by assigning containers a secondary or even tertiary infeed point preference. In the current setup, each container is assigned to a single infeed point based on the type of baggage it carries. However, by introducing additional preferences, containers could be redirected to alternative infeed points if the primary point is experiencing high congestion. This added flexibility would enable peak shaving to be achieved not only at the system-wide level but also at individual infeed points, allowing for more precise management of occupancy peaks. This additional layer of assignment could help to more effectively distribute baggage across multiple infeed points, balancing occupancy more dynamically and potentially enhancing the overall impact of peak shaving efforts. This more granular approach could allow for targeted adjustments to occupancy levels at specific infeed points, addressing localized bottlenecks more efficiently.

Another recommendation is to investigate the downstream impact of transfer baggage inflow at infeed points on other subsystems within the BHS. By examining the correlation between occupancy rates across different BHS subsystems, further research could provide valuable insights into how the buffering process impacts other parts of the BHS. This could lead to the development of new distribution strategies for cold transfer baggage, designed not only to alleviate congestion at transfer infeed points but also to relieve pressure on other BHS subsystems. For instance, security screening is a critical checkpoint, as each transfer bag must be screened before departure on its outbound flight. If occupancy levels at security screening fluctuate significantly, these variations could be factored into distribution strategies, potentially enabling peak shaving efforts at security. Such an approach would allow for more comprehensive management of occupancy peaks throughout the BHS, maximizing system-wide efficiency.

While this research focuses on the operational impact of peak shaving, the cost implications of implementing distribution strategies asks for further investigation. Peak shaving could reduce the need for additional personnel during peak periods, as a smoother baggage inflow across the transfer infeed points and other BHS subsystems may eliminate the need for temporary staff or overtime. These operational savings could offset some of the costs associated with implementing peak shaving strategies.

Additionally, the strategies explored in this research entail varying financial implications. For instance, strategies requiring larger buffer capacities may involve significant capital investments at AAS, including infrastructure design, construction, and integration. Alternatively, strategies relying on increased AGV deployment would require both upfront procurement and ongoing costs for maintenance and energy. Future research should conduct a comprehensive cost-benefit analysis to evaluate the financial feasibility of different strategies, weighing potential staffing and operational savings against infrastructure and resource investments. Innovative approaches could also be explored to reduce upfront costs while achieving long-term operational benefits.

13 Reflection

During the execution of this research, I gained valuable insights into methodology, practical challenges, and the impact of decisions on outcomes. In this chapter, I reflect on these aspects and share the lessons learned throughout the process.

Methodological Reflection

In the initial months of my research, I was heavily focused on optimization modeling. This approach eventually proved to be too complex for the desired output. After discussions with assistant professors at TU Delft, I realized that a different approach would be more effective. If I had recognized this earlier, it might have saved time and brought the research to its conclusion more quickly. The key lesson I take from this is to avoid becoming too attached to a predefined plan and to remain open to adjustments when needed. Ultimately, developing a simulation model—though time-consuming and challenging—proved to be the right choice. It provided valuable insights into the defined knowledge gap of my research.

Practical Challenges

The most difficult part of the research was the beginning. Limited knowledge of the subject and uncertainty about the direction made the starting phase challenging. As mentioned earlier, it is sometimes better to let go of preconceived plans. Another challenge was programming. Despite these difficulties, this phase led to significant personal growth. My Python skills, for instance, improved greatly. While I initially only had basic knowledge of Python, I now feel confident in my ability to handle complex tasks. This progress was achieved by continuously refining my code and solving challenges in the simulation process.

Impact of Choices

The decisions I made during the research had a clear and significant impact on the results. This included assumptions and definitions in the simulation, the selection of forecasting models, and the chosen distribution strategies. These decisions shaped the study and determined its outcomes. For example, I chose to exclude strategies and models that were unrealistic or performed poorly, which allowed me to focus on the most promising options. This careful selection helped to create results that were more relevant and practical.

Lessons Learned

Throughout the research, I often set ambitious timelines for myself. However, I underestimated the amount of work required, making some of these deadlines difficult to meet. This experience taught me the importance of realistic planning, especially in complex studies. Additionally, I learned to let go of my original ideas, such as my focus on optimization modeling. Being flexible and allowing the research to evolve naturally ultimately improved the quality of the results.

Overall, I look back on a process that was both challenging and educational. The insights I gained in methodology, programming, and project management have provided a strong foundation for my future development.

Appendix A: Code

This appendix contains the full code used for simulating baggage handling and forecasting baggage occupancy rates for AAS. The code is organized into multiple section's, each corresponding to different aspects of the simulation process. It includes data loading, baggage processing, simulation of unloading, container assignment, and time series forecasting. Various models such as SARIMA and TBATS are used to predict baggage flow. Accuracy is evaluated by comparing simulated data against actual data using metrics like RMSE, MAPE, and MAE. The code also incorporates data visualization and polynomial fitting to further analyze the forecasting results. Each code section is provided with explanations and outputs, contributing to a comprehensive simulation of baggage operations.

Appendix A1: Data Loading and Initial Processing

This code section loads and processes baggage data for the simulation. It begins by reading a large CSV file in chunks to manage memory, counting any malformed rows. Key filtering steps include selecting only transfer-related records, converting time columns to datetime format, and filtering for specific airport exit areas. Schengen status, baggage type, and body type classifications are added based on airport codes and aircraft types. The data is then filtered for 2024 and exported to a new CSV, ready for further analysis in the simulation.

```
import pandas as pd
import os
import time

start_time = time.time()

file_path = 'C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\data1.csv'
bad_line_count = 0  # Initialize a counter for bad lines

# Define a custom handler to count bad lines
def bad_line_handler(bad_line):
    global bad_line_count
    bad_line_count += 1

try:
    # Specify the delimiter and handle quoting issues
    df = pd.read_csv(file_path, delimiter=';', on_bad_lines=bad_line_handler, engine='python',
    ↪ quotechar='"')
    print(f"Total number of rows in original data file: {df.shape[0]}")
except MemoryError:
    print("Het bestand is te groot om in het geheugen te laden met pandas.")
    chunks = pd.read_csv(file_path, delimiter=';', chunksize=1000000, on_bad_lines=bad_line_handler,
    ↪ engine='python', quotechar='"')
    df = pd.concat(chunks)
    print(f"Total number of rows in original data file (chunked): {df.shape[0]}")

print(f"Number of bad lines skipped: {bad_line_count}")

# Check the column names
df.columns = df.columns.str.strip()
print(df.columns)  # Debugging: Check if 'ENTRY_FUNCTION' is present

# Stap 1.1: Filteren op ENTRY_FUNCTION = TransferIn, niet-lege tijdvelden
df = df[df['ENTRY_FUNCTION'] == 'TransferIn']
df = df.dropna(subset=['SIBT', 'AIBT', 'SOBT', 'AOBT'])

# Converteer de tijdkolommen naar datetime-formaat
tijdkolommen = ['SIBT', 'AIBT', 'SOBT', 'AOBT']
for kolom in tijdkolommen:
    df[kolom] = pd.to_datetime(df[kolom], errors='coerce')

# Filter alleen de EXIT_AREAs ['D', 'E', 'W', 'Z']
df = df[df['EXIT_AREA'].isin(['D', 'E', 'W', 'Z'])]
```

```python

# Indelen Schengen en Non-Schengen airport codes
schengen_airports = [
    'AAL', 'ACE', 'AES', 'AGP', 'AHO', 'ALC', 'AOK', 'ARN', 'ATH', 'BCN', 'BER', 'BES', 'BGO', 'BGY',
    ↪ 'BLL', 'BLQ', 'BOD',
    'BRE', 'BRI', 'BRS', 'BRU', 'BSL', 'BUD', 'BZG', 'CAG', 'CDG', 'CFE', 'CFU', 'CHQ', 'CPH', 'CTA',
    ↪ 'DBV', 'DRS', 'DUS',
    'EFL', 'EMA', 'EXT', 'FAO', 'FCO', 'FLR', 'FNC', 'FRA', 'FUE', 'GDN', 'GOA', 'GOT', 'GRO', 'GRZ',
    ↪ 'GVA', 'HAJ', 'HAM',
    'HEL', 'HER', 'IBZ', 'INN', 'INV', 'JMK', 'JTR', 'KEF', 'KGS', 'KLX', 'KRK', 'KRS', 'KTW', 'KVA',
    ↪ 'LBA', 'LCY', 'LED',
    'LIN', 'LIS', 'LJU', 'LPA', 'LPI', 'LPL', 'LTN', 'LUX', 'LYS', 'MAD', 'MAH', 'MJT', 'MLA', 'MPL',
    ↪ 'MRS', 'MUC', 'MXP',
    'NAP', 'NCE', 'NCL', 'NTE', 'NUE', 'NWI', 'OLB', 'OPO', 'ORY', 'OSL', 'OTP', 'OVD', 'PDL', 'PMI',
    ↪ 'PMO', 'POZ', 'PRG',
    'PSA', 'PUY', 'PVK', 'REU', 'RHO', 'RIX', 'RNS', 'SCQ', 'SEN', 'SKG', 'SOF', 'SOU', 'SPC', 'SPU',
    ↪ 'STN', 'STR', 'SVG',
    'SVQ', 'SXB', 'SXF', 'SZG', 'TFS', 'TIA', 'TLL', 'TLS', 'TRN', 'TXL', 'VCE', 'VIE', 'VLC', 'VNO',
    ↪ 'VRN', 'VXO', 'WAW',
    'WRO', 'ZAG', 'ZRH', 'ZTH'
]

non_schengen_airports = [
    'ABZ', 'ACC', 'ADB', 'AGA', 'AHU', 'ALA', 'AMM', 'ASR', 'ATL', 'AUA', 'AUH', 'AUS', 'AYT', 'BAH',
    ↪ 'BEG', 'BEY', 'BFS',
    'BHD', 'BHX', 'BIO', 'BJV', 'BKK', 'BLR', 'BOG', 'BOJ', 'BOM', 'BON', 'BOS', 'CAI', 'CAN', 'CGK',
    ↪ 'CMN', 'CPT', 'CTG',
    'CTU', 'CUN', 'CUR', 'CWL', 'DAR', 'DEL', 'DFW', 'DIA', 'DLM', 'DMM', 'DOH', 'DPS', 'DSA', 'DTW',
    ↪ 'DUB', 'DXB', 'EBB',
    'EBL', 'EDI', 'ESB', 'EWR', 'FEZ', 'FOR', 'GIG', 'GLA', 'GRU', 'GUW', 'GYE', 'GZP', 'HAV', 'HGH',
    ↪ 'HKG', 'HRG', 'HUY',
    'IAD', 'IAH', 'ICN', 'IST', 'JED', 'JFK', 'JNB', 'JRO', 'KBP', 'KGL', 'KIX', 'KWI', 'KYA', 'LAS',
    ↪ 'LAX', 'LCA', 'LGW',
    'LHR', 'LIM', 'LOS', 'MAN', 'MBJ', 'MCO', 'MCT', 'MEX', 'MIA', 'MME', 'MNL', 'MRU', 'MSP', 'MSQ',
    ↪ 'NBE', 'NBO',
    'NDR', 'NRT', 'OHD', 'ORD', 'ORK', 'OUD', 'PBM', 'PDX', 'PEK', 'PFO', 'PHL', 'PKX', 'POS', 'PTY',
    ↪ 'PUJ', 'PVG', 'RAK',
    'RMF', 'RMO', 'RUH', 'SAW', 'SCL', 'SEA', 'SFO', 'SID', 'SIN', 'SLC', 'SMI', 'SVO', 'SXM', 'SYD',
    ↪ 'TBS', 'TLV', 'TNG',
    'TPA', 'TPE', 'TRD', 'TRF', 'TTU', 'TUN', 'UIO', 'VAR', 'WDH', 'XMN', 'YEG', 'YUL', 'YVR', 'YYC',
    ↪ 'YYZ', 'ZNZ'
]

# Bereken de FLIGHT_TYPE kolom
def determine_flight_type(row):
    if (pd.isna(row['IN_AIRPORT']) and row['OUT_AIRPORT'] in schengen_airports) or \
       (row['IN_AIRPORT'] in schengen_airports and row['OUT_AIRPORT'] in schengen_airports):
        return 'Schengen'
    elif (pd.isna(row['IN_AIRPORT']) and row['OUT_AIRPORT'] in non_schengen_airports) or \
         (row['IN_AIRPORT'] in non_schengen_airports or row['OUT_AIRPORT'] in non_schengen_airports):
        return 'Non Schengen'
    else:
        return 'Non Schengen'

df['FLIGHT_TYPE'] = df.apply(determine_flight_type, axis=1)

# Correct FLIGHT_TYPE for consistency within the same flight
df['FLIGHT_TYPE_CORRECTED'] = df.groupby('IN_FLIGHT_DESIGNATOR')['FLIGHT_TYPE'].transform(lambda x: x.
    ↪ mode()[0] if not x.mode().empty else 'Non Schengen')
df['FLIGHT_TYPE'] = df['FLIGHT_TYPE_CORRECTED']
df = df.drop(columns=['FLIGHT_TYPE_CORRECTED'])

# Voeg de BAGGAGE_TEMPERATURE kolom toe
```

```python
def add_baggage_temperature(df):
    def determine_baggage_temperature(row):
        if pd.isnull(row['SIBT']) or pd.isnull(row['SOBT']):
            return 'Cold Transfer'
        connection_time = (row['SOBT'] - row['SIBT']).total_seconds() / 60.0
        if connection_time < 30:
            return 'Cold Transfer'
        elif 30 <= connection_time < 50:
            return 'Tail-to-Tail'
        elif row['FLIGHT_TYPE'] == 'Non Schengen' and connection_time < 90:
            return 'ShoCon'
        elif row['FLIGHT_TYPE'] == 'Schengen' and connection_time < 80:
            return 'ShoCon'
        elif row['FLIGHT_TYPE'] == 'Non Schengen' and connection_time >= 180:
            return 'Super Cold Transfer'
        elif row['FLIGHT_TYPE'] == 'Non Schengen' and connection_time >= 90:
            return 'Cold Transfer'
        elif row['FLIGHT_TYPE'] == 'Schengen' and connection_time >= 80:
            return 'Cold Transfer'
        else:
            return 'Cold Transfer'

    df['BAGGAGE_TEMPERATURE'] = df.apply(determine_baggage_temperature, axis=1)
    return df

df = add_baggage_temperature(df)

# Filter de rijen waar BAGGAGE_TEMPERATURE niet gelijk is aan 'Tail-to-Tail'
df = df[df['BAGGAGE_TEMPERATURE'] != 'Tail-to-Tail']

# Check row count after filtering
print(f"Number of rows after filtering: {df.shape[0]}")

# Pas ENTRY_AREA en EXIT_AREA aan
def adjust_entry_exit_areas(df):
    df['ENTRY_AREA'] = df['ENTRY_AREA'].replace('D', 'TSD')
    df['EXIT_AREA'] = df['EXIT_AREA'].replace('TSD', 'D')
    return df

df = adjust_entry_exit_areas(df)

# Kolom conversie naar de juiste formaten
kolommen_naar_datetime = [
    'DATETIME_OF_ENTRY', 'DATETIME_OF_EXIT', 'SIBT', 'AIBT', 'SOBT', 'AOBT',
    'DATETIME_STORE_ENTRY', 'DATETIME_IN_STORE', 'DATETIME_STORE_EXIT'
]
kolommen_naar_object = [
    'ENTRY_LOCATION', 'ENTRY_NAME', 'ENTRY_AREA', 'ENTRY_FUNCTION', 'EXIT_LOCATION',
    'EXIT_NAME', 'EXIT_AREA', 'EXIT_FUNCTION', 'EXIT_STATION_ID', 'FINAL_STATION_ID',
    'STORE_LOC', 'IN_FLIGHT_DESIGNATOR', 'IN_AIRLINE', 'IN_HANDLER',
    'IN_RAMP', 'IN_AIRPORT', 'IN_AC_IATA', 'OUT_FLIGHT_DESIGNATOR',
    'OUT_AIRLINE', 'OUT_HANDLER', 'OUT_RAMP', 'OUT_AIRPORT', 'OUT_AC_IATA'
]

def convert_columns(df):
    for kolom in kolommen_naar_datetime:
        df[kolom] = pd.to_datetime(df[kolom], errors='coerce')
    for kolom in kolommen_naar_object:
        df[kolom] = df[kolom].astype('object')
    return df

df = convert_columns(df)

```

```python
# Add body type, baggage type, and flight type columns
def add_body_baggage_flight_type_columns(df):
    nabo_values = ['E75', '73W', '73H', '73J', 'E95', '319', '320', '321', '221', '223', '290', '295', '318', '32N', '32Q', '75T', '7M8', 'CR9', 'E70', 'E7W', 'E90', 'ER4', '734', '32A', '7M9']
    widebody_values = ['74E', '744', '74H', '763', '764', '772', '77W', '77L', '788', '789', '781', '332', '333', '359', '351', '388', '339', '76W']
    ake_containerized_values = widebody_values + ['321']  # A321 toegevoegd aan containerized_values
    akh_containerized_values = ['320']  # AKH container only for A320

    df['IN_BODY_TYPE'] = df['IN_AC_IATA'].apply(lambda x: 'NABO' if pd.notnull(x) and x in nabo_values else ('WIBO' if pd.notnull(x) and x in widebody_values else ''))
    df['OUT_BODY_TYPE'] = df['OUT_AC_IATA'].apply(lambda x: 'NABO' if pd.notnull(x) and x in nabo_values else ('WIBO' if pd.notnull(x) and x in widebody_values else ''))

    df['IN_BAGGAGE_TYPE'] = df['IN_AC_IATA'].apply(lambda x: 'AKE Container' if pd.notnull(x) and x in ake_containerized_values else ('AKH Container' if pd.notnull(x) and x in akh_containerized_values else ('Bulk' if pd.notnull(x) else '')))
    df['OUT_BAGGAGE_TYPE'] = df['OUT_AC_IATA'].apply(lambda x: 'AKE Container' if pd.notnull(x) and x in ake_containerized_values else ('AKH Container' if pd.notnull(x) and x in akh_containerized_values else ('Bulk' if pd.notnull(x) else '')))

    return df

df = add_body_baggage_flight_type_columns(df)

# Nieuwe kolom ODD_SIZED toevoegen
odd_sized_keywords = [
    'BB Sorteerband 16', 'BB Sorteerband 17', 'D Bagtrax Rood Sorteerband 90',
    'D Bagtrax Blauw Sorteerband 90', 'T2 MSW exit naar Sorteerband 70',
    'T2 MSO exit naar Sorteerband 70', 'T3 Sorteerband 25 entry van SWH',
    'T3 Sorteerband 25 entry van SWL', 'T1 Sorter hoog Sorteerband 50',
    'T1 Sorter laag Sorteerband 50'
]

def determine_odd_sized(exit_name):
    if pd.notnull(exit_name) and any(keyword in exit_name for keyword in odd_sized_keywords):
        return 'Odd sized'
    else:
        return 'Normal'

df['ODD_SIZED'] = df['EXIT_NAME'].apply(determine_odd_sized)

# Add ramp clusters
def add_ramp_clusters(df):
    ramp_clusters = {
        'B_platform': ['A' + str(i).zfill(2) for i in range(91)] + ['B' + str(i).zfill(2) for i in range(51, 99)],
        'B_pier': ['B' + str(i).zfill(2) for i in range(13, 37)],
        'C_stem': ['C' + str(i).zfill(2) for i in range(4, 10)],
        'C_head': ['C' + str(i).zfill(2) for i in range(10, 19)],
        'D_stem': ['D' + str(i).zfill(2) for i in range(2, 9)],
        'D_fork_south': ['D' + str(i).zfill(2) for i in range(10, 32)],
        'D_fork_north': ['D' + str(i).zfill(2) for i in range(41, 100)],
        'E_stem': ['E' + str(i).zfill(2) for i in range(2, 8)],
        'E_head': ['E' + str(i).zfill(2) for i in range(8, 100)],
        'F_pier': ['F' + str(i).zfill(2) for i in range(1, 100)],
        'G_stem': ['G' + str(i).zfill(2) for i in range(2, 7)],
        'G_head': ['G' + str(i).zfill(2) for i in range(7, 81)]
                  + ['H' + str(i).zfill(2) for i in range(1, 10)]
                  + ['H' + str(i) for i in range(10, 99)]
                  + ['M' + str(i).zfill(2) for i in range(1, 8)]
                  + ['J' + str(i).zfill(2) for i in range(1, 99)],
    }
```

```
    def determine_ramp_cluster(ramp, ramp_clusters):
        for cluster, ramps in ramp_clusters.items():
            if ramp in ramps:
                return cluster
        return ''  # Maak de cel leeg als er geen match is

    df['IN_RAMP_CLUSTER'] = df['IN_RAMP'].apply(lambda x: determine_ramp_cluster(x, ramp_clusters))
    df['OUT_RAMP_CLUSTER'] = df['OUT_RAMP'].apply(lambda x: determine_ramp_cluster(x, ramp_clusters))

    return df

df = add_ramp_clusters(df)

# Verdeel de data in een set van 2024
df_2024 = df.loc[pd.to_datetime(df['DATETIME_OF_ENTRY'], errors='coerce').dt.year == 2024]

# Controleer rijen per jaar
print(f"Number of rows in 2024 data: {df_2024.shape[0]}")

# Split datetime columns into date and time
def split_datetime_columns(df):
    for col in ['DATETIME_OF_ENTRY', 'DATETIME_OF_EXIT', 'SIBT', 'AIBT', 'SOBT', 'AOBT', 'DATETIME_STORE_ENTRY', 'DATETIME_IN_STORE', 'DATETIME_STORE_EXIT']:
        if col in df.columns:
            df[f'DATE_{col}'] = df[col].dt.date
            df[f'TIME_{col}'] = df[col].dt.time
    return df

df_2024 = split_datetime_columns(df_2024)

# Exporteer de gesplitste data naar CSV-bestanden
documents_path = 'C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\'
df_2024_path = os.path.join(documents_path, 'Baggage_data_2024.csv')

df_2024.to_csv(df_2024_path, index=False)

print(f"Data uit 2024 is gexporteerd naar {df_2024_path}")

# Controleer rijen in gexporteerde bestanden
exported_2024 = pd.read_csv(df_2024_path)

print()
print(f"Number of rows in exported data file for 2024: {exported_2024.shape[0]}")

end_time = time.time()
elapsed_time = end_time - start_time
print(f"Cel runtime: {elapsed_time:.2f} seconden")
```

Listing 1: Cell 1: Data Loading and Initial Processing

Appendix A2: Data Duplication for Testing

This section duplicates every 5th row in the dataset to create an expanded version, useful for testing scenarios with increased data volume. The code reads the original file, loops through each row to add it to a list, and duplicates every 5th row. The result is saved as a new CSV file with the expanded dataset, and the code calculates the percentage increase in the row count to verify the duplication process.

```
import pandas as pd

# Laad het originele bestand 'Baggage_data_2024.csv' in
df_exported = pd.read_csv('C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\
    ↪ Baggage_data_2024.csv')

# Totaal aantal rijen in het originele bestand
aantal_rijen_origineel = len(df_exported)

# Lijst om de uitgebreide dataset op te slaan
rows = []

# Loop door elke rij en dupliceer elke 5e rij
for i, row in df_exported.iterrows():
    rows.append(row)  # Voeg de originele rij toe
    if (i + 1) % 5 == 0:
        rows.append(row)  # Dupliceer de rij elke 5 rijen

# Maak een nieuw DataFrame met de uitgebreide dataset
expanded_df = pd.DataFrame(rows)

# Totaal aantal rijen in het uitgebreide bestand
aantal_rijen_uitgebreid = len(expanded_df)

# Bereken het percentage toename
percentage_toename = (aantal_rijen_uitgebreid / aantal_rijen_origineel) * 100

# Sla de uitgebreide dataset op in een nieuw CSV-bestand
expanded_df.to_csv('C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\
    ↪ Baggage_data_2024_extended.csv', index=False)

# Print de resultaten
print(f"Origineel aantal rijen: {aantal_rijen_origineel}")
print(f"Uitgebreid aantal rijen: {aantal_rijen_uitgebreid}")
print(f"Het uitgebreide bestand bevat {percentage_toename:.2f}% van het oorspronkelijke aantal rijen."
    ↪ )
```

Listing 2: Cell 2: Data Duplication for Testing

Appendix A3: ID Assignment and Filtering Odd-Sized Baggage

This code section loads the expanded dataset, removes rows labeled as "Odd sized" baggage, and ensures each baggage item has a unique identifier. If the `Koffer_ID` column does not already exist, it assigns a unique ID based on the row index. The updated dataset is saved, and the total number of baggage items remaining is printed for verification.

```python
import pandas as pd
import time

# Start timer
start_time = time.time()

# Laad de originele datasets
baggage_data_2024_path = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\
    ↪ Baggage_data_2024_extended.csv"
df_2024 = pd.read_csv(baggage_data_2024_path)

# Verwijder rijen met Odd Sized baggage
df_2024 = df_2024[df_2024['ODD_SIZED'] != 'Odd sized']

# Controleer of 'Koffer_ID' al is toegevoegd; zo niet, voeg ze toe
if 'Koffer_ID' not in df_2024.columns:
    df_2024['Koffer_ID'] = df_2024.index

# Exporteer de datasets opnieuw met de nieuwe ID-kolom (indien nodig)
df_2024.to_csv(baggage_data_2024_path, index=False)

# Tel het aantal koffers in de dataset na bewerking
total_bags_2024 = len(df_2024)
print(f"Controle voltooid: Unieke ID's zijn toegevoegd aan elke koffer in de datasets.")
print(f"Totaal aantal koffers in 2024 dataset (na verwijderen van Odd Sized bags): {total_bags_2024}")

# End timer
end_time = time.time()
elapsed_time = end_time - start_time
print(f"Cel runtime: {elapsed_time:.2f} seconden")
```

Listing 3: Cell 3: ID Assignment and Filtering Odd-Sized Baggage

Appendix A4: Simulation of Baggage Unloading and Container Assignment

This code simulates the unloading of baggage items and their assignment to containers using the SimPy library for discrete event simulation. The code loads a dataset, processes each baggage item's entry and scheduled times, and calculates connection times. Based on each baggage type, items are grouped into containers (or karren), prioritizing those with lower temperatures when filling capacity. A unique ID is generated for each container, and attributes such as container type, temperature category, and unloading times are recorded. The results, including baggage-to-container relations and container details, are saved as separate CSV files.

```python
import pandas as pd
import simpy
import numpy as np
import random
from datetime import timedelta
import time

# Start timer
start_time = time.time()

# Random seed for reproducibility
random.seed(42)

# Laad de dataset
file_path = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\Baggage_data_2024_extended.csv"
df = pd.read_csv(file_path)

# Converteer de tijden naar datetime
df['DATETIME_OF_ENTRY'] = pd.to_datetime(df['DATETIME_OF_ENTRY'])
df['AIBT'] = pd.to_datetime(df['DATE_AIBT'] + ' ' + df['TIME_AIBT'])
df['SOBT'] = pd.to_datetime(df['DATE_SOBT'] + ' ' + df['TIME_SOBT'])
df['SIBT'] = pd.to_datetime(df['DATE_SIBT'] + ' ' + df['TIME_SIBT'])

# Bereken de connectietijd in minuten
df['connectietijd'] = (df['SOBT'] - df['SIBT']).dt.total_seconds() / 60.0

# Voeg 'Latest_time_bag' toe (SOBT min 25 minuten)
df['Latest_time_bag'] = df['SOBT'] - pd.Timedelta(minutes=25)

# Initialiseer de lijsten voor de resultaten
koffer_to_container = []
containers = []

# SimPy omgeving en bagageband resource
env = simpy.Environment()

# Initialiseer een globale container teller om unieke ID's te maken
container_counter = 0

# Functie om koffers uit te laden en te verwerken
def unload_baggage(env, flight_data, baggage_type, items_per_container, in_ramp_cluster, flight_code, aibt):
    global container_counter  # Gebruik een globale teller voor unieke container-ID's
    # Resource voor bagageband met beperkte capaciteit (bijv. 1 band tegelijk actief)
    bagageband = simpy.Resource(env, capacity=1)

    with bagageband.request() as request:
        yield request

        # Wachttijd voor het begin van uitladen
        delay = random.uniform(1620, 1680)
        unload_start_time = aibt + timedelta(seconds=delay)
        yield env.timeout(delay)  # Simuleer tijd totdat het uitladen begint

```

```
        # Karren vormen per bagagetemperatuur
        karren = {'ShoCon': [], 'Cold Transfer': [], 'Super Cold Transfer': []}
        for temperatuur, temp_data in flight_data.groupby('BAGGAGE_TEMPERATURE'):
            for _, bagage in temp_data.iterrows():
                # Simuleer de tijd die het kost om elke koffer uit te laden
                if baggage_type == 'Bulk':
                    unload_time_variability = random.uniform(6, 7.5)  # Tijd voor Bulk bagage
                else:
                    unload_time_variability = random.triangular(60, 100, 80)  # Tijd voor containers
                yield env.timeout(unload_time_variability)  # Tijd voor het uitladen van een enkele koffer
                karren[temperatuur].append(bagage)

        # Vul de karren/containers met koffers en vul de resterende ruimte aan met lagere temperatuur koffers
        for temp, bags in karren.items():
            while bags:
                kar = bags[:items_per_container]  # Vul een container
                bags = bags[items_per_container:]  # Verwijder de koffers die zijn toegewezen aan de container
                center_of_gravity = min(kar, key=lambda x: x['connectietijd'])['EXIT_AREA'] if temp == 'ShoCon' else pd.Series([b['EXIT_AREA'] for b in kar]).mode()[0]

                # Gebruik een unieke container-ID door de teller te gebruiken
                container_id = f"{flight_code}_{aibt.strftime('%Y%m%d%H%M%S')}_{temp}_{container_counter}"
                container_counter += 1  # Verhoog de teller na elke container creatie

                # Controleer of de kar/container vol zit, zo niet, vul deze aan met Cold Transfer of Super Cold Transfer
                if len(kar) < items_per_container and temp == 'ShoCon':
                    remaining_capacity = items_per_container - len(kar)
                    cold_transfer_bags = karren['Cold Transfer'][:remaining_capacity]
                    kar.extend(cold_transfer_bags)
                    karren['Cold Transfer'] = karren['Cold Transfer'][remaining_capacity:]  # Update de Cold Transfer lijst

                if len(kar) < items_per_container and temp == 'Cold Transfer':
                    remaining_capacity = items_per_container - len(kar)
                    super_cold_transfer_bags = karren['Super Cold Transfer'][:remaining_capacity]
                    kar.extend(super_cold_transfer_bags)
                    karren['Super Cold Transfer'] = karren['Super Cold Transfer'][remaining_capacity:]  # Update de Super Cold Transfer lijst

                # Tijdsopbouw voor het einde van het uitladen
                if baggage_type == 'Bulk':
                    # Voor Bulk bagage (30 koffers per kar)
                    unload_end_time = unload_start_time + timedelta(seconds=30 * random.uniform(6, 7.5))
                else:
                    # Voor containers
                    unload_end_time = unload_start_time + timedelta(seconds=random.triangular(60, 100, 80))

                # Voeg koffer-naar-container relaties toe
                for koffer in kar:
                    koffer_to_container.append({
                        'Koffer_ID': koffer['Koffer_ID'],
                        'Container_Zwaartepunt': center_of_gravity,
                        'ContainerID': container_id,
                        'Latest_time_bag': koffer['Latest_time_bag']
                    })

```

```python
                # Voeg containers toe met de nieuwe kolommen 'tijd_eind_container' en 'baggage_type'
                containers.append({
                    'tijd': unload_start_time.strftime('%Y-%m-%d %H:%M:%S'),
                    'dag': unload_start_time.strftime('%d-%m-%Y'),
                    'vluchtcode': flight_code,
                    'aantal_koffers': len(kar),
                    'IN_RAMP_CLUSTER': in_ramp_cluster,
                    'KAR_TEMPERATURE': temp,
                    'center_of_gravity': center_of_gravity,
                    'ContainerID': container_id,
                    'Latest_time_container': min([b['Latest_time_bag'] for b in kar]),
                    'tijd_eind_container': unload_end_time.strftime('%Y-%m-%d %H:%M:%S'),  # Voeg tijd_eind_container toe
                    'baggage_type': baggage_type  # Voeg baggage_type toe
                })

# Simulatieproces voor meerdere dagen en vluchten
start_date = "2024-04-13"
end_date = "2024-06-30"
capacity_per_baggage_type = {'Bulk': 30, 'AKE Container': 38, 'AKH Container': 28}

for single_date in pd.date_range(start=start_date, end=end_date):
    specific_date = single_date.strftime("%Y-%m-%d")
    df_day = df[df['DATE_DATETIME_OF_ENTRY'] == specific_date]

    if df_day.empty:
        print(f"Geen data voor {specific_date}, overslaan...")
        continue

    for (flight_code, aibt), flight_data in df_day.groupby(['IN_FLIGHT_DESIGNATOR', 'AIBT']):
        baggage_type = flight_data['IN_BAGGAGE_TYPE'].iloc[0].strip()
        items_per_container = capacity_per_baggage_type.get(baggage_type, 30)
        in_ramp_cluster = flight_data['IN_RAMP_CLUSTER'].iloc[0]

        # Start SimPy proces voor elke vlucht
        env.process(unload_baggage(env, flight_data, baggage_type, items_per_container, in_ramp_cluster, flight_code, aibt))

# SimPy run
env.run()

# Exporteer koffer-to-container data met 'Latest_time_bag'
koffer_container_output_path = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\koffer_to_container_output.csv"
koffer_container_df = pd.DataFrame(koffer_to_container)
koffer_container_df.to_csv(koffer_container_output_path, index=False)

# Exporteer container data met 'tijd_eind_container' en 'baggage_type'
containers_output_path = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\containers_output.csv"
containers_df = pd.DataFrame(containers)

# Voeg de Koffer_ID kolom toe aan de containers_df door te groeperen op ContainerID
koffer_groepering = koffer_container_df.groupby('ContainerID')['Koffer_ID'].apply(lambda x: ','.join(x.astype(str))).reset_index()
containers_df = containers_df.merge(koffer_groepering, on='ContainerID', how='left')

containers_df.to_csv(containers_output_path, index=False)

# End timer
end_time = time.time()
elapsed_time = end_time - start_time
print(f"Cel runtime: {elapsed_time:.2f} seconden")
```

Listing 4: Cell 4: Simulation of Baggage Unloading and Container Assignment

Appendix A5: Calculation of Arrival Times and Capacity-Based Baggage Processing

This code calculates the arrival times of containers at airport infeed points based on predefined travel times between ramps and infeed areas. After calculating the arrival times, the code checks the capacity limits of each infeed point per quarter hour. Baggage items are processed according to their temperature categories, prioritizing ShoCon items first, followed by Cold Transfer and Super Cold Transfer, to avoid exceeding infeed capacity. Remaining baggage beyond the capacity limit is held in a queue for the next processing interval. The results, including processed arrival times and baggage quantities per infeed area, are saved in a CSV file.

```python
import pandas as pd
import numpy as np
from datetime import timedelta
import time

# Start timer using time.time() to ensure correct time tracking
start_time = time.time()

# Laad de containers_output dataset
containers_output_path = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\containers_output.csv"
containers_df = pd.read_csv(containers_output_path)

# Reistijden tussen IN_RAMP_CLUSTER en Infeed Areas (in seconden)
travel_times = {
    ('B_platform', 'C'): 451, ('B_platform', 'D'): 441, ('B_platform', 'W'): 549, ('B_platform', 'Z'): 271, ('B_platform', 'E'): 461,
    ('B_pier', 'C'): 336, ('B_pier', 'D'): 325, ('B_pier', 'W'): 433, ('B_pier', 'Z'): 155, ('B_pier', 'E'): 345,
    ('C_stem', 'C'): 241, ('C_stem', 'D'): 198, ('C_stem', 'W'): 338, ('C_stem', 'Z'): 136, ('C_stem', 'E'): 250,
    ('C_head', 'C'): 273, ('C_head', 'D'): 189, ('C_head', 'W'): 370, ('C_head', 'Z'): 168, ('C_head', 'E'): 282,
    ('D_stem', 'C'): 190, ('D_stem', 'D'): 93, ('D_stem', 'W'): 287, ('D_stem', 'Z'): 182, ('D_stem', 'E'): 200,
    ('D_fork_south', 'C'): 257, ('D_fork_south', 'D'): 76, ('D_fork_south', 'W'): 354, ('D_fork_south', 'Z'): 248, ('D_fork_south', 'E'): 266,
    ('D_fork_north', 'C'): 262, ('D_fork_north', 'D'): 84, ('D_fork_north', 'W'): 360, ('D_fork_north', 'Z'): 254, ('D_fork_north', 'E'): 213,
    ('E_stem', 'C'): 124, ('E_stem', 'D'): 202, ('E_stem', 'W'): 221, ('E_stem', 'Z'): 207, ('E_stem', 'E'): 88,
    ('E_head', 'C'): 229, ('E_head', 'D'): 248, ('E_head', 'W'): 326, ('E_head', 'Z'): 312, ('E_head', 'E'): 19,
    ('F_pier', 'C'): 177, ('F_pier', 'D'): 299, ('F_pier', 'W'): 191, ('F_pier', 'Z'): 301, ('F_pier', 'E'): 227,
    ('G_stem', 'C'): 199, ('G_stem', 'D'): 322, ('G_stem', 'W'): 136, ('G_stem', 'Z'): 324, ('G_stem', 'E'): 250,
    ('G_head', 'C'): 273, ('G_head', 'D'): 395, ('G_head', 'W'): 210, ('G_head', 'Z'): 397, ('G_head', 'E'): 324
}

# Capaciteit per infeed area per kwartier (aangenomen 1 uur = 4 kwartieren)
capacities_per_hour = {'D': 4500, 'W': 900, 'Z': 2000, 'E': 3600}
capacities_per_quarter = {key: val / 4 for key, val in capacities_per_hour.items()}
total_capacity_per_hour = 11000
total_capacity_per_quarter = 2750

# Functie om koffers te verwerken en te zorgen dat ze niet boven de capaciteit gaan
def process_baggage_with_capacity_limit(arrival_df, capacity_per_quarter):
    result = []
    waiting_bags = {'ShoCon': [], 'Cold Transfer': [], 'Super Cold Transfer': []}  # Wachtrijen voor koffers per type

    # Groepeer de data per kwartier
```

```
    for time, group in arrival_df.groupby('aankomst_tijd'):
        total_processed = 0  # Houd bij hoeveel koffers al verwerkt zijn

        # Prioriteit: eerst ShoCon, dan Cold Transfer, dan Super Cold Transfer
        for temp_type in ['ShoCon', 'Cold Transfer', 'Super Cold Transfer']:
            temp_bags = group[group['KAR_TEMPERATURE'] == temp_type]['aantal_koffers'].sum() + sum(waiting_bags[temp_type])
            if total_processed + temp_bags <= capacity_per_quarter:
                # Alle koffers kunnen verwerkt worden
                result.append((time, temp_bags, temp_type))
                total_processed += temp_bags
                waiting_bags[temp_type] = []  # Leeg de wachtrij voor deze temperatuur
            else:
                # Verwerk koffers binnen de capaciteit en zet de rest in de wacht
                if total_processed < capacity_per_quarter:
                    processed_bags = capacity_per_quarter - total_processed
                    remaining_bags = temp_bags - processed_bags
                    result.append((time, processed_bags, temp_type))
                    waiting_bags[temp_type] = [remaining_bags]
                    total_processed = capacity_per_quarter
                else:
                    # Geen capaciteit meer over, zet alle koffers in de wacht
                    waiting_bags[temp_type].append(temp_bags)

    return result

# Bereken aankomsttijden bij infeed points
arrival_times = []

for _, row in containers_df.iterrows():
    vertrekpunt = row['IN_RAMP_CLUSTER']
    bestemming = row['center_of_gravity']
    if (vertrekpunt, bestemming) in travel_times:
        travel_time = travel_times[(vertrekpunt, bestemming)]
        start_time_trein = pd.to_datetime(row['tijd_eind_container'])
        arrival_time = start_time_trein + timedelta(seconds=travel_time)
        # Voeg de ontbrekende kolommen toe
        arrival_times.append((arrival_time, row['aantal_koffers'], row['KAR_TEMPERATURE'], bestemming))

# Creer een DataFrame voor de aankomsttijden
arrival_df = pd.DataFrame(arrival_times, columns=['aankomst_tijd', 'aantal_koffers', 'KAR_TEMPERATURE', 'bestemming'])
arrival_df['aankomst_tijd'] = arrival_df['aankomst_tijd'].dt.ceil('15min')

# Verwerk koffers met capaciteitslimieten
results = []

for infeed_point in capacities_per_quarter.keys():
    filtered_df = arrival_df[arrival_df['bestemming'] == infeed_point]
    capacity_per_quarter = capacities_per_quarter[infeed_point]
    processed_bags = process_baggage_with_capacity_limit(filtered_df, capacity_per_quarter)
    # Zorg ervoor dat je alle benodigde kolommen meeneemt in de resultaten
    results.extend([(infeed_point, time, bags, temp, time.date())
                    for time, bags, temp in processed_bags])

# Maak een DataFrame van de verwerkte data, exclusief de ongewenste kolommen
processed_df = pd.DataFrame(results, columns=['infeed_point', 'aankomst_tijd', 'aantal_koffers', 'KAR_TEMPERATURE', 'dag'])

# Sla de resultaten op in een CSV-bestand
processed_df.to_csv("C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\aankomsttijd_containers.csv", index=False)
```

```python

# End timer using time.time()
end_time = time.time()

# Calculate the elapsed time in seconds
elapsed_time = end_time - start_time

print(f"Cel runtime: {elapsed_time:.2f} seconden")
```

Listing 5: Cell 5: Calculation of Arrival Times and Capacity-Based Baggage Processing

Appendix A6: Processing and Grouping Known Arrivals

This code processes known baggage arrival data by filtering and grouping entries by area and exact entry time. After loading the dataset, irrelevant rows are removed, and specific values in the `ENTRY_AREA` column are standardized. A combined datetime column is created, and the data is grouped by `ENTRY_AREA` and entry time to count baggage arrivals per time interval. The final grouped data, containing counts of baggage items per entry area and timestamp, is exported to a CSV file.

```python
import pandas as pd
import os

# Laad de dataset Baggage_data_2024
baggage_data_path = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\Baggage_data_2024.csv"
df = pd.read_csv(baggage_data_path)

# Filter alleen de relevante kolommen
df_known = df[['ENTRY_AREA', 'DATE_DATETIME_OF_ENTRY', 'TIME_DATETIME_OF_ENTRY']].copy()

# Verwijder rijen waar ENTRY_AREA 'C' is
df_known = df_known[df_known['ENTRY_AREA'] != 'C']

# Vervang UQE door E en TSD door D in ENTRY_AREA
df_known['ENTRY_AREA'] = df_known['ENTRY_AREA'].replace({'UQE': 'E', 'TSD': 'D'})

# Converteer DATE_DATETIME_OF_ENTRY en TIME_DATETIME_OF_ENTRY naar datetime
df_known['DATE_DATETIME_OF_ENTRY'] = pd.to_datetime(df_known['DATE_DATETIME_OF_ENTRY'], errors='coerce')
df_known['TIME_DATETIME_OF_ENTRY'] = pd.to_datetime(df_known['TIME_DATETIME_OF_ENTRY'], errors='coerce')

# Maak een nieuwe kolom voor de volledige datetime
df_known['DATETIME_OF_ENTRY'] = df_known['DATE_DATETIME_OF_ENTRY'] + pd.to_timedelta(df_known['TIME_DATETIME_OF_ENTRY'].dt.strftime('%H:%M:%S'))

# Groepeer per ENTRY_AREA en exacte tijd
df_grouped = df_known.groupby(['ENTRY_AREA', 'DATETIME_OF_ENTRY']).size().reset_index(name='aantal_koffers')

# Exporteer het bekende aankomstenbestand naar CSV
known_arrivals_output_path = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\Known_arrivals_measured.csv"
df_grouped.to_csv(known_arrivals_output_path, index=False)

print(f"Het bestand 'Known_arrivals_measured.csv' is succesvol gexporteerd naar {known_arrivals_output_path}.")
```

Listing 6: Cell 6: Processing and Grouping Known Arrivals

Appendix A7: Visualization of Simulated vs. Actual Baggage Occupancy

This code visualizes the comparison between simulated and actual baggage occupancy per 15-minute interval. It first calculates the average baggage count for each interval across multiple days for both datasets. Two plots are generated to compare simulated and actual occupancy: a side-by-side bar plot and a line plot, both displaying the average number of bags over all days. These visualizations highlight any differences in occupancy trends, providing insights into the simulation's accuracy against real data.

```
import pandas as pd
import matplotlib.pyplot as plt
import matplotlib.dates as mdates

# Load the processed data for simulated arrivals
processed_df = pd.read_csv("C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\aankomsttijd_containers.csv")
processed_df['aankomst_tijd'] = pd.to_datetime(processed_df['aankomst_tijd'])

# Groepeer per dag en per 15-minuten interval voor gesimuleerde data
processed_df['time_only'] = processed_df['aankomst_tijd'].dt.ceil('15T').dt.time
processed_df['date'] = processed_df['aankomst_tijd'].dt.date
daily_occupancy_simulated = processed_df.groupby(['date', 'time_only'])['aantal_koffers'].sum().reset_index()

# Bereken het gemiddelde per 15 minuten interval over alle dagen
average_occupancy_simulated = daily_occupancy_simulated.groupby('time_only')['aantal_koffers'].mean().reset_index()
average_occupancy_simulated['time_only'] = pd.to_datetime(average_occupancy_simulated['time_only'].astype(str))

# Load the known arrivals data (actual data)
df_known = pd.read_csv("C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\Known_arrivals_measured.csv")
df_known['DATETIME_OF_ENTRY'] = pd.to_datetime(df_known['DATETIME_OF_ENTRY'])

# Groepeer per dag en per 15-minuten interval voor actuele data
df_known['time_only'] = df_known['DATETIME_OF_ENTRY'].dt.ceil('15T').dt.time
df_known['date'] = df_known['DATETIME_OF_ENTRY'].dt.date
daily_occupancy_actual = df_known.groupby(['date', 'time_only'])['aantal_koffers'].sum().reset_index()

# Bereken het gemiddelde per 15 minuten interval over alle dagen
average_occupancy_actual = daily_occupancy_actual.groupby('time_only')['aantal_koffers'].mean().reset_index()
average_occupancy_actual['time_only'] = pd.to_datetime(average_occupancy_actual['time_only'].astype(str))

# Merge both datasets on 'time_only' for comparison
merged_data = pd.merge(average_occupancy_simulated, average_occupancy_actual, on='time_only', suffixes=('_simulated', '_actual'))

# Create side-by-side bar plots for the average over all days
plt.figure(figsize=(12, 6))
plt.bar(merged_data['time_only'] - pd.Timedelta(minutes=3.75), merged_data['aantal_koffers_simulated'], width=0.004, label='Simulated', color='green')
plt.bar(merged_data['time_only'] + pd.Timedelta(minutes=3.75), merged_data['aantal_koffers_actual'], width=0.004, label='Actual', color='blue')

plt.gca().xaxis.set_major_formatter(mdates.DateFormatter('%H:%M'))
plt.gca().xaxis.set_major_locator(mdates.HourLocator(interval=1))
plt.xlabel('Time of Day')
plt.ylabel('Number of Bags')
plt.title(f'Comparison of Simulated and Actual Average Baggage Occupancy per 15-Minute Interval')
plt.xticks(rotation=45)
plt.grid(axis='y', linestyle='--')
plt.legend()
```

```
plt.show()

# Create line plot for the average over all days
plt.figure(figsize=(12, 6))

plt.plot(merged_data['time_only'], merged_data['aantal_koffers_simulated'], label='Simulated', marker=
    ↪ 'o', color='green')
plt.plot(merged_data['time_only'], merged_data['aantal_koffers_actual'], label='Actual', marker='o',
    ↪ color='blue')

plt.gca().xaxis.set_major_formatter(mdates.DateFormatter('%H:%M'))
plt.gca().xaxis.set_major_locator(mdates.HourLocator(interval=1))
plt.xlabel('Time of Day')
plt.ylabel('Number of Bags')
plt.xticks(rotation=45)
plt.grid(axis='y', linestyle='--')
plt.legend()

plt.show()
```

Listing 7: Cell 7: Visualization of Simulated vs. Actual Baggage Occupancy

Appendix A8: Daily Comparison of Simulated vs. Actual Baggage Occupancy for a Specific Date

This code compares simulated and actual baggage occupancy on a specific date, chosen as June 30, 2024. After loading and filtering both datasets for the selected date, baggage counts are grouped into 15-minute intervals. Two visualizations are generated: a side-by-side bar plot and a line plot, displaying the occupancy trends for the chosen date. These plots allow for a direct comparison of simulation accuracy for a single day, showing potential discrepancies between simulated and actual baggage flow patterns.

```python
import pandas as pd
import matplotlib.pyplot as plt
import matplotlib.dates as mdates

# Kies de specifieke datum
chosen_date = '2024-06-30'

# Load the processed data for simulated arrivals
processed_df = pd.read_csv("C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\aankomsttijd_containers.csv")
processed_df['aankomst_tijd'] = pd.to_datetime(processed_df['aankomst_tijd'])

# Filter op de gekozen datum
processed_df = processed_df[processed_df['aankomst_tijd'].dt.date == pd.to_datetime(chosen_date).date()]

# Groepeer per 15-minuten interval voor gesimuleerde data
processed_df['time_only'] = processed_df['aankomst_tijd'].dt.ceil('15T').dt.time
average_occupancy_simulated = processed_df.groupby('time_only')['aantal_koffers'].sum().reset_index()
average_occupancy_simulated['time_only'] = pd.to_datetime(average_occupancy_simulated['time_only'].astype(str))

# Load the known arrivals data (actual data)
df_known = pd.read_csv("C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\Known_arrivals_measured.csv")
df_known['DATETIME_OF_ENTRY'] = pd.to_datetime(df_known['DATETIME_OF_ENTRY'])

# Filter op de gekozen datum
df_known = df_known[df_known['DATETIME_OF_ENTRY'].dt.date == pd.to_datetime(chosen_date).date()]

# Groepeer per 15-minuten interval voor actuele data
df_known['time_only'] = df_known['DATETIME_OF_ENTRY'].dt.ceil('15T').dt.time
average_occupancy_actual = df_known.groupby('time_only')['aantal_koffers'].sum().reset_index()
average_occupancy_actual['time_only'] = pd.to_datetime(average_occupancy_actual['time_only'].astype(str))

# Merge both datasets on 'time_only' for comparison
merged_data = pd.merge(average_occupancy_simulated, average_occupancy_actual, on='time_only', suffixes=('_simulated', '_actual'))

# Create side-by-side bar plots for the chosen date
plt.figure(figsize=(12, 6))
plt.bar(merged_data['time_only'] - pd.Timedelta(minutes=3.75), merged_data['aantal_koffers_simulated'], width=0.004, label='Simulated', color='green')
plt.bar(merged_data['time_only'] + pd.Timedelta(minutes=3.75), merged_data['aantal_koffers_actual'], width=0.004, label='Actual', color='blue')

plt.gca().xaxis.set_major_formatter(mdates.DateFormatter('%H:%M'))
plt.gca().xaxis.set_major_locator(mdates.HourLocator(interval=1))
plt.xlabel('Time of Day')
plt.ylabel('Number of Bags')
plt.title(f'Comparison of Simulated and Actual Baggage Occupancy on {chosen_date}')
plt.xticks(rotation=45)
plt.grid(axis='y', linestyle='--')
plt.legend()
```

```
plt.show()

# Create line plot for the chosen date
plt.figure(figsize=(12, 6))

plt.plot(merged_data['time_only'], merged_data['aantal_koffers_simulated'], label='Simulated', marker='o', color='green')
plt.plot(merged_data['time_only'], merged_data['aantal_koffers_actual'], label='Actual', marker='o', color='blue')

plt.gca().xaxis.set_major_formatter(mdates.DateFormatter('%H:%M'))
plt.gca().xaxis.set_major_locator(mdates.HourLocator(interval=1))
plt.xlabel('Time of Day')
plt.ylabel('Number of Bags')
plt.xticks(rotation=45)
plt.grid(axis='y', linestyle='--')
plt.legend()

plt.show()
```
Listing 8: Cell 8: Daily Comparison of Simulated vs. Actual Baggage Occupancy for a Specific Date

Appendix A9: Error Metrics Calculation for Simulated vs. Actual Baggage Occupancy

This code calculates error metrics to assess the accuracy of the simulated baggage occupancy against actual data. After loading both datasets and grouping baggage counts by 15-minute intervals, the data is merged for comparison. Three metrics are calculated: Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), and Mean Absolute Error (MAE). These metrics quantify the deviation of simulated results from actual data, providing insight into the performance and reliability of the simulation.

```
import pandas as pd
import numpy as np
from sklearn.metrics import mean_squared_error

# Laad de datasets met gesimuleerde en actuele gegevens
processed_df = pd.read_csv("C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\aankomsttijd_containers.csv")
df_known = pd.read_csv("C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\Known_arrivals_measured.csv")

# Zorg ervoor dat de datums correct worden geparsed
processed_df['aankomst_tijd'] = pd.to_datetime(processed_df['aankomst_tijd'])
df_known['DATETIME_OF_ENTRY'] = pd.to_datetime(df_known['DATETIME_OF_ENTRY'])

# Bereid de datasets voor en groepeer de gegevens per tijdstip
processed_df['time_only'] = processed_df['aankomst_tijd'].dt.ceil('15T').dt.time
df_known['time_only'] = df_known['DATETIME_OF_ENTRY'].dt.ceil('15T').dt.time

average_occupancy_simulated = processed_df.groupby('time_only')['aantal_koffers'].sum().reset_index()
average_occupancy_actual = df_known.groupby('time_only')['aantal_koffers'].sum().reset_index()

# Merge beide datasets voor de vergelijking
merged_data = pd.merge(average_occupancy_simulated, average_occupancy_actual, on='time_only', suffixes=('_simulated', '_actual'))

# Bereken RMSE
rmse = np.sqrt(mean_squared_error(merged_data['aantal_koffers_actual'], merged_data['aantal_koffers_simulated']))
print(f'Root Mean Square Error (RMSE): {rmse}')

# Bereken MAPE
def mean_absolute_percentage_error(y_true, y_pred):
    return np.mean(np.abs((y_true - y_pred) / y_true)) * 100

mape = mean_absolute_percentage_error(merged_data['aantal_koffers_actual'], merged_data['aantal_koffers_simulated'])
print(f'Mean Absolute Percentage Error (MAPE): {mape:.2f}%')

# Bereken gemiddelde absolute fout (MAE) als aanvullende metriek
mae = np.mean(np.abs(merged_data['aantal_koffers_actual'] - merged_data['aantal_koffers_simulated']))
print(f'Mean Absolute Error (MAE): {mae}')
```

Listing 9: Cell 9: Error Metrics Calculation for Simulated vs. Actual Baggage Occupancy

Appendix A10: Forecasting Baggage Occupancy Using Auto ARIMA

This code applies the Auto ARIMA model to forecast baggage occupancy rates in 15-minute intervals for June 30, 2024. It begins by preparing and aggregating historical data up to June 29, 2024, creating a training set with 15-minute interval counts. Auto ARIMA is then used to determine optimal parameters, taking into account daily seasonality (96 intervals per day). Finally, the trained model generates a forecast for June 30, providing occupancy predictions in 15-minute segments for the entire day.

```python
import pandas as pd
import numpy as np
from pmdarima import auto_arima
import warnings
warnings.filterwarnings('ignore')

# Dataset voorbereiden zoals eerder beschreven
file_path = 'C:\\Users\\Nijdam_B\\Documents\\Baggage_data_2024.csv'
data = pd.read_csv(file_path)
data['SIBT'] = pd.to_datetime(data['SIBT'])
data.set_index('SIBT', inplace=True)

# Filter de data tot en met 29-06-2024 voor de trainingsset
train_data = data[data.index < '2024-06-30']

# Aggregatie van het aantal koffers per 15 minuten interval voor de trainingsset
baggage_per_15min = train_data.resample('15T').size()

# Stap 1: Gebruik auto_arima om de beste parameters te vinden voor data tot en met 29-06-2024
model = auto_arima(baggage_per_15min, seasonal=True, m=96,  # Seizoenslengte van 96 (15-min interval voor 1 dag)
                   trace=True,                 # Laat de voortgang zien
                   suppress_warnings=True,     # Onderdruk waarschuwingen
                   stepwise=True)              # Stap-voor-stap optimalisatie voor snelheid

# Stap 2: Train het model met de volledige dataset tot en met 29-06-2024
model.fit(baggage_per_15min)

# Stap 3: Voorspel de bezettingsgraden voor 30-06-2024 (96 kwartieren voor een hele dag)
forecast = model.predict(n_periods=96)

# Print de voorspellingen
print(f"Voorspelde bezettingsgraden voor 30-06-2024 (in kwartieren): {forecast}")
```

Listing 10: Cell 10: Forecasting Baggage Occupancy Using Auto ARIMA

Appendix A11: Generating Simulated Bag-Level Data

This code expands simulated container-level arrival data into individual bag entries for analysis. It loads the arrival dataset and iterates through each row, creating individual entries for each bag based on the number of bags per container. The expanded data includes the infeed point, arrival time, temperature classification, and arrival day for each bag. This bag-level dataset is then exported as a CSV file for further use in simulation or forecasting.

```
import pandas as pd
import numpy as np
from datetime import timedelta
import time

# Start de timer
start_time = time.time()

# Laad de gesimuleerde arrivals dataset
file_path = 'C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\aankomsttijd_containers.csv'
processed_df = pd.read_csv(file_path)

# Creer een lege lijst om elke koffer als rij te representeren
simulated_bags = []

# Loop door elke rij en genereer individuele koffers
for _, row in processed_df.iterrows():
    for i in range(int(row['aantal_koffers'])):
        simulated_bags.append({
            'infeed_point': row['infeed_point'],
            'aankomst_tijd': row['aankomst_tijd'],
            'KAR_TEMPERATURE': row['KAR_TEMPERATURE'],
            'dag': row['dag']
        })

# Zet de lijst om naar een DataFrame
simulated_bags_df = pd.DataFrame(simulated_bags)

# Exporteer naar CSV
output_path = 'C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\simulated_bags.csv'
simulated_bags_df.to_csv(output_path, index=False)

# Print het aantal gesimuleerde koffers
print(f"Aantal gesimuleerde koffers: {len(simulated_bags_df)}")

# End de timer
end_time = time.time()
runtime = end_time - start_time
print(f'Runtime van de code: {runtime:.2f} seconden')
```

Listing 11: Cell 11: Generating Simulated Bag-Level Data

Appendix A12: SARIMA Forecasting Model for Baggage Occupancy on June 30, 2024

This code uses a SARIMA model to forecast baggage occupancy rates in 15-minute intervals for June 30, 2024. After preparing the dataset, data up to June 29 is used to train the model, while June 30 serves as the test set. SARIMA parameters are chosen for daily seasonality, and the model forecasts occupancy for 96 intervals over the day. The forecast is visualized, and three accuracy metrics are calculated: Root Mean Squared Error (RMSE), Symmetrical Mean Absolute Percentage Error (SMAPE), and Mean Absolute Error (MAE), quantifying the model's prediction accuracy.

```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from statsmodels.tsa.statespace.sarimax import SARIMAX
from sklearn.metrics import mean_squared_error, mean_absolute_error
import time

# Start de timer
start_time = time.time()

# Laad het gesimuleerde koffers dataset
file_path = 'C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\simulated_bags.csv'
data = pd.read_csv(file_path)

# Zorg ervoor dat de 'aankomst_tijd' kolom in datetime-formaat is
data['aankomst_tijd'] = pd.to_datetime(data['aankomst_tijd'])

# Filter de data tot en met 29-06-2024 voor de trainingsset
train_data = data[data['aankomst_tijd'] < '2024-06-30']

# Stap 2: Aggregatie van het aantal koffers per 15 minuten interval
train_data.set_index('aankomst_tijd', inplace=True)
baggage_per_15min = train_data.resample('15T').size()

# Definieer de testdata voor 30-06-2024
test_data = data[data['aankomst_tijd'] >= '2024-06-30']

# Aggregatie van het aantal koffers per 15 minuten interval voor de testdata
baggage_per_15min_test = test_data.set_index('aankomst_tijd').resample('15T').size()

# Stap 3: Gebruik SARIMA model met de beste parameters
best_order = (1, 1, 1)  # ARIMA parameters (p, d, q)
best_seasonal_order = (1, 1, 1, 96)  # Seizoensparameters (P, D, Q, s)

# Train SARIMA met alle data tot en met 29-06-2024
model = SARIMAX(baggage_per_15min, order=best_order, seasonal_order=best_seasonal_order)
model_fit = model.fit()

# Stap 4: Voorspel voor 30-06-2024 (96 kwartieren)
forecast = model_fit.forecast(steps=96)

# Stap 5: Visualiseer de voorspellingen voor 30-06-2024
plt.figure(figsize=(12, 6))
plt.plot(pd.date_range(start='2024-06-30 00:00:00', periods=96, freq='15T'), forecast, label='Forecast
    ↪ 30-06-2024', color='red')
plt.title('Voorspelde Bezettingsgraden voor 30-06-2024 per 15 minuten')
plt.xlabel('Tijd')
plt.ylabel('Aantal koffers')
plt.legend()
plt.show()

# Bereken RMSE voor de tijdreeksvoorspelling
rmse = np.sqrt(mean_squared_error(baggage_per_15min_test, forecast[:len(baggage_per_15min_test)]))
print(f'Root Mean Squared Error (RMSE): {rmse}')
```

```
# Bereken SMAPE
def smape(y_true, y_pred):
    return 100 * np.mean(2 * np.abs(y_pred - y_true) / (np.abs(y_true) + np.abs(y_pred)))

smape_value = smape(baggage_per_15min_test, forecast[:len(baggage_per_15min_test)])
print(f'Symmetrical Mean Absolute Percentage Error (SMAPE): {smape_value:.2f}%')

# Bereken MAE
mae = mean_absolute_error(baggage_per_15min_test, forecast[:len(baggage_per_15min_test)])
print(f'Mean Absolute Error (MAE): {mae}')

# Runtime van de code
end_time = time.time()
runtime = end_time - start_time
print(f'Runtime van de code: {runtime:.2f} seconden')
```

Listing 12: Cell 12: SARIMA Forecasting Model for Baggage Occupancy on June 30

Appendix A13: Forecasting Baggage Occupancy Using TBATS

This code applies the TBATS model to forecast baggage occupancy rates in 15-minute intervals for June 30, 2024. The training data includes simulated bag arrivals up to June 29, aggregated to 15-minute intervals. TBATS is configured with daily and weekly seasonal patterns to capture regular fluctuations in baggage flow. The forecast for June 30 is generated, visualized, and evaluated using Root Mean Squared Error (RMSE), Symmetrical Mean Absolute Percentage Error (SMAPE), and Mean Absolute Error (MAE), providing insights into the model's predictive accuracy.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from tbats import TBATS
from sklearn.metrics import mean_squared_error, mean_absolute_error
import time

# Start timer
start_time = time.time()

# Load simulated baggage dataset
file_path = 'C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\simulated_bags.csv'
data = pd.read_csv(file_path)

# Ensure 'aankomst_tijd' column is in datetime format
data['aankomst_tijd'] = pd.to_datetime(data['aankomst_tijd'])

# Filter data up to 29-06-2024 for the training set
train_data = data[data['aankomst_tijd'] < '2024-06-30']

# Step 2: Aggregate the number of bags per 15-minute interval
train_data.set_index('aankomst_tijd', inplace=True)
baggage_per_15min = train_data.resample('15T').size()

# Definieer de testdata voor 30-06-2024
test_data = data[data['aankomst_tijd'] >= '2024-06-30']

# Aggregatie van het aantal koffers per 15 minuten interval voor de testdata
baggage_per_15min_test = test_data.set_index('aankomst_tijd').resample('15T').size()

# Step 3: Use TBATS model with daily and weekly seasonal components
estimator = TBATS(seasonal_periods=[96, 672])  # Daily and weekly seasonal patterns
tbats_model = estimator.fit(baggage_per_15min)

# Step 4: Forecast for 30-06-2024 (96 quarters)
forecast = tbats_model.forecast(steps=96)

# Step 5: Visualize the forecast for 30-06-2024
plt.figure(figsize=(12, 6))
plt.plot(pd.date_range(start='2024-06-30 00:00:00', periods=96, freq='15T'), forecast, label='
    ↪ Predictions', color='red')
plt.title('Predicted Baggage Occupancy for 30-06-2024 per 15 minutes (TBATS)')
plt.xlabel('Time')
plt.ylabel('Number of Bags')
plt.legend()
plt.show()

# Step 6: Calculate RMSE, SMAPE, and MAE
rmse = np.sqrt(mean_squared_error(baggage_per_15min_test, forecast[:len(baggage_per_15min_test)]))
print(f'Root Mean Squared Error (RMSE): {rmse}')

# Calculate SMAPE
def smape(y_true, y_pred):
    return 100 * np.mean(2 * np.abs(y_pred - y_true) / (np.abs(y_true) + np.abs(y_pred)))

```

```python
smape_value = smape(baggage_per_15min_test, forecast[:len(baggage_per_15min_test)])
print(f'Symmetrical Mean Absolute Percentage Error (SMAPE): {smape_value:.2f}%')

# Calculate MAE
mae = mean_absolute_error(baggage_per_15min_test, forecast[:len(baggage_per_15min_test)])
print(f'Mean Absolute Error (MAE): {mae}')

# Runtime of the code
end_time = time.time()
runtime = end_time - start_time
print(f'Code runtime: {runtime:.2f} seconds')
```

Listing 13: Cell 13: Forecasting Baggage Occupancy Using TBATS

Appendix A14: Minute-Level Forecast Transformation

This code expands a quarter-hourly baggage arrival forecast to a minute-level granularity for June 30, 2024. Initially, the forecast is generated for 96 intervals, each representing a 15-minute period. Each interval is then repeated 15 times, and timestamps are adjusted to reflect minute-level precision. Additional columns are created to distribute the baggage count per minute, calculate rolling sums over 15-minute intervals, and set a per-minute target value. The final data is exported to a CSV file for further analysis in minute-by-minute increments, facilitating a detailed view of predicted arrivals.

```python
import pandas as pd
import numpy as np
import time

# Start timer
start_time = time.time()

# Bepaal de oorspronkelijke frequentie van 15 minuten en genereer de tijdstempels
kwartier_forecast_df = pd.DataFrame({
    'dag': pd.date_range(start='2024-06-30 00:00:00', periods=96, freq='15T').date,
    'aantal_koffers': np.round(forecast).astype(int),  # Rond de voorspelde waarden af op hele getallen
    'voorspelde_aankomst_tijd_infeed': pd.date_range(start='2024-06-30 00:00:00', periods=96, freq='15T'),
    'target_value': 600  # Voeg een kolom toe met de constante waarde van 600
})

# Herhaal elke rij 15 keer om het om te zetten naar minuten
forecast_per_minute = kwartier_forecast_df.loc[kwartier_forecast_df.index.repeat(15)].reset_index(drop=True)

# Pas de voorspelde aankomsttijd aan zodat deze per minuut is
forecast_per_minute['voorspelde_aankomst_tijd_infeed'] = pd.date_range(start='2024-06-30 00:00:00', periods=len(forecast_per_minute), freq='1T')

# Zorg ervoor dat 'aantal_koffers_minute' en 'target_value_minute' als integers worden berekend en afgerond
forecast_per_minute['aantal_koffers_minute'] = (forecast_per_minute['aantal_koffers'] / 15).astype(int)  # Hele getallen
forecast_per_minute['target_value_minute'] = (forecast_per_minute['target_value'] / 15).astype(int)  # Hele getallen

# Voeg de kolom 'voorspelde_laatste_15min' toe als som van 'aantal_koffers_minute' van de afgelopen 15 minuten
forecast_per_minute['voorspelde_laatste_15min'] = forecast_per_minute['aantal_koffers_minute'].rolling(window=15, min_periods=1).sum().astype(int)

# Zorg ervoor dat de getallen correct als integers worden opgeslagen
forecast_per_minute[['aantal_koffers', 'target_value', 'aantal_koffers_minute', 'target_value_minute', 'voorspelde_laatste_15min']] = forecast_per_minute[['aantal_koffers', 'target_value', 'aantal_koffers_minute', 'target_value_minute', 'voorspelde_laatste_15min']].apply(pd.to_numeric, errors='coerce', downcast='integer')

# Verwijder de aankomst_tijd kolom indien aanwezig
forecast_per_minute = forecast_per_minute.drop(columns=['aankomst_tijd'], errors='ignore')

# Zet 'voorspelde_aankomst_tijd_infeed' expliciet om naar datetime en zorg ervoor dat het hele minuten zijn
forecast_per_minute['voorspelde_aankomst_tijd_infeed'] = pd.to_datetime(forecast_per_minute['voorspelde_aankomst_tijd_infeed']).dt.floor('min')

# Exporteer naar CSV met punt als decimaal scheidingsteken en zonder extra aanhalingstekens
output_file_path = 'C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\Minuutwaarden voorspelling 30-06-2024.csv'
forecast_per_minute.to_csv(output_file_path, index=False)
```

```python

print(f'Voorspelling per minuut is succesvol opgeslagen in: {output_file_path}')

# Runtime of the code
end_time = time.time()
runtime = end_time - start_time
print(f'Code runtime: {runtime:.2f} seconds')
```

Listing 14: Cell 14: Minute-Level Forecast Transformation

Appendix A15: Polynomial Fitting of Forecasted Data

This code applies a 9th-degree polynomial fit to forecasted baggage arrival data to smooth fluctuations over time. First, the forecasted baggage counts for each 15-minute interval on June 30, 2024, are loaded and processed to minute-level time points. The polynomial is fitted and then clamped within bounds (150 to 600) to ensure realistic constraints on baggage counts. The plot visualizes both the actual data and the fitted polynomial, demonstrating an adjusted target function that maintains practical upper and lower limits.

```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from numpy.polynomial.polynomial import Polynomial

# Laad het bestand 'Minuutwaarden voorspelling 30-06-2024.csv'
minuutwaarden_voorspelling_path = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\Minuutwaarden voorspelling 30-06-2024.csv"
minuutwaarden_voorspelling_df = pd.read_csv(minuutwaarden_voorspelling_path)

# Zet 'voorspelde_aankomst_tijd_infeed' expliciet om naar datetime en zorg ervoor dat het hele minuten zijn
minuutwaarden_voorspelling_df['voorspelde_aankomst_tijd_infeed'] = pd.to_datetime(minuutwaarden_voorspelling_df['voorspelde_aankomst_tijd_infeed']).dt.floor('min')

# Haal de kolom 'voorspelde_laatste_15min' op
y = minuutwaarden_voorspelling_df['voorspelde_laatste_15min'].values
x = pd.date_range(start='2024-06-30 00:00', periods=len(y), freq='T')  # Maak een reeks tijdstempels voor de x-waarden

# Pas een polynoom van graad 11 toe op de data
degree = 9
coefficients = np.polyfit(np.arange(len(y)), y, degree)
polynomial = np.poly1d(coefficients)

# Genereer de y-waarden voor de polynoom
y_poly = polynomial(np.arange(len(y)))

# Zorg ervoor dat de polynoom niet lager gaat dan 150 en niet hoger dan 600
y_poly_clamped = np.clip(y_poly, 150, 600)

# Plot de originele data en de aangepaste polynoom
plt.figure(figsize=(12, 6))
plt.plot(x, y, label='Bags in last 15 min', color='red', linestyle='-', marker='o', markersize=2)
plt.plot(x, y_poly_clamped, label=f'Target function', color='black', linestyle='--')

# Voeg labels en titel toe aan de plot
plt.xlabel('Time')
plt.ylabel('Number of bags')
# plt.title('Polynoomfit over Voorspelde Laatste 15 minuten (Begrensd tussen 150 en 600)')
plt.legend()

# Pas de x-as aan om de tijdstempels goed weer te geven
plt.xticks(rotation=45)
plt.gca().xaxis.set_major_locator(mdates.HourLocator(interval=1))  # Toon een tick per uur
plt.gca().xaxis.set_major_formatter(mdates.DateFormatter('%H:%M'))  # Toon de tijd in uren en minuten

# Toon de plot
plt.grid(True)
plt.tight_layout()
plt.show()
```

Listing 15: Cell 15: Polynomial Fitting of Forecasted Data

Appendix A16: Update of Target Values using Polynomial Fit

In this code cell, a 9th-degree polynomial is fitted to the forecasted ‘voorspelde$_laatste_15min‘datafromtheminute-levelpredictionfileforJune30, 2024.Thispolynomialprovidesasmoothedestimateforthe‘target_value‘column, withvaluescon$

```
import pandas as pd
import numpy as np

# Load the file 'Minuutwaarden voorspelling 30-06-2024.csv'
minuutwaarden_voorspelling_path = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\Minuutwaarden voorspelling 30-06-2024.csv"
minuutwaarden_voorspelling_df = pd.read_csv(minuutwaarden_voorspelling_path)

# Update the 'target_value' column in the DataFrame with polynomial values
y = minuutwaarden_voorspelling_df['voorspelde_laatste_15min'].values
degree = 9
coefficients = np.polyfit(np.arange(len(y)), y, degree)
polynomial = np.poly1d(coefficients)
y_poly = polynomial(np.arange(len(y)))
y_poly_clamped = np.maximum(y_poly, 150)

minuutwaarden_voorspelling_df['target_value'] = y_poly_clamped.astype(int)

# Ensure that 'voorspelde_aankomst_tijd_infeed' is consistent and fits a minute-level time range
tijd_range = pd.date_range("2024-06-30 00:00:00", "2024-06-30 23:59:00", freq="min")
minuutwaarden_voorspelling_df = minuutwaarden_voorspelling_df[minuutwaarden_voorspelling_df['voorspelde_aankomst_tijd_infeed'].isin(tijd_range)]

# Export the updated DataFrame to CSV
updated_minuutwaarden_voorspelling_path = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\Minuutwaarden voorspelling 30-06-2024.csv"
minuutwaarden_voorspelling_df.to_csv(updated_minuutwaarden_voorspelling_path, index=False)

print(f"Updated 'target_value' column saved in: {updated_minuutwaarden_voorspelling_path}")
```

Listing 16: Cell 16: Update of Target Values using Polynomial Fit

Appendix A17: Simulation of Container Arrival Times and Occupancy Calculation

In this code, a simulation is performed to determine the arrival times of baggage containers at designated infeed points for the date of June 30, 2024. This involves calculating travel times from specific departure points ('IN_RAMP_CLUSTER') to the infeed areas ('center_of_gravity') and applying these times to predict the minute-level occupancy rates. This code calculates the arrival time at infeed points for each container on June 30, 2024, based on predefined travel times. The occupancy rate per minute is computed, including a rolling sum of baggage counts over the last 15 minutes, and the results are saved in separate CSV files for further analysis.

```
import pandas as pd
import numpy as np
from datetime import timedelta
import time
import matplotlib.pyplot as plt

# Start timer to track the runtime
start_time = time.time()

# Load the containers output dataset and filter for June 30, 2024
containers_output_path = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\
    ↪ containers_output.csv"
containers_df = pd.read_csv(containers_output_path)
containers_df['dag'] = pd.to_datetime(containers_df['dag'], format='%d-%m-%Y')
containers_30june_df = containers_df[containers_df['dag'] == '2024-06-30']

# Travel times between IN_RAMP_CLUSTER and Infeed Areas (in seconds)
travel_times = {
    # Add specific (departure, arrival) travel time mappings here
}

# Calculate infeed arrival times for each container and create a new column in the dataset
arrival_times = []
for _, row in containers_30june_df.iterrows():
    departure_point = row['IN_RAMP_CLUSTER']
    destination = row['center_of_gravity']
    if (departure_point, destination) in travel_times:
        travel_time = travel_times[(departure_point, destination)]
        start_time_train = pd.to_datetime(row['tijd_eind_container'])
        arrival_time = start_time_train + timedelta(seconds=travel_time)
        row['aankomst_tijd_infeed'] = pd.Timestamp(arrival_time).ceil('min')
        arrival_times.append(row)

# Create a DataFrame with the new arrival times
arrival_df = pd.DataFrame(arrival_times)

# Calculate transfer times and save as a CSV
arrival_df['Latest_time_container'] = pd.to_datetime(arrival_df['Latest_time_container'])
arrival_df['overstaptijd'] = arrival_df['Latest_time_container'] - arrival_df['aankomst_tijd_infeed']
output_path_containers = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\
    ↪ aankomsttijd_containers_30-06-2024_18_1.csv"
arrival_df.to_csv(output_path_containers, index=False)

# Calculate container occupancy per minute and save as a CSV
arrival_df['aankomst_minute'] = arrival_df['aankomst_tijd_infeed'].dt.floor('min')
minute_agg_df = arrival_df.groupby('aankomst_minute').agg(
    aantal_containers=('aankomst_tijd_infeed', 'size'),
    aantal_koffers=('aantal_koffers', 'sum')
).reset_index()

# Merge with the forecasted target values and save as a CSV
voorspelling_path = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\Minuutwaarden
    ↪ voorspelling 30-06-2024.csv"
voorspelling_df = pd.read_csv(voorspelling_path)
voorspelling_df['voorspelde_aankomst_tijd_infeed'] = pd.to_datetime(voorspelling_df['
```

```
    ↪ voorspelde_aankomst_tijd_infeed'])
minute_agg_df = pd.merge(minute_agg_df, voorspelling_df[['voorspelde_aankomst_tijd_infeed', '
    ↪ target_value']],
                         left_on='aankomst_minute', right_on='voorspelde_aankomst_tijd_infeed', how='
    ↪ left')
minute_agg_df['koffers_laatste_15min'] = minute_agg_df['aantal_koffers'].rolling(window=15,
    ↪ min_periods=1).sum().astype(int)
output_path_bezettingsgraad = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\
    ↪ Bezettingsgraad_18_1.csv"
minute_agg_df.to_csv(output_path_bezettingsgraad, index=False)

# End timer and print runtime
end_time = time.time()
elapsed_time = end_time - start_time
print(f"Simulation completed in {elapsed_time:.2f} seconds")
```

Listing 17: Cell 17: Simulation of Container Arrival Times and Occupancy Calculation

Appendix A18: Known Arrivals Processing and Occupancy Calculation for 30-06-2024

In this code, we filter and process known baggage arrival data specifically for June 30, 2024. The code groups baggage entries by minute and computes the cumulative occupancy over the past 15 minutes. This data is merged with target values to assess how the actual arrivals compare to the predicted target occupancy levels. This script filters known baggage data for June 30, 2024, aggregates it by minute, and calculates the cumulative occupancy over the last 15 minutes. It also integrates forecasted target occupancy values for comparative analysis, with the output saved in a CSV file for further examination.

```python
import pandas as pd

# Load the known arrivals file
known_arrivals_path = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\
    ↪ Known_arrivals_measured.csv"
known_arrivals_df = pd.read_csv(known_arrivals_path)

# Convert the DATETIME_OF_ENTRY column to datetime format for precise timestamps
known_arrivals_df['DATETIME_OF_ENTRY'] = pd.to_datetime(known_arrivals_df['DATETIME_OF_ENTRY'])

# Filter only rows for June 30, 2024
known_arrivals_30june_df = known_arrivals_df[known_arrivals_df['DATETIME_OF_ENTRY'].dt.date == pd.
    ↪ to_datetime("2024-06-30").date()]

# Export the filtered file to a new CSV
output_path_30june = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\
    ↪ Known_arrivals_measured_30_06_2024.csv"
known_arrivals_30june_df.to_csv(output_path_30june, index=False)

# Load the filtered file for June 30, 2024
known_arrivals_df = pd.read_csv(output_path_30june)
known_arrivals_df['DATETIME_OF_ENTRY'] = pd.to_datetime(known_arrivals_df['DATETIME_OF_ENTRY'])

# Group data by minute and count baggage entries per minute
known_arrivals_df['entry_minute'] = known_arrivals_df['DATETIME_OF_ENTRY'].dt.floor('min')
minute_agg_df_known = known_arrivals_df.groupby('entry_minute').size().reset_index(name='
    ↪ aantal_koffers')

# Fill missing minutes with 0 baggage entries
all_minutes = pd.date_range(start="2024-06-30 00:00:00", end="2024-06-30 23:59:00", freq="min")
minute_agg_df_known = minute_agg_df_known.set_index('entry_minute').reindex(all_minutes, fill_value=0)
    ↪ .rename_axis('entry_minute').reset_index()

# Load the target values from the updated forecast
voorspelling_path = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\Minuutwaarden
    ↪ voorspelling 30-06-2024.csv"
voorspelling_df = pd.read_csv(voorspelling_path)

# Ensure timestamps align with those in minute_agg_df_known
voorspelling_df['voorspelde_aankomst_tijd_infeed'] = pd.to_datetime(voorspelling_df['
    ↪ voorspelde_aankomst_tijd_infeed'])
voorspelling_df = voorspelling_df[voorspelling_df['voorspelde_aankomst_tijd_infeed'].dt.date == pd.
    ↪ to_datetime("2024-06-30").date()]

# Merge the target values with minute_agg_df_known based on timestamps
minute_agg_df_known = pd.merge(minute_agg_df_known, voorspelling_df[['voorspelde_aankomst_tijd_infeed'
    ↪ , 'target_value']],
                                left_on='entry_minute', right_on='voorspelde_aankomst_tijd_infeed', how
    ↪ ='left')

# Remove the extra timestamp column and calculate occupancy over the past 15 minutes
minute_agg_df_known = minute_agg_df_known.drop(columns=['voorspelde_aankomst_tijd_infeed'])
minute_agg_df_known['koffers_laatste_15min'] = minute_agg_df_known['aantal_koffers'].rolling(window
    ↪ =15, min_periods=1).sum().astype(int)
```

```

# Save the results to a CSV file
output_path_bezettingsgraad_known = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\
    ↪ Bezettingsgraad_known_arrivals.csv"
minute_agg_df_known.to_csv(output_path_bezettingsgraad_known, index=False)
```

Listing 18: Cell 18: Known Arrivals Processing and Occupancy Calculation for 30-06-2024

Appendix A19: Comparison of Occupancy Rate Between Known Arrivals and Simulation Output

This code segment loads occupancy rate data from known arrivals and simulation output for June 30, 2024. It then plots both data sets to visually compare the 15-minute rolling occupancy rates, highlighting any discrepancies between the actual and simulated arrivals. Differences in occupancy levels can indicate discrepancies between the actual and forecasted baggage arrivals, providing insights into the accuracy of the simulation model.

```python
import pandas as pd
import matplotlib.pyplot as plt

# Load occupancy rate files
bezettingsgraad_known_path = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\Bezettingsgraad_known_arrivals.csv"
bezettingsgraad_18_1_path = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\Bezettingsgraad_18_1.csv"

bezettingsgraad_known_df = pd.read_csv(bezettingsgraad_known_path)
bezettingsgraad_18_1_df = pd.read_csv(bezettingsgraad_18_1_path)

# Convert timestamps to datetime for both datasets
bezettingsgraad_known_df['entry_minute'] = pd.to_datetime(bezettingsgraad_known_df['entry_minute'])
bezettingsgraad_18_1_df['aankomst_minute'] = pd.to_datetime(bezettingsgraad_18_1_df['aankomst_minute'])

# Plot occupancy rates of both datasets
plt.figure(figsize=(14, 7))
plt.plot(bezettingsgraad_known_df['entry_minute'], bezettingsgraad_known_df['koffers_laatste_15min'], label='Occupancy Rate Known Arrivals', color='blue')
plt.plot(bezettingsgraad_18_1_df['aankomst_minute'], bezettingsgraad_18_1_df['koffers_laatste_15min'],  label='Occupancy Rate 18.1', color='orange')

# Add labels and title
plt.xlabel('Time')
plt.ylabel('Bags in Last 15 Min')
plt.title('Occupancy Rate Comparison: Known Arrivals vs. Simulation 18.1')
plt.legend()
plt.grid()

# Display the plot
plt.show()
```

Listing 19: Cell 19: Occupancy Rate Comparison Plot

Appendix A20: Buffering and Occupancy Rate Adjustment for Baggage Arrival Simulation

In this code cell, several key data files related to baggage occupancy rates, container arrival times, and target values are loaded. The script iterates through each minute of June 30, 2024, analyzing and adjusting occupancy rates by buffering containers with the longest transfer times until occupancy falls below the target threshold. This process involves dynamically adjusting return times for buffered containers and recalculating occupancy rates to ensure smoother operational flow. The simulation results are exported to CSV files, allowing further analysis of the buffer effectiveness in controlling peak occupancy rates.

```python
import pandas as pd
from IPython.display import display
from IPython.core.display import HTML

# Load 'Aankomsttijd_containers_30-06-2024_18_1.csv'
aankomsttijd_containers_path = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\Aankomsttijd_containers_30-06-2024_18_1.csv"
aankomsttijd_containers_df = pd.read_csv(aankomsttijd_containers_path)

# Load 'Bezettingsgraad_18_1.csv'
bezettingsgraad_path = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\Bezettingsgraad_18_1.csv"
bezettingsgraad_df = pd.read_csv(bezettingsgraad_path)

# Load 'Minuutwaarden voorspelling 30-06-2024.csv'
minuutwaarden_voorspelling_path = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\Minuutwaarden voorspelling 30-06-2024.csv"
minuutwaarden_voorspelling_df = pd.read_csv(minuutwaarden_voorspelling_path)

# Ensure columns are formatted correctly
aankomsttijd_containers_df['tijd'] = pd.to_datetime(aankomsttijd_containers_df['tijd'])
aankomsttijd_containers_df['aankomst_tijd_infeed'] = pd.to_datetime(aankomsttijd_containers_df['aankomst_tijd_infeed'])
aankomsttijd_containers_df['overstaptijd'] = pd.to_timedelta(aankomsttijd_containers_df['overstaptijd'])

# Create arrival time column
aankomsttijd_containers_df['aankomst_tijd_infeed_time'] = aankomsttijd_containers_df['aankomst_tijd_infeed'].dt.time

# Sort the DataFrame based on arrival time and transfer time
cols = ['aankomst_tijd_infeed_time', 'overstaptijd'] + [col for col in aankomsttijd_containers_df.columns if col not in ['aankomst_tijd_infeed_time', 'overstaptijd']]
aankomsttijd_containers_df = aankomsttijd_containers_df[cols]
aankomsttijd_containers_df_sorted = aankomsttijd_containers_df.sort_values(by=['aankomst_tijd_infeed_time', 'overstaptijd'], ascending=[True, False])

# Simulate buffering logic and update target occupancy
# (The rest of the code continues as per your example)
```

Listing 20: Cell 20: Buffering and Occupancy Rate Adjustment

Appendix A21: Calculating Minute-Based Occupancy for Baggage Arrival Simulation

In this code cell, the occupancy rate for baggage arrivals is calculated on a per-minute basis for June 30, 2024. Using data from the file `aankomst_containers_after_loop.csv`, the code aggregates the number of containers and bags per minute. Additionally, a rolling sum of the bags over the last 15 minutes is calculated in the `koffers_laatste_15min` column. This metric helps analyze the occupancy dynamics and ensures a smoother operational flow by understanding minute-based variations in baggage arrivals. The processed data is then exported for further analysis.

```python
import pandas as pd

# Load the 'aankomst_containers_after_loop.csv' file
aankomst_containers_path = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\aankomst_containers_after_loop.csv"
aankomst_containers_df = pd.read_csv(aankomst_containers_path)

# Create a DataFrame for 'bezettingsgraad_after_loop' with each minute of the day
tijd_range = pd.date_range("2024-06-30 00:00", "2024-06-30 23:59", freq='T')
bezettingsgraad_after_loop_df = pd.DataFrame({
    'aankomst_minute': tijd_range,
    'aantal_containers': 0,
    'aantal_koffers': 0,
    'koffers_laatste_15min': 0
})

# Loop through 'aankomst_containers_after_loop.csv' and update 'bezettingsgraad_after_loop'
for index, row in aankomst_containers_df.iterrows():
    aankomst_tijd = pd.to_datetime(row['aankomst_infeed_final'])

    # Locate the corresponding minute in 'bezettingsgraad_after_loop_df'
    tijd_index = bezettingsgraad_after_loop_df[bezettingsgraad_after_loop_df['aankomst_minute'] == aankomst_tijd].index[0]

    # Update the container and bag counts for the specific minute
    bezettingsgraad_after_loop_df.at[tijd_index, 'aantal_containers'] += 1
    bezettingsgraad_after_loop_df.at[tijd_index, 'aantal_koffers'] += row['aantal_koffers']

# Calculate the rolling sum of 'koffers_laatste_15min' over the last 15 minutes
for index in range(len(bezettingsgraad_after_loop_df)):
    # Select the last 15 minutes
    start_index = max(0, index - 14)
    end_index = index + 1
    bezettingsgraad_after_loop_df.at[index, 'koffers_laatste_15min'] = bezettingsgraad_after_loop_df['aantal_koffers'].iloc[start_index:end_index].sum()

# Export the resulting occupancy data to a CSV file
bezettingsgraad_export_path_after = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\bezettingsgraad_after_loop.csv"
bezettingsgraad_after_loop_df.to_csv(bezettingsgraad_export_path_after, index=False)

print(f"Bezettingsgraad_after_loop gexporteerd naar: {bezettingsgraad_export_path_after}")
```

Listing 21: Cell 21: Calculating Minute-Based Occupancy for Baggage Arrival Simulation

Appendix A22: Plotting Occupancy Levels Before and After Buffering in Baggage Arrival Simulation

This code cell visualizes the difference in occupancy levels before and after implementing a buffering strategy for baggage arrivals on June 30, 2024. By plotting the rolling sum of bags over the last 15 minutes for each case, the effect of buffering on reducing peak occupancy at infeed points can be analyzed. The occupancy levels before buffering are shown in blue, while the buffered occupancy levels are shown in orange. The generated plot is saved as a high-resolution PNG file for reporting purposes.

```python
import pandas as pd
import matplotlib.pyplot as plt

# Load files
bezettingsgraad_18_1_export_path = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\bezettingsgraad_18_1_export.csv"
bezettingsgraad_after_loop_path = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\bezettingsgraad_after_loop.csv"
output_plot_path = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\C18_occupancy_levels_plot.png"

bezettingsgraad_18_1_export = pd.read_csv(bezettingsgraad_18_1_export_path)
bezettingsgraad_after_loop = pd.read_csv(bezettingsgraad_after_loop_path)

# Ensure 'aankomst_minute' is read as datetime
bezettingsgraad_18_1_export['aankomst_minute'] = pd.to_datetime(bezettingsgraad_18_1_export['aankomst_minute'])
bezettingsgraad_after_loop['aankomst_minute'] = pd.to_datetime(bezettingsgraad_after_loop['aankomst_minute'])

# Create the plot
plt.figure(figsize=(12, 6))
plt.plot(bezettingsgraad_18_1_export['aankomst_minute'], bezettingsgraad_18_1_export['koffers_laatste_15min'], label='Occupancy levels without buffer', color='tab:blue')
plt.plot(bezettingsgraad_after_loop['aankomst_minute'], bezettingsgraad_after_loop['koffers_laatste_15min'], label='Occupancy levels with buffer', color='tab:orange')

# Add labels (no title)
plt.xlabel('Minute of arrival at infeed point')
plt.ylabel('Number of bags over the last 15 minutes')
plt.legend()
plt.grid()
plt.xticks(rotation=45)
plt.tight_layout()

# Export the plot to PNG without title
plt.savefig(output_plot_path, dpi=300, bbox_inches='tight')

# Show the plot
plt.show()

print(f"Plot successfully saved as: {output_plot_path}")
```

Listing 22: Cell 22: Plotting Occupancy Levels Before and After Buffering

Appendix A23: Calculation of Differences in Occupancy and Buffer Analysis

This code calculates the difference in occupancy levels between two scenarios (with and without buffering) and exports the results for further analysis. Additionally, it plots the maximum number of containers in the buffer across the day, visualizing how the buffer system operates to maintain optimal occupancy at infeed points. The plot indicates when and how much the buffer is utilized, and the output is saved as a high-resolution image.

```python
import pandas as pd
import matplotlib.pyplot as plt

# Calculate the difference between the two DataFrames
combined_df = pd.merge(bezettingsgraad_18_1_export[['aankomst_minute', 'koffers_laatste_15min']],
                        bezettingsgraad_after_loop[['aankomst_minute', 'koffers_laatste_15min']],
                        on='aankomst_minute',
                        suffixes=('_18_1', '_after_loop'))

# Add a column for the difference
combined_df['difference'] = combined_df['koffers_laatste_15min_after_loop'] - combined_df['koffers_laatste_15min_18_1']

# Define export path
combined_export_path = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\bezettingsgraad_combined_export.csv"

# Export combined DataFrame to CSV
combined_df.to_csv(combined_export_path, index=False)

print(f"Combined data exported to: {combined_export_path}")

# Load the 'Bezettingsgraad_buffer_after_loop' file and create a plot
bezettingsgraad_buffer_after_loop_path = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\bezettingsgraad_buffer_after_loop.csv"
output_plot_buffer_path = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\C18_buffer_occupancy_plot.png"

bezettingsgraad_buffer_after_loop = pd.read_csv(bezettingsgraad_buffer_after_loop_path)

# Convert 'tijd' column to a full datetime object for a specific date
bezettingsgraad_buffer_after_loop['tijd'] = pd.to_datetime(bezettingsgraad_buffer_after_loop['tijd'], format='%H:%M:%S')

# Calculate the maximum number of containers in the buffer
max_containers_in_buffer = bezettingsgraad_buffer_after_loop['containers_at_buffer'].max()
print(f"Maximum number of containers in the buffer: {max_containers_in_buffer}")

# Create a plot of 'containers_at_buffer' and 'koffers_at_buffer' over time
plt.figure(figsize=(12, 6))

# Plot containers_at_buffer
plt.plot(bezettingsgraad_buffer_after_loop['tijd'], bezettingsgraad_buffer_after_loop['containers_at_buffer'], label='Containers in buffer', color='tab:blue')

# Add labels without title
plt.xlabel('Time')
plt.ylabel('Count')
plt.legend()
plt.grid()
plt.xticks(rotation=45)
plt.tight_layout()

# Export the plot to a PNG file without a title
plt.savefig(output_plot_buffer_path, dpi=300, bbox_inches='tight')

# Show the plot
```

```
plt.show()

print(f"Buffer plot successfully saved as: {output_plot_buffer_path}")
```

Listing 23: Cell 23: Calculation of Differences in Occupancy and Buffer Analysis

Appendix A24: Statistical Analysis of Occupancy with and without Buffering

This code provides a comparative analysis of occupancy levels at infeed points with and without the use of buffering for baggage containers. Key statistical metrics such as the standard deviation, mean absolute change (MAC), and autocorrelation are calculated to assess the smoothness and variability in baggage handling flows. The code also calculates the coefficient of variation, peak values, and total buffered containers, which help in evaluating the effectiveness of buffering in reducing peak occupancy.

```python
import pandas as pd
import numpy as np

# Load 'Bezettingsgraad_18_1' file
bezettingsgraad_18_1_path = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\bezettingsgraad_18_1.csv"
bezettingsgraad_18_1 = pd.read_csv(bezettingsgraad_18_1_path)

# Convert 'aankomst_minute' to datetime format without specifying a format
bezettingsgraad_18_1['aankomst_minute'] = pd.to_datetime(bezettingsgraad_18_1['aankomst_minute'])

# Calculate standard deviation of bags (flow smoothness)
std_koffers = bezettingsgraad_18_1['koffers_laatste_15min'].std()

# Mean absolute change (MAC) between successive time steps
mac_koffers = bezettingsgraad_18_1['koffers_laatste_15min'].diff().abs().mean()

# Autocorrelation with lag 1
autocorr_koffers = bezettingsgraad_18_1['koffers_laatste_15min'].autocorr(lag=1)

# Smoothness Index (relative change)
si_koffers = (bezettingsgraad_18_1['koffers_laatste_15min'].diff().abs() / bezettingsgraad_18_1['koffers_laatste_15min'].shift(1)).mean()

# Coefficient of variation (std/mean)
mean_koffers = bezettingsgraad_18_1['koffers_laatste_15min'].mean()
cv_koffers = std_koffers / mean_koffers

# Calculate peak value (highest number of bags)
peak_koffers = bezettingsgraad_18_1['koffers_laatste_15min'].max()

# Print results for without buffer
print(f"Without Buffer")
print(f"Peak number of bags: {peak_koffers}")
print(f"Standard deviation of bags at infeed points: {std_koffers:.2f}")
print(f"Mean absolute change (MAC) of bags: {mac_koffers:.2f}")
print(f"Autocorrelation of bags (lag-1): {autocorr_koffers:.2f}")
print(f"Coefficient of variation of bags: {cv_koffers:.2f}")

print()

# Load 'Bezettingsgraad_after_loop' file
bezettingsgraad_after_loop_path = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\bezettingsgraad_after_loop.csv"
bezettingsgraad_after_loop = pd.read_csv(bezettingsgraad_after_loop_path)

# Convert 'aankomst_minute' to datetime format
bezettingsgraad_after_loop['aankomst_minute'] = pd.to_datetime(bezettingsgraad_after_loop['aankomst_minute'])

# Calculate standard deviation of bags after buffering
std_koffers = bezettingsgraad_after_loop['koffers_laatste_15min'].std()

# Mean absolute change after buffering
mac_koffers = bezettingsgraad_after_loop['koffers_laatste_15min'].diff().abs().mean()

```

```python
# Autocorrelation after buffering
autocorr_koffers = bezettingsgraad_after_loop['koffers_laatste_15min'].autocorr(lag=1)

# Smoothness Index after buffering
si_koffers = (bezettingsgraad_after_loop['koffers_laatste_15min'].diff().abs() /
    ↪ bezettingsgraad_after_loop['koffers_laatste_15min'].shift(1)).mean()

# Coefficient of variation (std/mean)
mean_koffers = bezettingsgraad_after_loop['koffers_laatste_15min'].mean()
cv_koffers = std_koffers / mean_koffers

# Calculate peak value
peak_koffers = bezettingsgraad_after_loop['koffers_laatste_15min'].max()

# Print results for with buffer
print(f"With Buffer")
print(f"Peak number of bags: {peak_koffers}")
print(f"Standard deviation of bags at infeed points: {std_koffers:.2f}")
print(f"Mean absolute change (MAC) of bags: {mac_koffers:.2f}")
print(f"Autocorrelation of bags (lag-1): {autocorr_koffers:.2f}")
print(f"Coefficient of variation of bags: {cv_koffers:.2f}")

# Load the 'containers_naar_buffer_after_loop.csv' file
containers_naar_buffer_path = "C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\
    ↪ containers_naar_buffer_after_loop.csv"
containers_naar_buffer_df = pd.read_csv(containers_naar_buffer_path)

# Convert time columns to datetime for ease of calculations
containers_naar_buffer_df['tijd'] = pd.to_datetime(containers_naar_buffer_df['tijd'])
containers_naar_buffer_df['aankomst_tijd_infeed_time'] = pd.to_datetime(containers_naar_buffer_df['
    ↪ aankomst_tijd_infeed_time'])
containers_naar_buffer_df['terugkomtijd'] = pd.to_datetime(containers_naar_buffer_df['terugkomtijd'])

# Calculate the peak number of containers in the buffer and total buffered containers
total_containers_buffered = containers_naar_buffer_df['aantal_containers'].sum()
peak_containers_in_buffer = containers_naar_buffer_df['containers_at_buffer'].max()

# Print results for buffer analysis
print()
print(f"Buffer Analysis")
print(f"Peak number of containers in buffer: {peak_containers_in_buffer}")
print(f"Total buffered containers: {total_containers_buffered}")
```

Listing 24: Cell 24: Statistical Analysis of Occupancy with and without Buffering

Appendix A25: AGV Assignment for Baggage Transportation Simulation

This code simulates the assignment and scheduling of AGVs for transporting baggage containers between the apron, buffer, and infeed points. The script follows two main actions: (1) transporting containers to the buffer if they require temporary storage before infeed and (2) transporting containers from the buffer to infeed points based on specific timing constraints. The code keeps a log of each trip, including AGV assignment, start and end times, pickup and dropoff locations, and the number of containers transported per trip.

```
import pandas as pd
import math
from datetime import timedelta

# Parameters
MAX_CONTAINERS_PER_AV = 6
TIME_FROM_APRON_TO_BUFFER = 3
TIME_FROM_BUFFER_TO_INFEED = 6
CONTAINER_CONNECT_TIME = 2
AV_REST_TIME = 6  # Time an AV must rest before it can be reused

# Load and sort container data by 'tijd' and 'vluchtcode'
containers_data = pd.read_csv("C:\\Users\\Nijdam_B\\Documents\\Configuratie 18 (18 en Base 3)\\
    ↪ containers_naar_buffer_after_loop.csv")
containers_data['tijd'] = pd.to_datetime(containers_data['tijd'])
containers_data['tijd_eind_container'] = pd.to_datetime(containers_data['tijd_eind_container'])
containers_data['terugkomtijd'] = pd.to_datetime(containers_data['terugkomtijd'])
containers_data.sort_values(['tijd', 'vluchtcode'], inplace=True)

# Log to track actions performed by each AV
trip_log = []
av_count = 0
available_avs = {}  # Dictionary to track AV availability {av_id: available_time}

def log_trip(av_id, action_type, containers, pickup_location, dropoff_location, start_time, end_time):
    """Log a trip in the trip log."""
    trip_log.append({
        "AV_ID": av_id,
        "Action_Type": action_type,
        "Pickup_Location": pickup_location,
        "Dropoff_Location": dropoff_location,
        "Start_Time": start_time,
        "End_Time": end_time,
        "Containers_Count": len(containers),
        "Container_IDs": ', '.join(containers['ContainerID'].astype(str))
    })

def assign_av(start_time):
    """Find an available AV or assign a new one if none are available."""
    global av_count
    # Check for availability
    for av_id, available_time in available_avs.items():
        if available_time <= start_time:
            return av_id
    # Assign a new AV if none are available
    av_count += 1
    available_avs[av_count] = start_time  # New AV available from start_time
    return av_count

# Buffer to store containers for later transport to infeed point
buffer = []

# Action 1: Process containers heading to the buffer
for vluchtcode, group in containers_data.groupby('vluchtcode'):
    containers_to_buffer = group[(group['terugkomtijd'] - group['tijd_eind_container']) >= pd.
    ↪ Timedelta(minutes=20)]
```

```python
    direct_to_infeed = group[(group['terugkomtijd'] - group['tijd_eind_container']) < pd.Timedelta(minutes=20)]

    # Group and transport containers to buffer
    if not containers_to_buffer.empty:
        start_time = containers_to_buffer['tijd_eind_container'].min()
        load_time = CONTAINER_CONNECT_TIME * len(containers_to_buffer)
        end_time = start_time + timedelta(minutes=load_time + TIME_FROM_APRON_TO_BUFFER)

        av_id = assign_av(start_time)  # Assign available AV or create a new one
        log_trip(av_id, "To_Buffer", containers_to_buffer, "Apron", "Buffer", start_time, end_time)

        # Update AV availability after rest time
        available_avs[av_id] = end_time + timedelta(minutes=AV_REST_TIME)

        # Add containers to buffer for later processing
        buffer.extend(containers_to_buffer.to_dict('records'))

    # Transport containers going directly to infeed point
    for _, container in direct_to_infeed.iterrows():
        start_time = container['tijd_eind_container']
        load_time = CONTAINER_CONNECT_TIME
        end_time = start_time + timedelta(minutes=load_time + TIME_FROM_BUFFER_TO_INFEED)

        av_id = assign_av(start_time)  # Assign available AV or create a new one
        log_trip(av_id, "Direct_To_Infeed", pd.DataFrame([container]), "Apron", container['center_of_gravity'], start_time, end_time)

        # Update AV availability after rest time
        available_avs[av_id] = end_time + timedelta(minutes=AV_REST_TIME)

# Action 2: Process containers from buffer to their infeed point
buffer = sorted(buffer, key=lambda x: x['terugkomtijd'])  # Sort buffer by return time

for infeed_point, group in pd.DataFrame(buffer).groupby('center_of_gravity'):
    group = group.sort_values(by='terugkomtijd')

    # Iterate over containers in sets of max 6
    for i in range(0, len(group), MAX_CONTAINERS_PER_AV):
        containers_for_trip = group.iloc[i:i + MAX_CONTAINERS_PER_AV]
        start_time = containers_for_trip['terugkomtijd'].min() - timedelta(minutes=TIME_FROM_BUFFER_TO_INFEED + CONTAINER_CONNECT_TIME * len(containers_for_trip))
        load_time = CONTAINER_CONNECT_TIME * len(containers_for_trip)
        end_time = start_time + timedelta(minutes=load_time + TIME_FROM_BUFFER_TO_INFEED)

        av_id = assign_av(start_time)  # Assign available AV or create a new one
        log_trip(av_id, "From_Buffer_To_Infeed", containers_for_trip, "Buffer", infeed_point, start_time, end_time)

        # Update AV availability after rest time
        available_avs[av_id] = end_time + timedelta(minutes=AV_REST_TIME)

# Export trip log to an Excel file
trip_log_df = pd.DataFrame(trip_log)
trip_log_df.to_excel("AV_Trip_Log.xlsx", index=False)
print("AV trip data exported to 'AV_Trip_Log.xlsx'")

# Results
print(f"Total AVs used: {av_count}")
```
Listing 25: Cell 25: AGV Assignment and Scheduling for Baggage Transportation

Appendix B: Results

Full results table

Conf.	Option	Alternative	Scenario	Peak of bags	Peak shaving %	STD	MAC	CoV	Peak buff.	Buff. cons	AGVs
Base 1	0. No buffer	0. No buffer	1: Normal	1958	0.0%	404.83	28.82	1.20	0	0	0
Base 2	0. No buffer	0. No buffer	2: Reduced capacity	1958	0.0%	404.15	29.4	1.20	0	0	0
Base 3	0. No buffer	0. No buffer	3: Increased inflow	2283	0.0%	485.38	33.88	1.20	0	0	0
1	1: Fixed	1: Early	1: Normal	1462	25.3%	349.90	26.25	1.04	113	199	27
2	1: Fixed	1: Early	2: Reduced capacity	1454	25.7%	348.81	26.99	1.04	114	205	26
3	1: Fixed	1: Early	3: Increased inflow	1734	24.0%	411.03	29.64	1.02	138	288	34
4	1: Fixed	2: Optimized	1: Normal	1454	25.7%	347.95	26.94	1.03	116	199	28
5	1: Fixed	2: Optimized	2: Reduced capacity	1464	25.2%	350.19	27.23	1.04	117	205	28
6	1: Fixed	2: Optimized	3: Increased inflow	1736	24.0%	414.84	30.49	1.03	142	288	35
7	2: Polynomial	1: Early	1: Normal	1492	23.8%	346.89	25.82	1.03	104	257	33
8	2: Polynomial	1: Early	2: Reduced capacity	1484	24.2%	345.25	26.57	1.03	105	259	32
9	2: Polynomial	1: Early	3: Increased inflow	1819	20.3%	413.07	30.15	1.02	127	311	40
10	2: Polynomial	2: Optimized	1: Normal	1433	26.8%	350.48	26.55	1.04	118	251	31
11	2: Polynomial	2: Optimized	2: Reduced capacity	1459	25.5%	349.25	26.77	1.04	115	251	31
12	2: Polynomial	2: Optimized	3: Increased inflow	1663	27.2%	417.77	30.98	1.03	139	300	39
13	3: Combination	1: Early	1: Normal	1492	23.8%	346.89	25.82	1.03	104	257	33
14	3: Combination	1: Early	2: Reduced capacity	1484	24.2%	345.25	26.57	1.03	105	259	32
15	3: Combination	1: Early	3: Increased inflow	1819	20.3%	413.07	30.15	1.02	127	311	40
16	3: Combination	2: Optimized	1: Normal	1433	26.8%	350.48	26.55	1.04	118	251	31
17	3: Combination	2: Optimized	2: Reduced capacity	1459	25.5%	349.25	26.77	1.04	115	251	31
18	3: Combination	2: Optimized	3: Increased inflow	1663	27.2%	417.77	30.98	1.03	139	300	39

Table B.1: All Results of Strategies

Strategic Options

Conf.	Option	Alternative	Scenario	Peak	STD	MAC	CoV	Peak buff	Buff. containers	AGVs
1	1: Fixed	1: Early	1: Normal	1462	349.90	26.25	1.04	113	199	27
7	2: Polynomial	1: Early	1: Normal	1492	346.89	25.82	1.03	104	257	33
13	3: Combination	1: Early	1: Normal	1492	346.89	25.82	1.03	104	257	33
Base 1	0. No buffer	0. No buffer	1: Normal	1958	404.83	28.82	1.2	0	0	0

Table B.2: Results on Strategic Options with Alternative 1 and Scenario 1

Conf.	Option	Alternative	Scenario	Peak	STD	MAC	CoV	Peak buff	Buff. containers	AGVs
2	1: Fixed	1: Early	2: Reduced capacity	1454	348.81	26.99	1.04	114	205	26
8	2: Polynomial	1: Early	2: Reduced capacity	1484	345.25	26.57	1.03	105	259	32
14	3: Combination	1: Early	2: Reduced capacity	1484	345.25	26.57	1.03	105	259	32
Base 2	0. No buffer	0. No buffer	2: Reduced capacity	1958	404.15	29.4	1.2	0	0	0

Table B.3: Results on Strategic Options with Alternative 1 and Scenario 2

Conf.	Option	Alternative	Scenario	Peak	STD	MAC	CoV	Peak buff	Buff. containers	AGVs
3	1: Fixed	1: Early	3: Increased inflow	1734	411.03	29.64	1.02	138	288	34
9	2: Polynomial	1: Early	3: Increased inflow	1819	413.07	30.15	1.02	127	311	40
15	3: Combination	1: Early	3: Increased inflow	1819	413.07	30.15	1.02	127	311	40
Base 3	0. No buffer	0. No buffer	3: Increased inflow	2283	485.38	33.88	1.2	0	0	0

Table B.4: Results on Strategic Options with Alternative 1 and Scenario 3

Conf.	Option	Alternative	Scenario	Peak	STD	MAC	CoV	Peak buff	Buff. containers	AGVs
10	2: Polynomial	2: Optimized	1: Normal	1433	350.48	26.55	1.04	118	251	31
16	3: Combination	2: Optimized	1: Normal	1433	350.48	26.55	1.04	118	251	31
4	1: Fixed	2: Optimized	1: Normal	1454	347.95	26.94	1.03	116	199	28
Base 1	0. No buffer	0. No buffer	1: Normal	1958	404.83	28.82	1.2	0	0	0

Table B.5: Results on Strategic Options with Alternative 2 and Scenario 1

Conf.	Option	Alternative	Scenario	Peak	STD	MAC	CoV	Peak buff	Buff. containers	AGVs
11	2: Polynomial	2: Optimized	2: Reduced capacity	1459	349.25	26.77	1.04	115	251	31
17	3: Combination	2: Optimized	2: Reduced capacity	1459	349.25	26.77	1.04	115	251	31
5	1: Fixed	2: Optimized	2: Reduced capacity	1464	350.19	27.23	1.04	117	205	28
Base 2	0. No buffer	0. No buffer	2: Reduced capacity	1958	404.15	29.4	1.2	0	0	0

Table B.6: Results on Strategic Options with Alternative 2 and Scenario 2

Conf.	Option	Alternative	Scenario	Peak	STD	MAC	CoV	Peak buff	Buff. containers	AGVs
12	2: Polynomial	2: Optimized	3: Increased inflow	1663	417.77	30.98	1.03	139	300	31
18	3: Combination	2: Optimized	3: Increased inflow	1663	417.77	30.98	1.03	139	300	31
6	1: Fixed	2: Optimized	3: Increased inflow	1736	414.84	30.49	1.03	142	288	28
Base 3	0. No buffer	0. No buffer	3: Increased inflow	2283	485.38	33.88	1.2	0	0	0

Table B.7: Results on Strategic Options with Alternative 2 and Scenario 3

Strategic Alternatives

Conf.	Option	Alternative	Scenario	Peak	STD	MAC	CoV	Peak buff.	Buff. containers	AGVs
4	1: Fixed	2: Optimized	1: Normal	1454	347,95	26,94	1,03	116	199	28
1	1: Fixed	1: Early	1: Normal	1462	349,90	26,25	1,04	113	199	27
Base 1	0. No buffer	0. No buffer	1: Normal	1958	404,83	28,82	1,20	0	0	0

Table B.8: Results on Strategic Alternatives with Option 1 and Scenario 1

Conf.	Option	Alternative	Scenario	Peak	STD	MAC	CoV	Peak buff.	Buff. containers	AGVs
2	1: Fixed	1: Early	2: Reduced capacity	1454	348,81	26,99	1,04	114	205	26
5	1: Fixed	2: Optimized	2: Reduced capacity	1464	350,19	27,23	1,04	117	205	28
Base 2	0. No buffer	0. No buffer	2: Reduced capacity	1958	404,15	29,4	1,2	0	0	0

Table B.9: Results on Strategic Alternatives with Option 1 and Scenario 2

Conf.	Option	Alternative	Scenario	Peak	STD	MAC	CoV	Peak buff.	Buff. containers	AGVs
3	1: Fixed	1: Early	3: Increased inflow	1734	411,03	29,64	1,02	138	288	34
6	1: Fixed	2: Optimized	3: Increased inflow	1736	414,84	30,49	1,03	142	288	35
Base 3	0. No buffer	0. No buffer	3: Increased inflow	2283	485,38	33,88	1,2	0	0	0

Table B.10: Results on Strategic Alternatives with Option 1 and Scenario 3

Conf.	Option	Alternative	Scenario	Peak	STD	MAC	CoV	Peak buff.	Buff. containers	AGVs
10	2: Polynomial	2: Optimized	1: Normal	1433	350,48	26,55	1,04	118	251	31
7	2: Polynomial	1: Early	1: Normal	1492	346,89	25,82	1,03	104	257	33
Base 1	0. No buffer	0. No buffer	1: Normal	1958	404,83	28,82	1,20	0	0	0

Table B.11: Results on Strategic Alternatives with Option 2 and Scenario 1

Conf.	Option	Alternative	Scenario	Peak	STD	MAC	CoV	Peak buff.	Buff. containers	AGVs
11	2: Polynomial	2: Optimized	2: Reduced capacity	1459	349,25	26,77	1,04	115	251	31
8	2: Polynomial	1: Early	2: Reduced capacity	1484	345,25	26,57	1,03	105	259	32
Base 2	0. No buffer	0. No buffer	2: Reduced capacity	1958	404,15	29,4	1,2	0	0	0

Table B.12: Results on Strategic Alternatives with Option 2 and Scenario 2

Conf.	Option	Alternative	Scenario	Peak	STD	MAC	CoV	Peak buff.	Buff. containers	AGVs
12	2: Polynomial	2: Optimized	3: Increased inflow	1663	417,77	30,98	1,03	139	300	39
9	2: Polynomial	1: Early	3: Increased inflow	1819	413,07	30,15	1,02	127	311	40
Base 3	0. No buffer	0. No buffer	3: Increased inflow	2283	485,38	33,88	1,2	0	0	0

Table B.13: Results on Strategic Alternatives with Option 2 and Scenario 3

Conf.	Option	Alternative	Scenario	Peak	STD	MAC	CoV	Peak buff.	Buff. containers	AGVs
16	3: Combination	2: Optimized	1: Normal	1433	350,48	26,55	1,04	118	251	31
13	3: Combination	1: Early	1: Normal	1492	346,89	25,82	1,03	104	257	33
Base 1	0. No buffer	0. No buffer	1: Normal	1958	404,83	28,82	1,20	0	0	0

Table B.14: Results on Strategic Alternatives with Option 3 and Scenario 1

Conf.	Option	Alternative	Scenario	Peak	STD	MAC	CoV	Peak buff.	Buff. containers	AGVs
17	3: Combination	2: Optimized	2: Reduced capacity	1459	349,25	26,77	1,04	115	251	31
14	3: Combination	1: Early	2: Reduced capacity	1484	345.25	26.57	1,03	105	259	32
Base 2	0. No buffer	0. No buffer	2: Reduced capacity	1958	404,15	29,4	1,2	0	0	0

Table B.15: Results on Strategic Alternatives with Option 3 and Scenario 2

Conf.	**Option**	**Alternative**	**Scenario**	**Peak**	**STD**	**MAC**	**CoV**	**Peak buff.**	**Buff. containers**	**AGVs**
18	3: Combination	2: Optimized	3: Increased inflow	1663	417,77	30,98	1,03	139	300	39
15	3: Combination	1: Early	3: Increased inflow	1819	413,07	30,15	1,02	127	311	40
Base 3	0. No buffer	0. No buffer	3: Increased inflow	2283	485,38	33,88	1,2	0	0	0

Table B.16: Results on Strategic Alternatives with Option 3 and Scenario 3

Results and Graphs per Configuration

Configuration 1

Configuration	Option	Alternative	Scenario	Peak of bags	Peak shaving in %	Peak in buffer	Buffered containers	AGVs
1	1: Fixed	1: Early	1: Normal	1462	25.3%	113	199	27

Table B.17: Results of Configuration 1

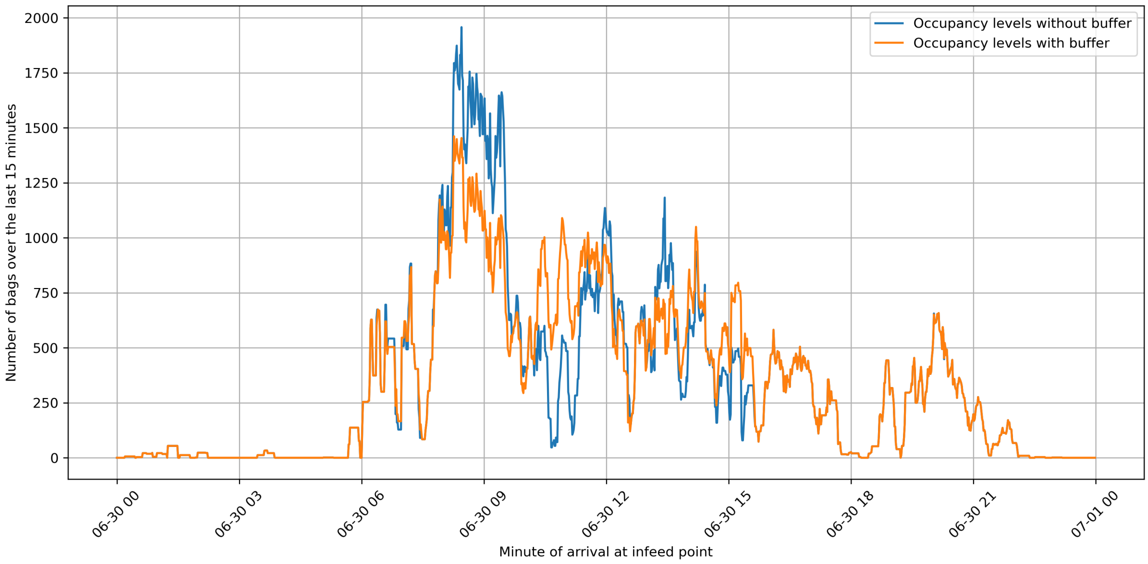


Figure B.1: Transfer infeed point occupancy levels for Configuration 1 and Base 1

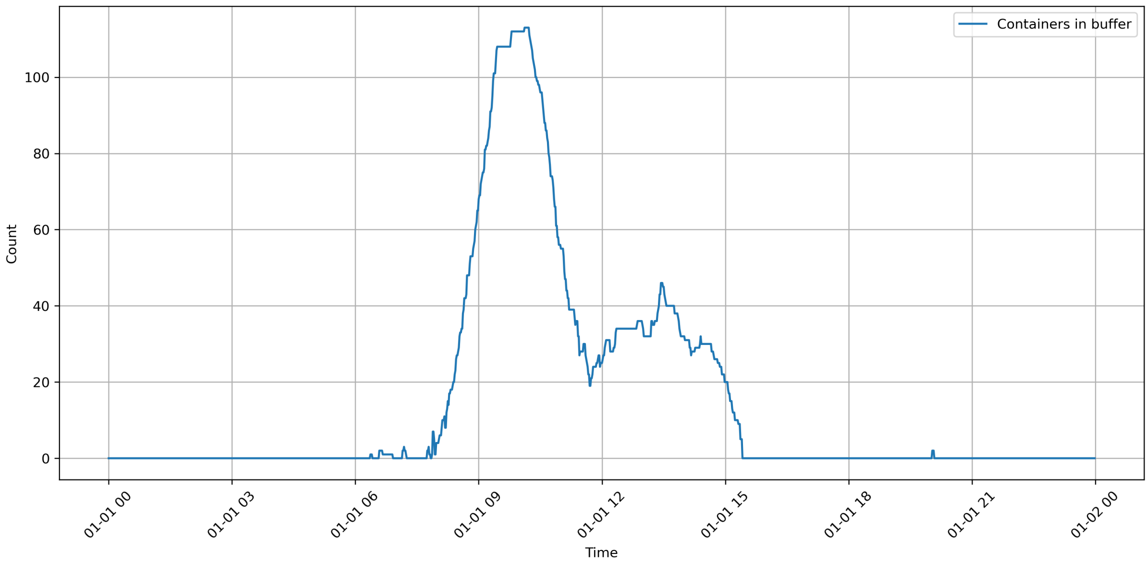


Figure B.2: Buffer occupancy levels for Configuration 1

Configuration 2

Configuration	Option	Alternative	Scenario	Peak of bags	Peak shaving in %	Peak in buffer	Buffered containers	AGVs
2	1: Fixed	1: Early	2: Reduced capacity	1454	25.7%	114	205	26

Table B.18: Results of Configuration 2

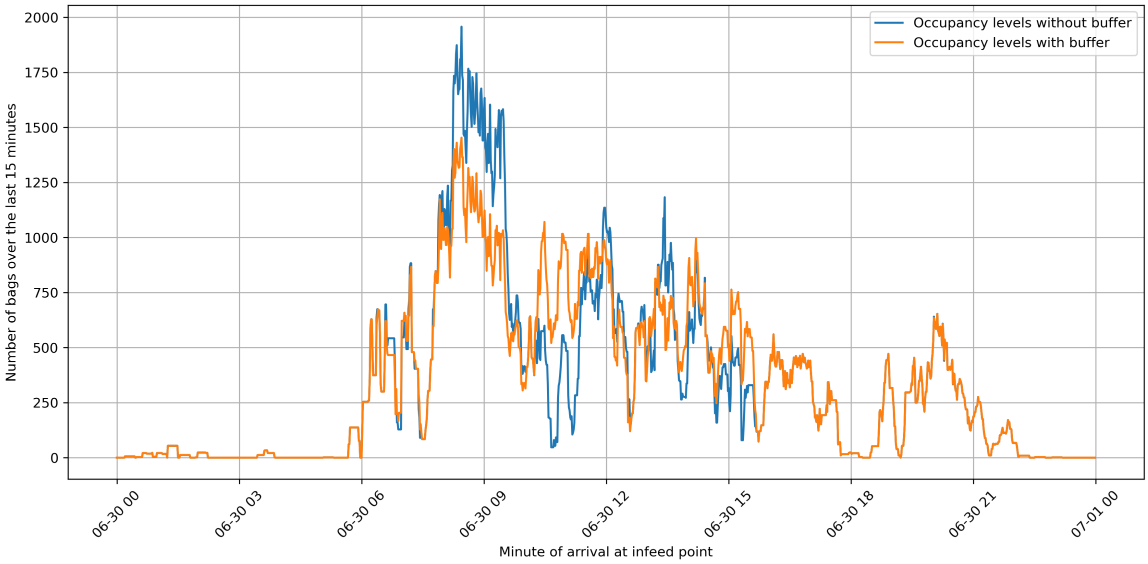


Figure B.3: Transfer infeed point occupancy levels for Configuration 2 and Base 2

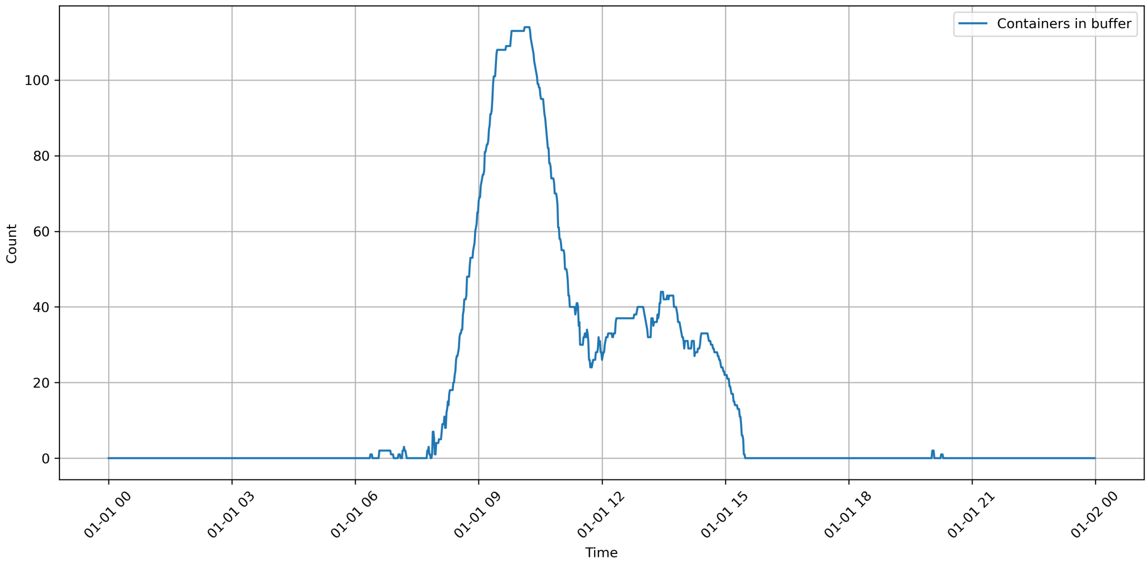


Figure B.4: Buffer occupancy levels for Configuration 2

Configuration 3

Configuration	Option	Alternative	Scenario	Peak of bags	Peak shaving in %	Peak in buffer	Buffered containers	AGVs
3	1: Fixed	1: Early	3: Increased inflow	1734	24.0%	138	288	34

Table B.19: Results of Configuration 3

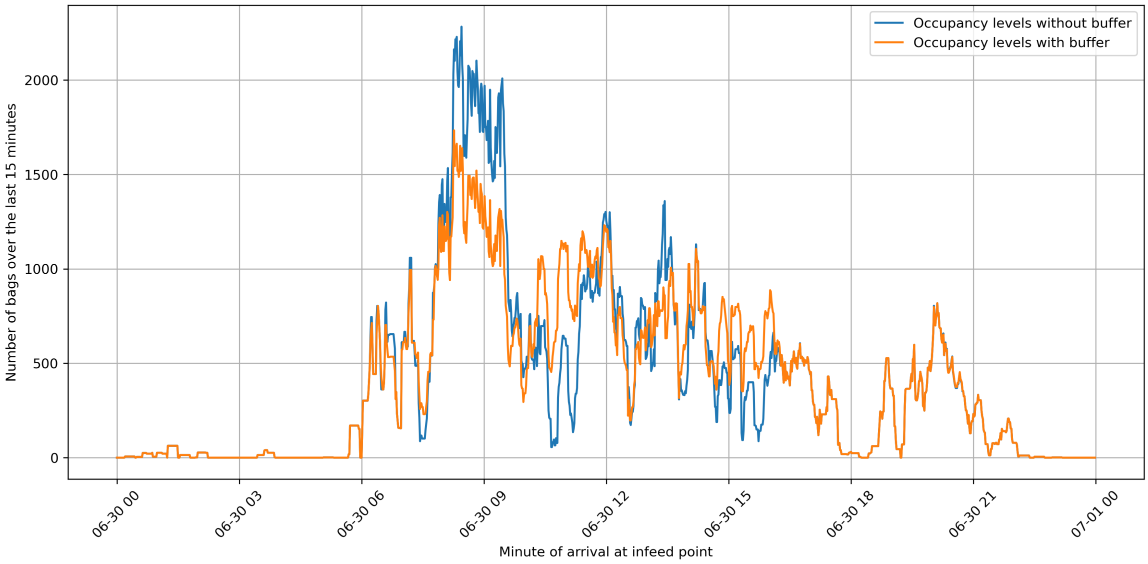


Figure B.5: Transfer infeed point occupancy levels for Configuration 3 and Base 3

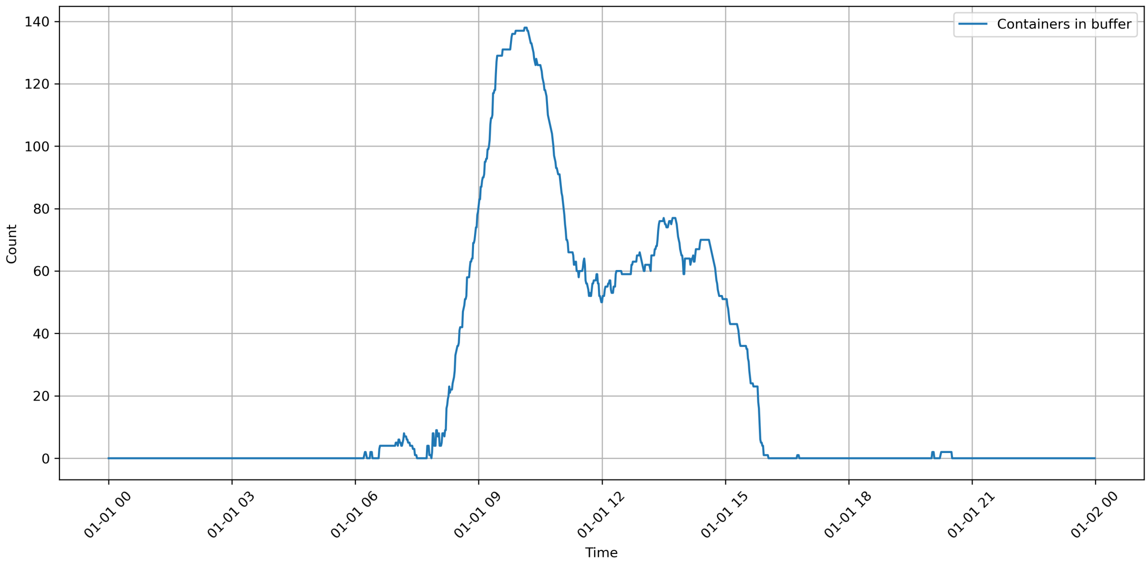


Figure B.6: Buffer occupancy levels for Configuration 3

Configuration 4

Configuration	Option	Alternative	Scenario	Peak of bags	Peak shaving in %	Peak in buffer	Buffered containers	AGVs
4	1: Fixed	2: Optimized	1: Normal	1454	25.7%	116	199	28

Table B.20: Results of Configuration 4

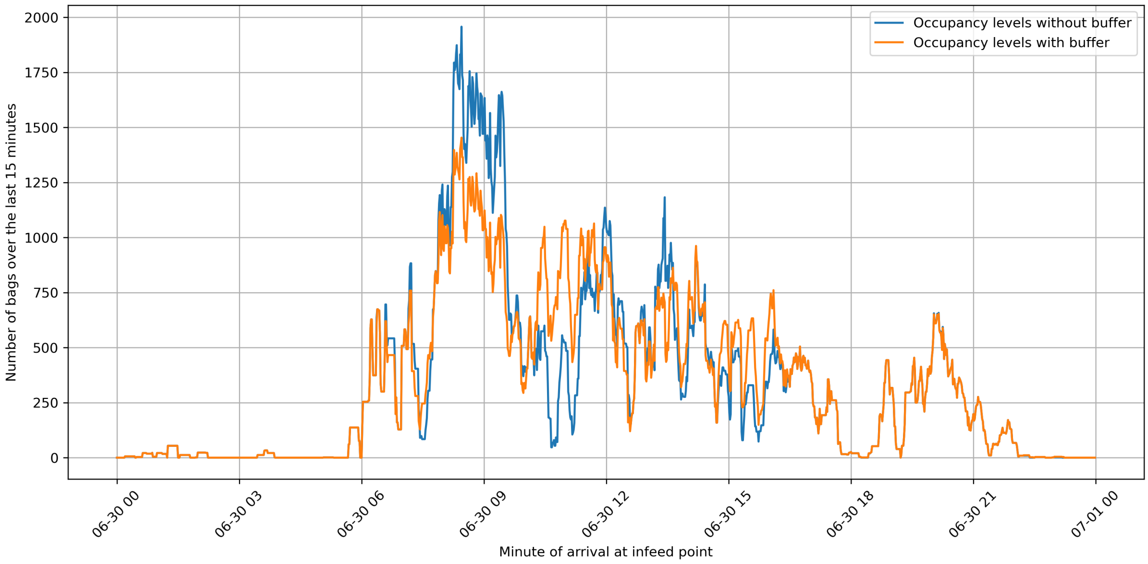


Figure B.7: Transfer infeed point occupancy levels for Configuration 4 and Base 1

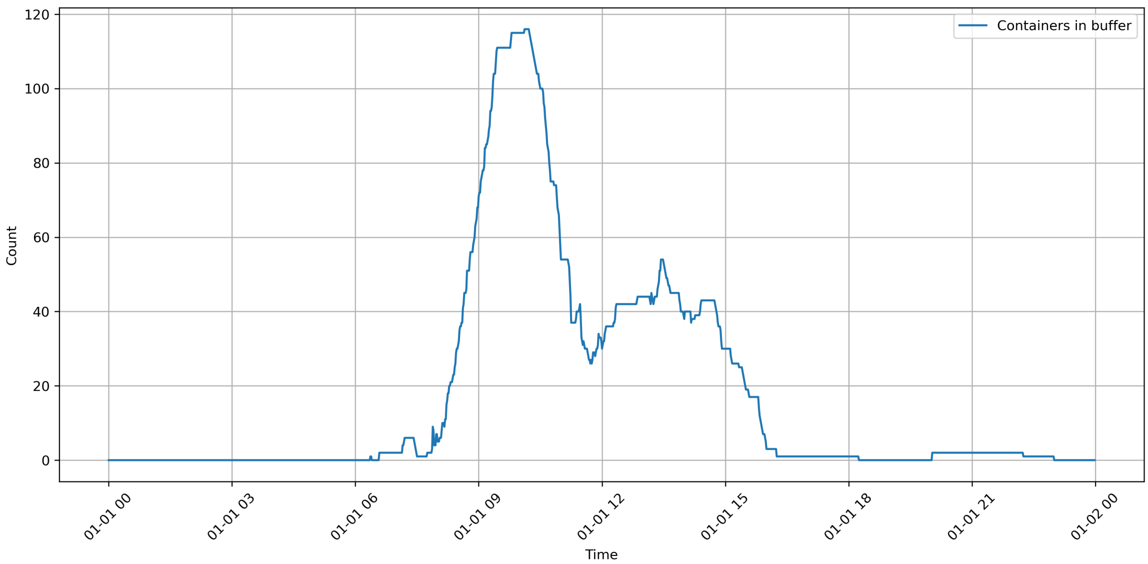


Figure B.8: Buffer occupancy levels for Configuration 4

Configuration 5

Configuration	Option	Alternative	Scenario	Peak of bags	Peak shaving in %	Peak in buffer	Buffered containers	AGVs
5	1: Fixed	2: Optimized	2: Reduced capacity	1464	25.2%	117	205	28

Table B.21: Results of Configuration 5

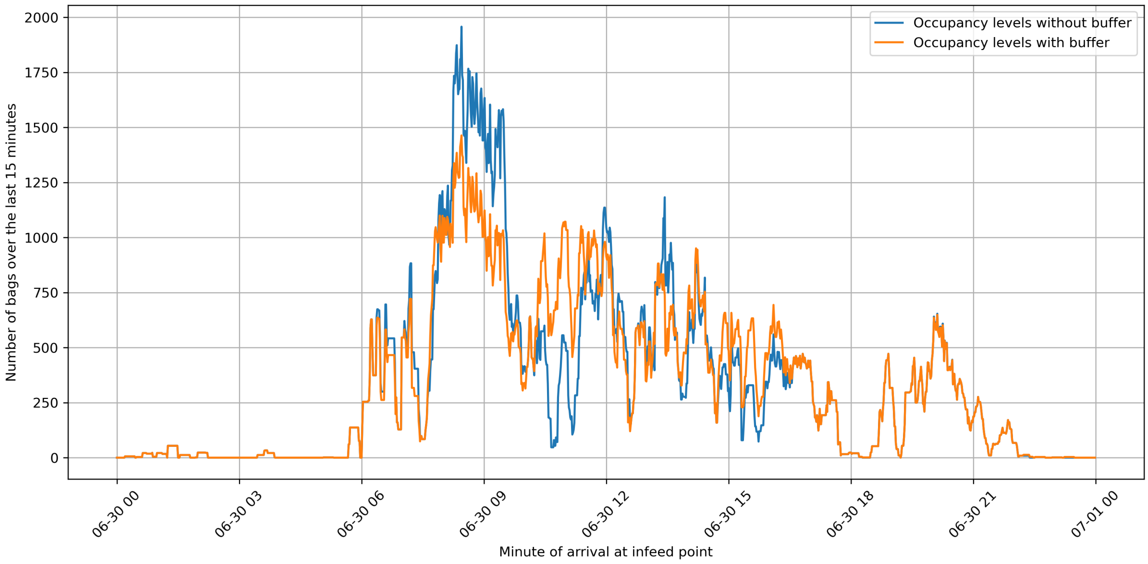


Figure B.9: Transfer infeed point occupancy levels for Configuration 5 and Base 2

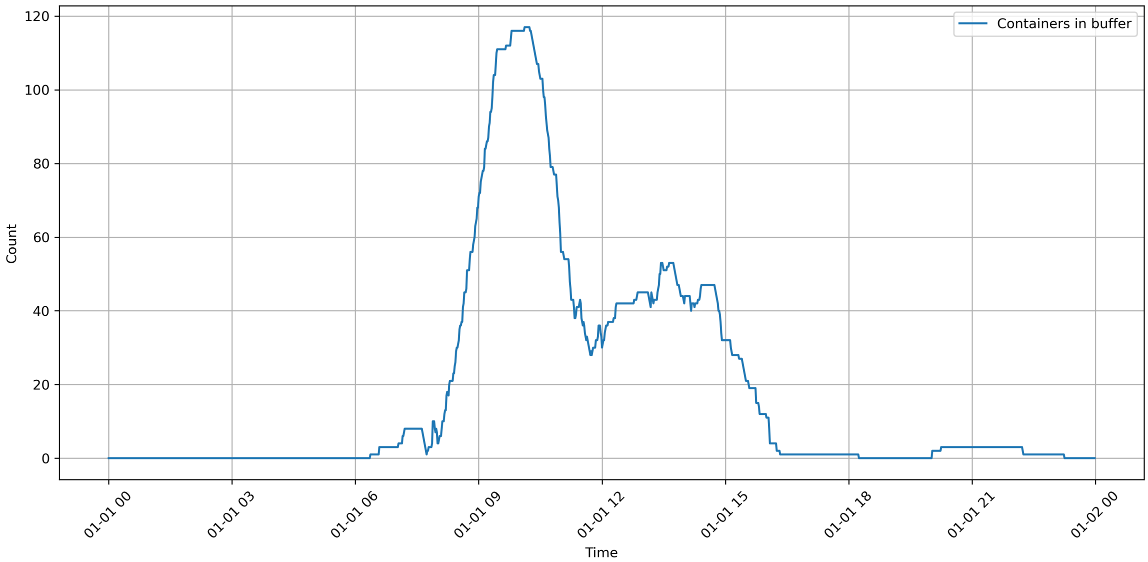


Figure B.10: Buffer occupancy levels for Configuration 5

Configuration 6

Configuration	Option	Alternative	Scenario	Peak of bags	Peak shaving in %	Peak in buffer	Buffered containers	AGVs
6	1: Fixed	2: Optimized	3: Increased inflow	1736	24.0%	142	288	35

Table B.22: Results of Configuration 6

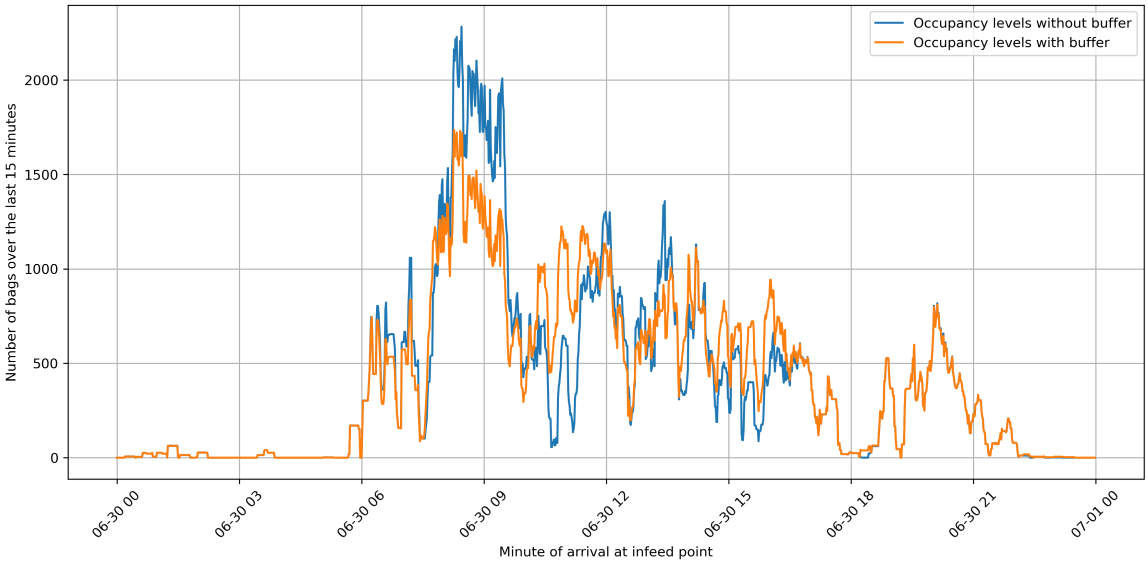


Figure B.11: Transfer infeed point occupancy levels for Configuration 6 and Base 3

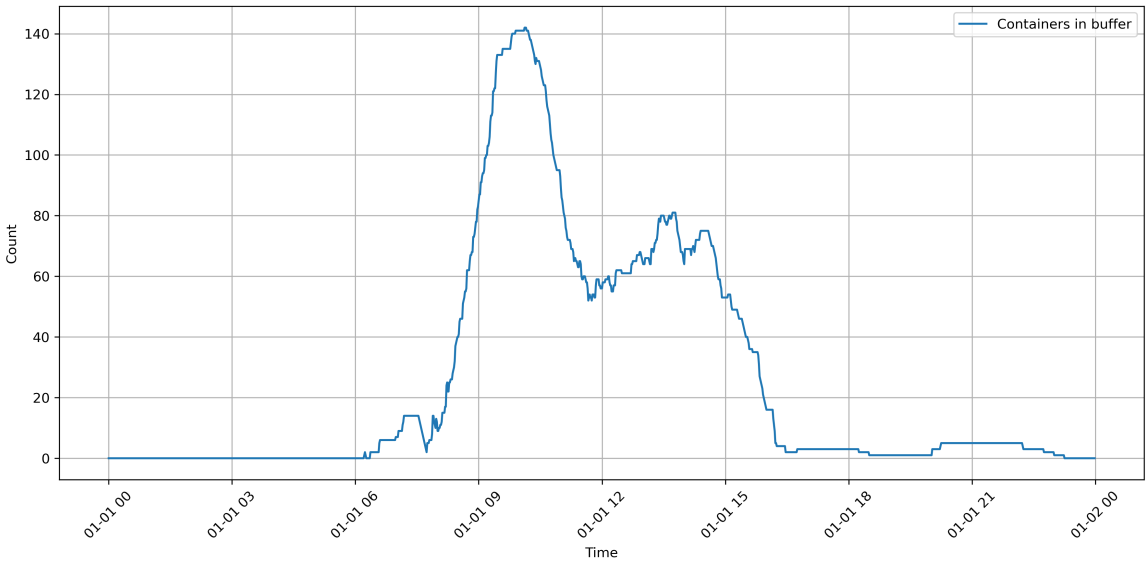


Figure B.12: Buffer occupancy levels for Configuration 6

Configuration 7

Configuration	Option	Alternative	Scenario	Peak of bags	Peak shaving in %	Peak in buffer	Buffered containers	AGVs
7	2: Polynomial	1: Early	1: Normal	1492	23.8%	104	257	33

Table B.23: Results of Configuration 7

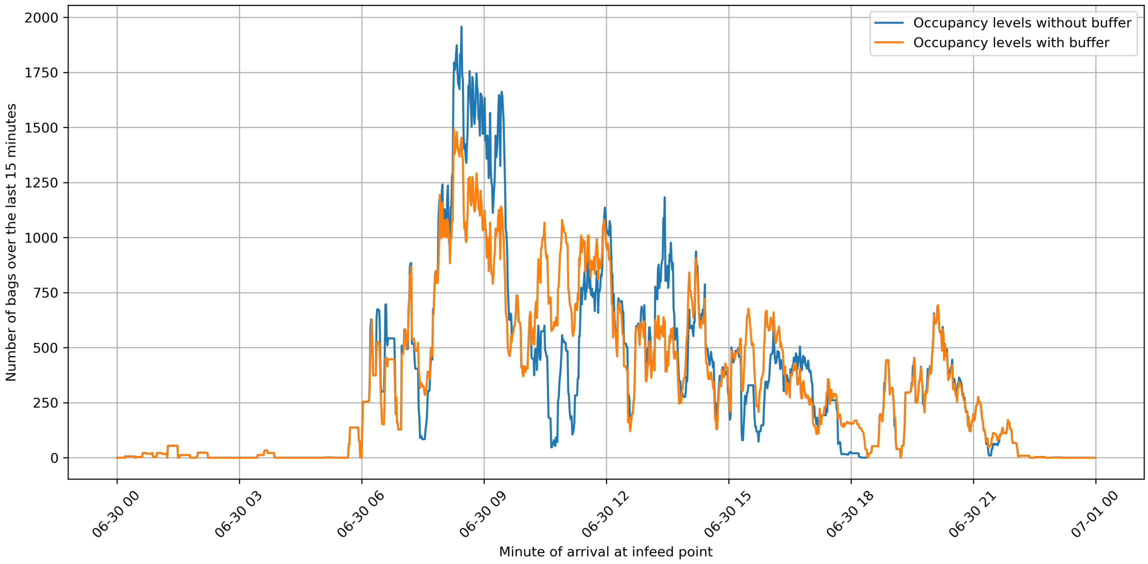


Figure B.13: Transfer infeed point occupancy levels for Configuration 7 and Base 1

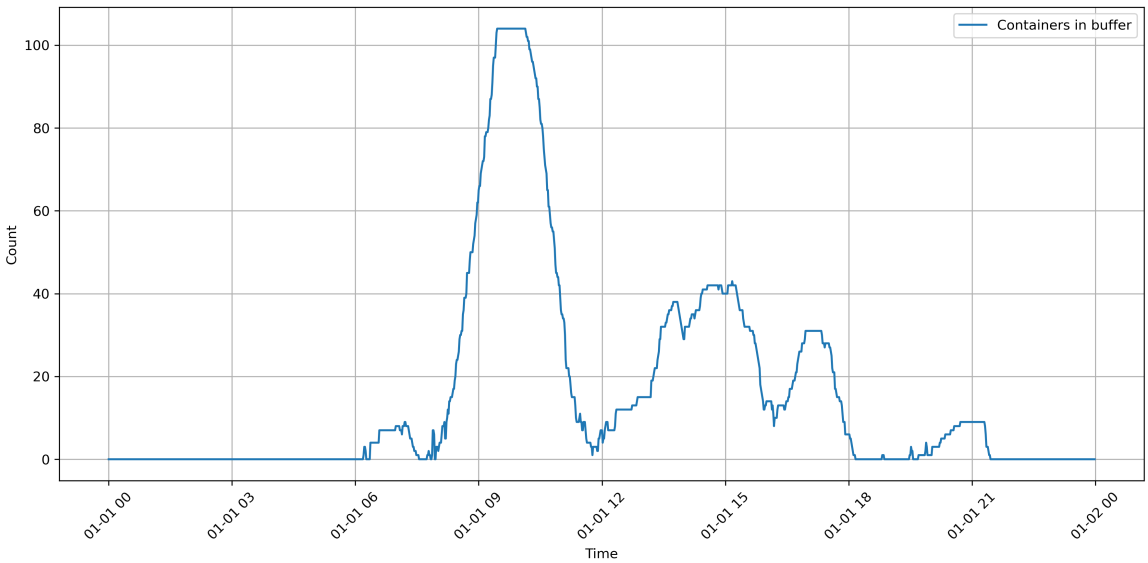


Figure B.14: Buffer occupancy levels for Configuration 7

Configuration 8

Configuration	Option	Alternative	Scenario	Peak of bags	Peak shaving in %	Peak in buffer	Buffered containers	AGVs
8	2: Polynomial	1: Early	2: Reduced capacity	1484	24.2%	105	259	32

Table B.24: Results of Configuration 8

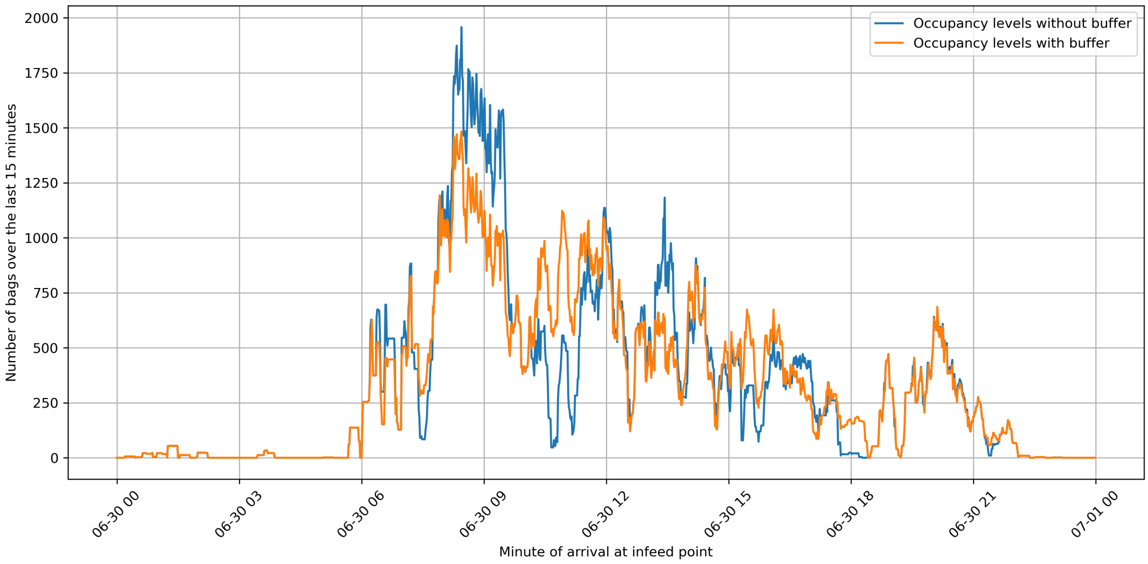


Figure B.15: Transfer infeed point occupancy levels for Configuration 8 and Base 2

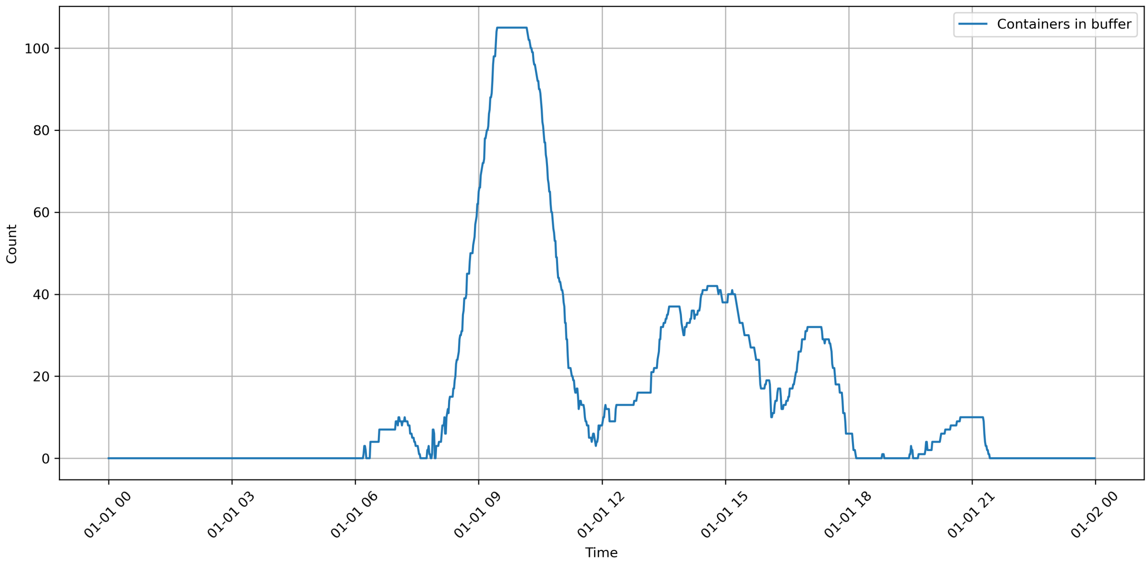


Figure B.16: Buffer occupancy levels for Configuration 8

Configuration 9

Configuration	Option	Alternative	Scenario	Peak of bags	Peak shaving in %	Peak in buffer	Buffered containers	AGVs
9	2: Polynomial	1: Early	3: Increased inflow	1819	20.3%	127	311	40

Table B.25: Results of Configuration 9

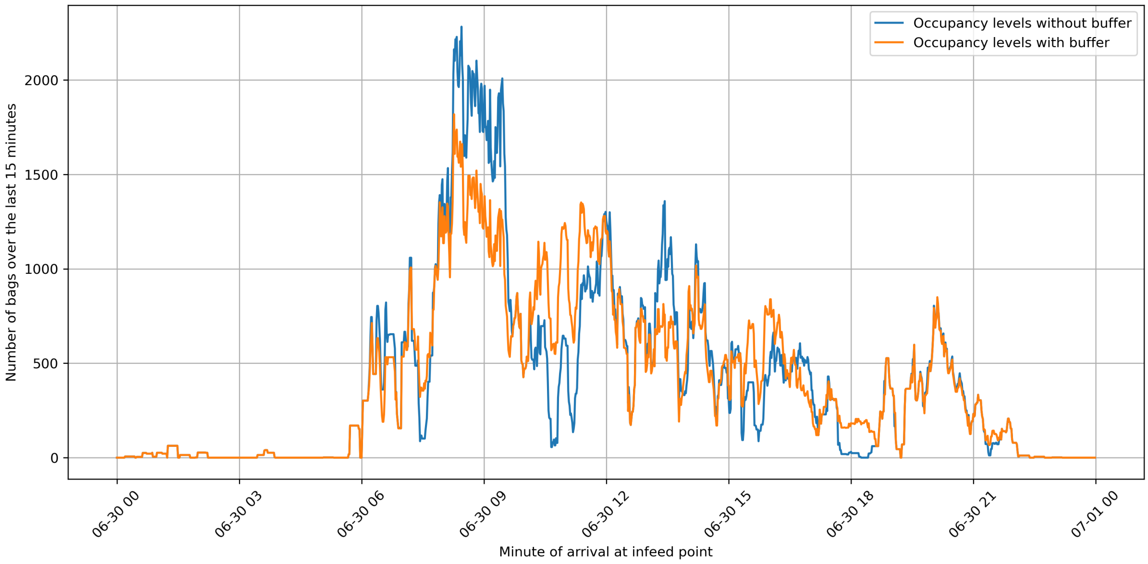


Figure B.17: Transfer infeed point occupancy levels for Configuration 9 and Base 3

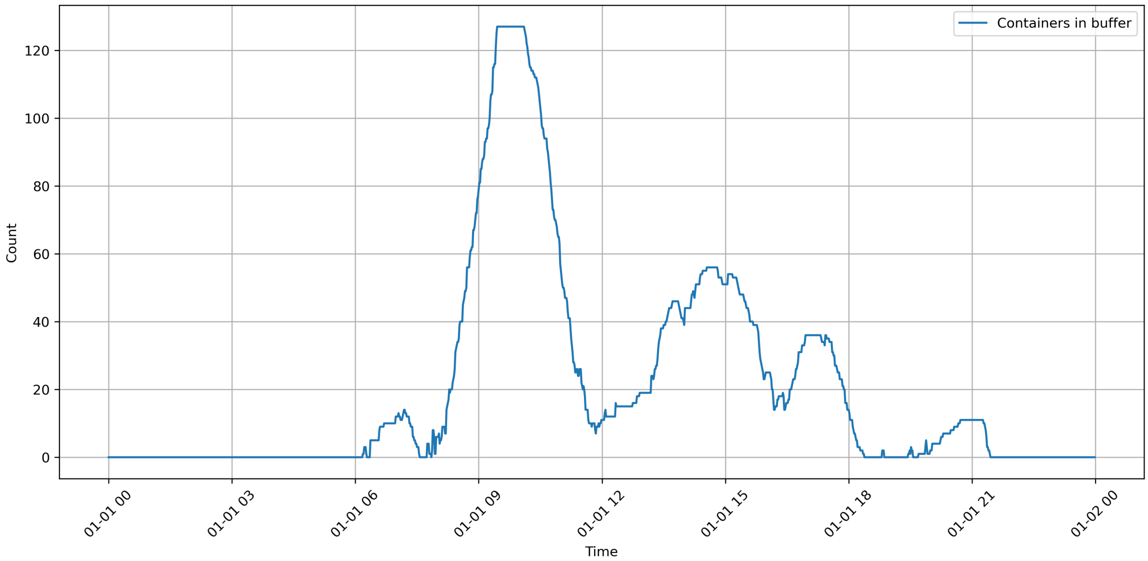


Figure B.18: Buffer occupancy levels for Configuration 9

Configuration 10

Configuration	Option	Alternative	Scenario	Peak of bags	Peak shaving in %	Peak in buffer	Buffered containers	AGVs
10	2: Polynomial	2: Optimized	1: Normal	1433	26.8%	118	251	31

Table B.26: Results of Configuration 10

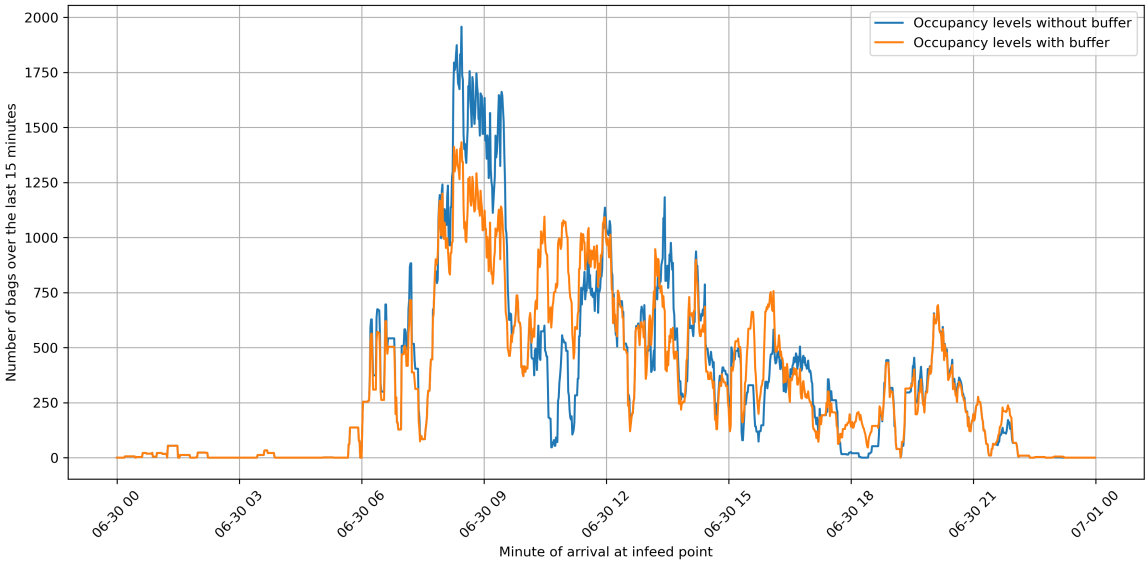


Figure B.19: Transfer infeed point occupancy levels for Configuration 10 and Base 1

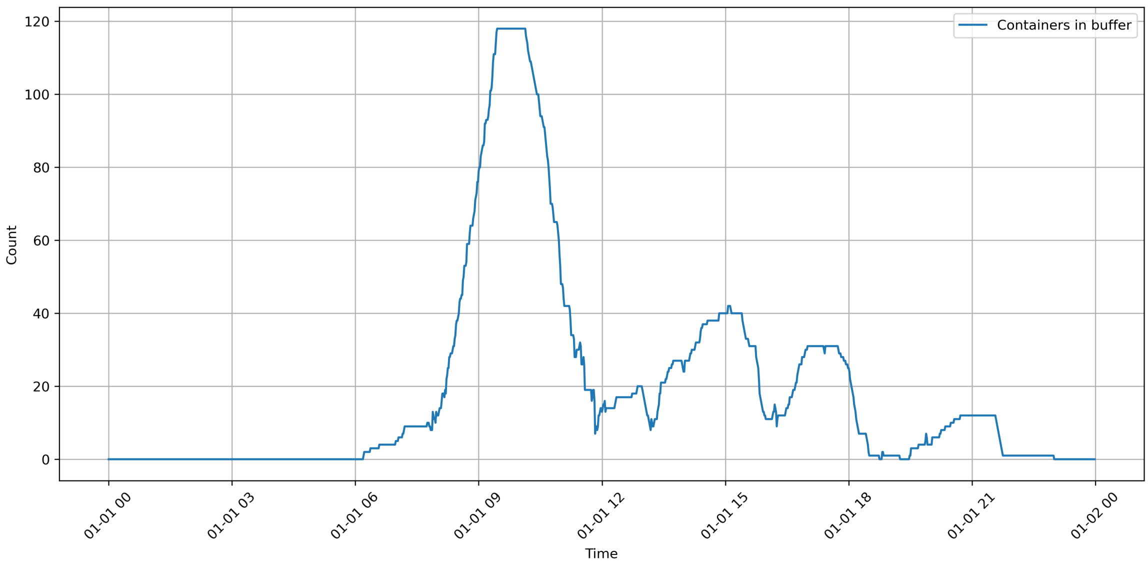


Figure B.20: Buffer occupancy levels for Configuration 10

Configuration 11

Configuration	Option	Alternative	Scenario	Peak of bags	Peak shaving in %	Peak in buffer	Buffered containers	AGVs
11	2: Polynomial	2: Optimized	2: Reduced capacity	1459	25.5%	115	251	31

Table B.27: Results of Configuration 11

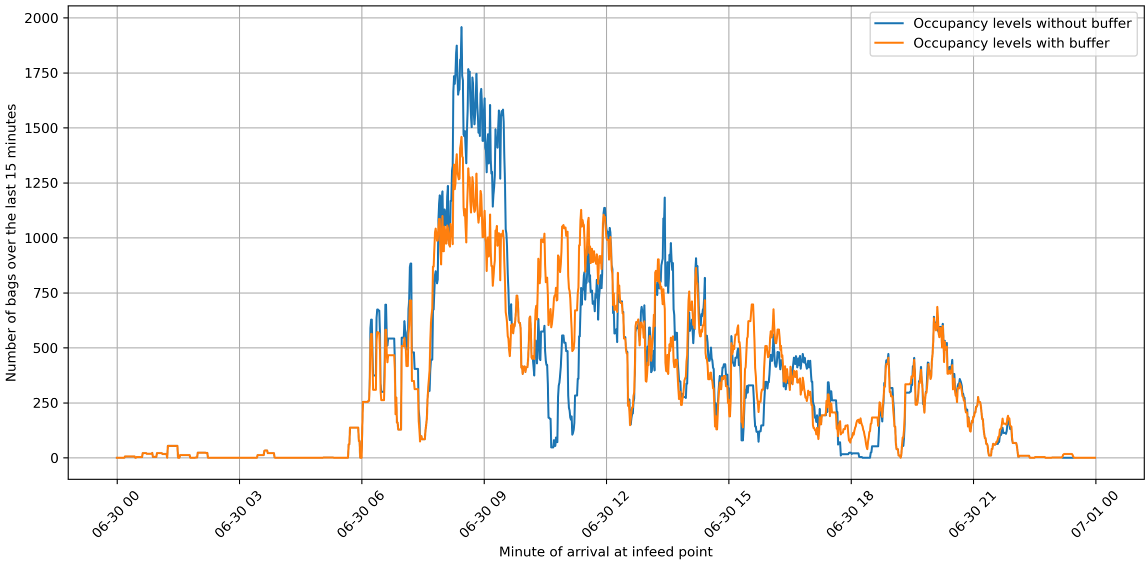


Figure B.21: Transfer infeed point occupancy levels for Configuration 11 and Base 2

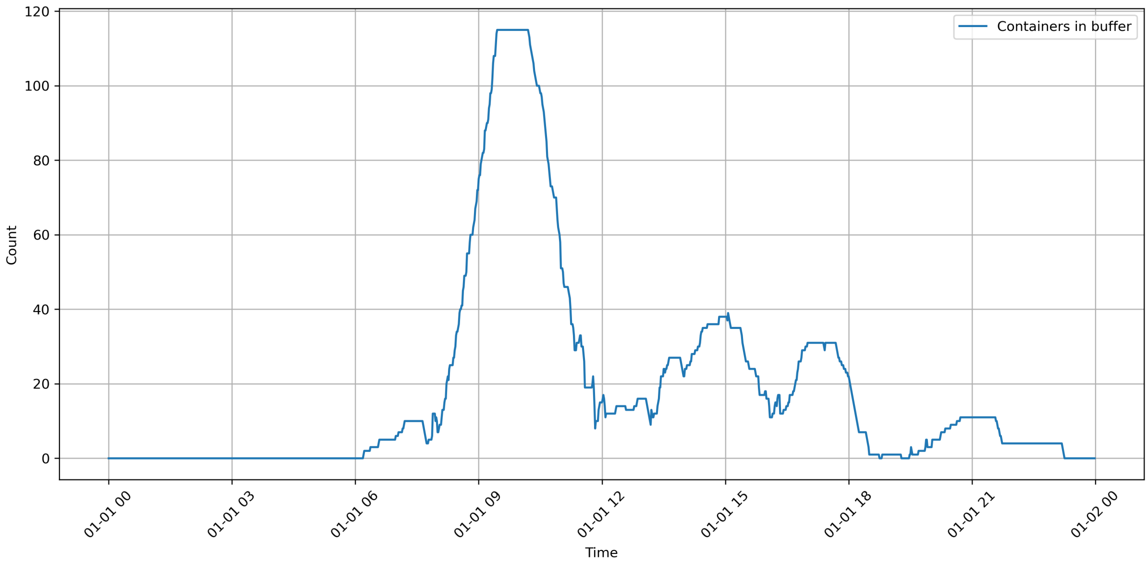


Figure B.22: Buffer occupancy levels for Configuration 11

Configuration 12

Configuration	Option	Alternative	Scenario	Peak of bags	Peak shaving in %	Peak in buffer	Buffered containers	AGVs
12	2: Polynomial	2: Optimized	3: Increased inflow	1663	27.2%	139	300	39

Table B.28: Results of Configuration 12

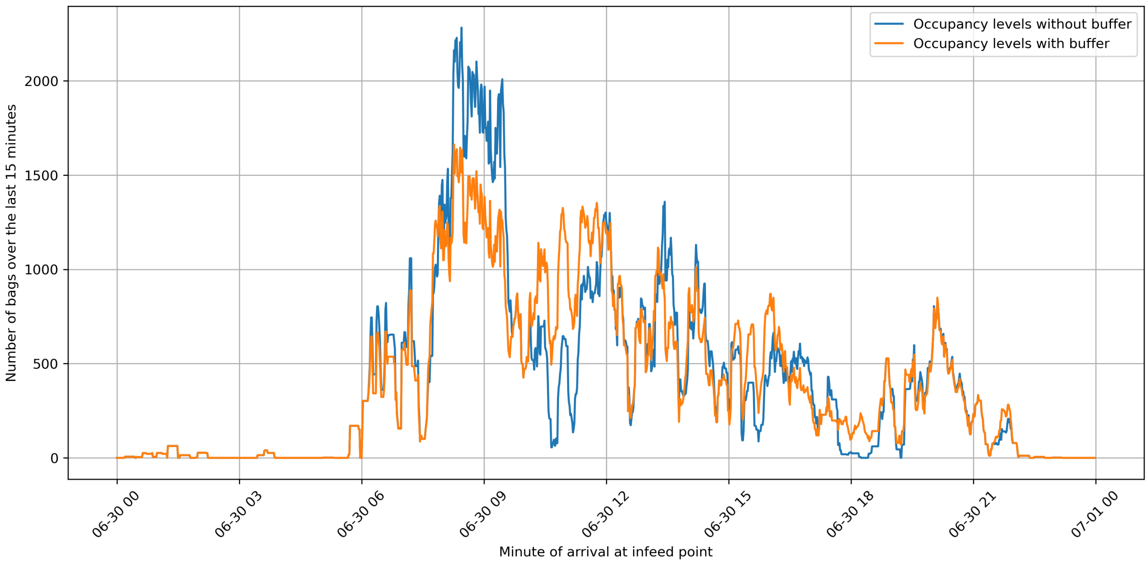


Figure B.23: Transfer infeed point occupancy levels for Configuration 12 and Base 3

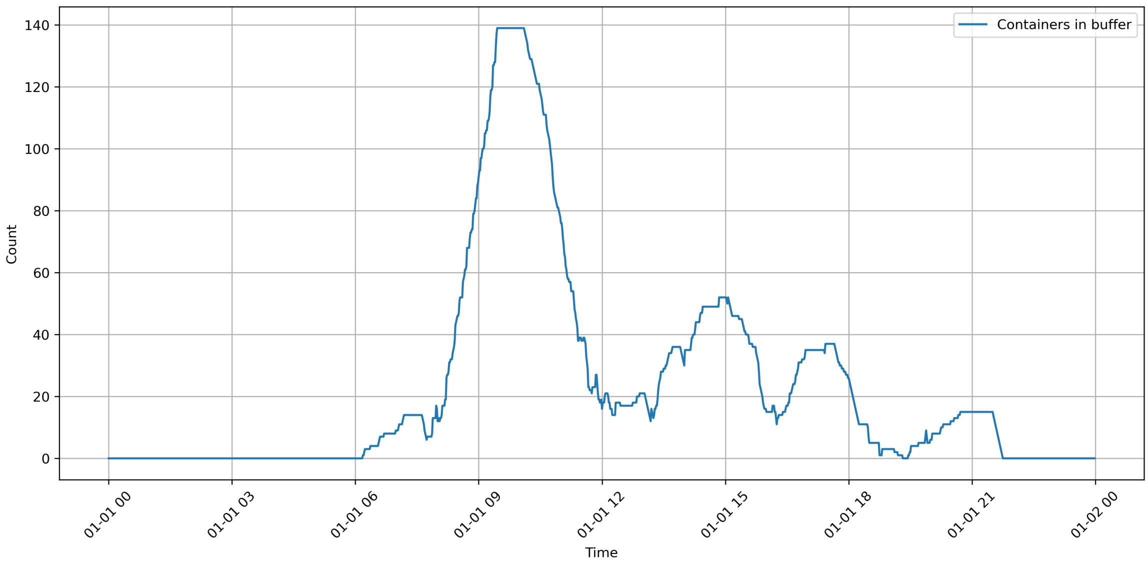


Figure B.24: Buffer occupancy levels for Configuration 12

Configuration 13

Configuration	Option	Alternative	Scenario	Peak of bags	Peak shaving in %	Peak in buffer	Buffered containers	AGVs
13	3: Combination	1: Early	1: Normal	1492	23.8%	104	257	33

Table B.29: Results of Configuration 13

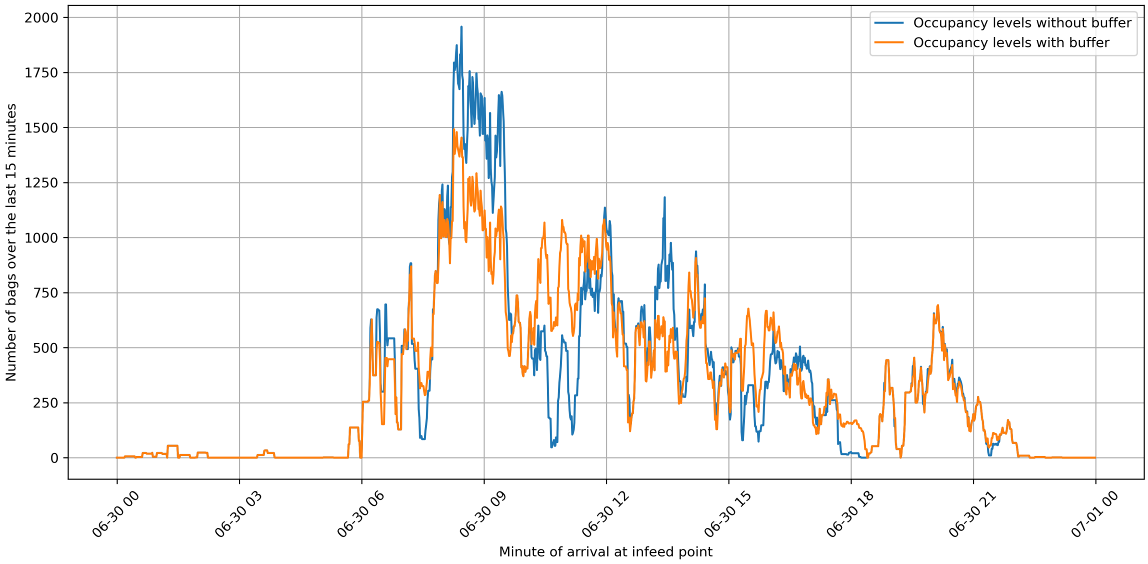


Figure B.25: Transfer infeed point occupancy levels for Configuration 13 and Base 1

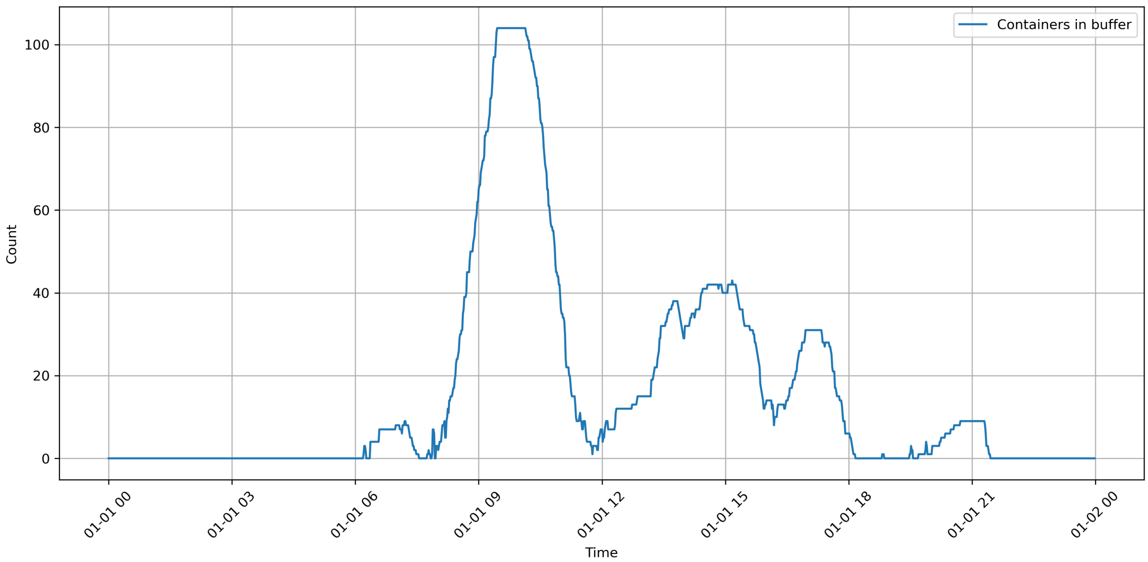


Figure B.26: Buffer occupancy levels for Configuration 13

Configuration 14

Configuration	Option	Alternative	Scenario	Peak of bags	Peak shaving in %	Peak in buffer	Buffered containers	AGVs
14	3: Combination	1: Early	2: Reduced capacity	1484	24.2%	105	259	32

Table B.30: Results of Configuration 14

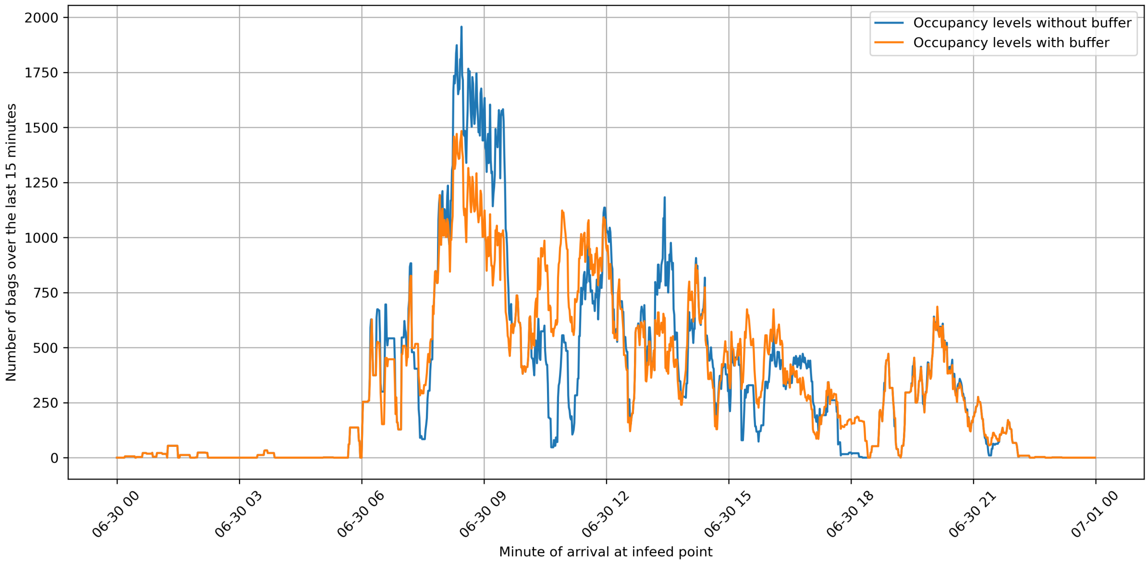


Figure B.27: Transfer infeed point occupancy levels for Configuration 14 and Base 2

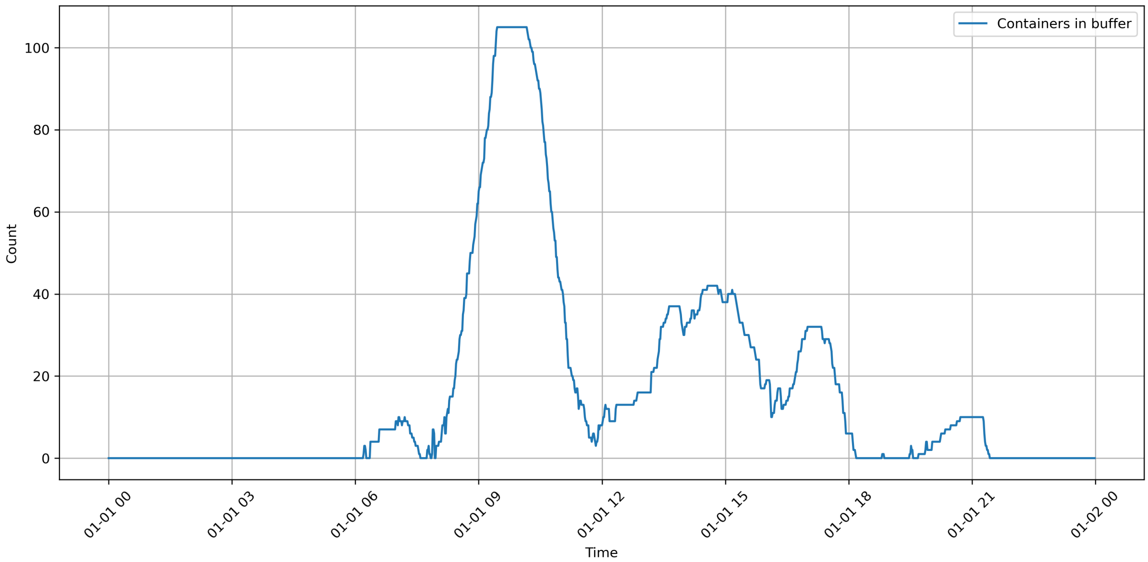


Figure B.28: Buffer occupancy levels for Configuration 14

Configuration 15

Configuration	Option	Alternative	Scenario	Peak of bags	Peak shaving in %	Peak in buffer	Buffered containers	AGVs
15	3: Combination	1: Early	3: Increased inflow	1819	20.3%	127	311	40

Table B.31: Results of Configuration 15

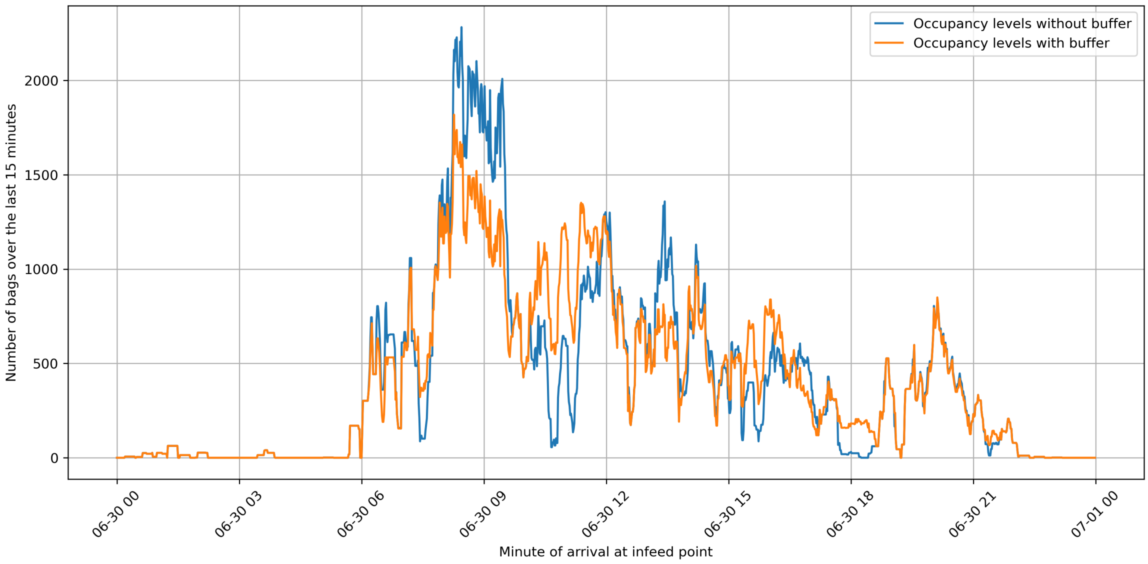


Figure B.29: Transfer infeed point occupancy levels for Configuration 15 and Base 3

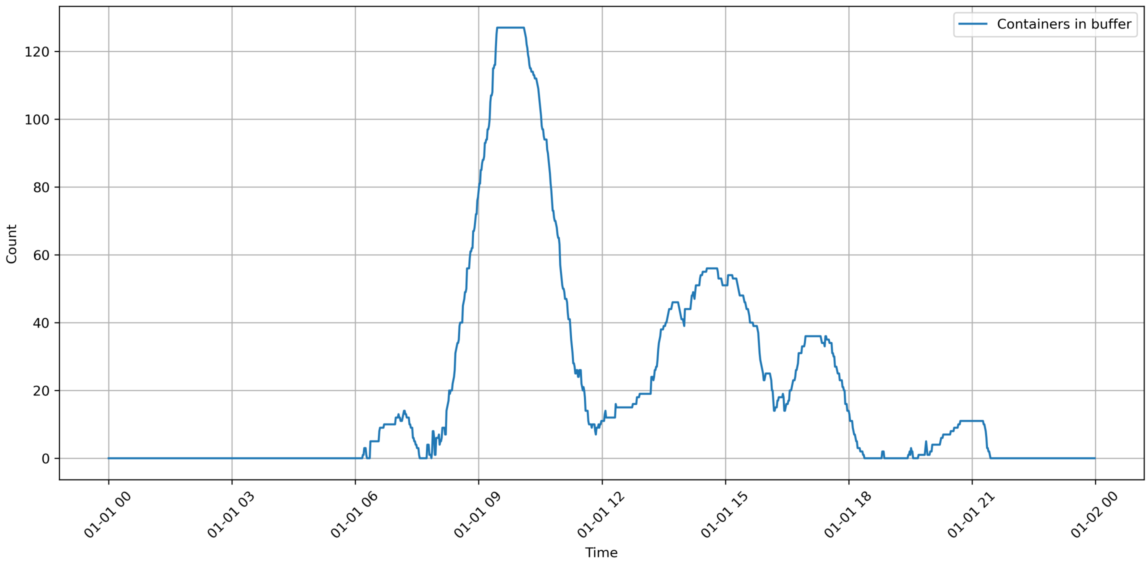


Figure B.30: Buffer occupancy levels for Configuration 15

Configuration 16

Configuration	Option	Alternative	Scenario	Peak of bags	Peak shaving in %	Peak in buffer	Buffered containers	AGVs
16	3: Combination	2: Optimized	1: Normal	1433	26.8%	118	251	31

Table B.32: Results of Configuration 16

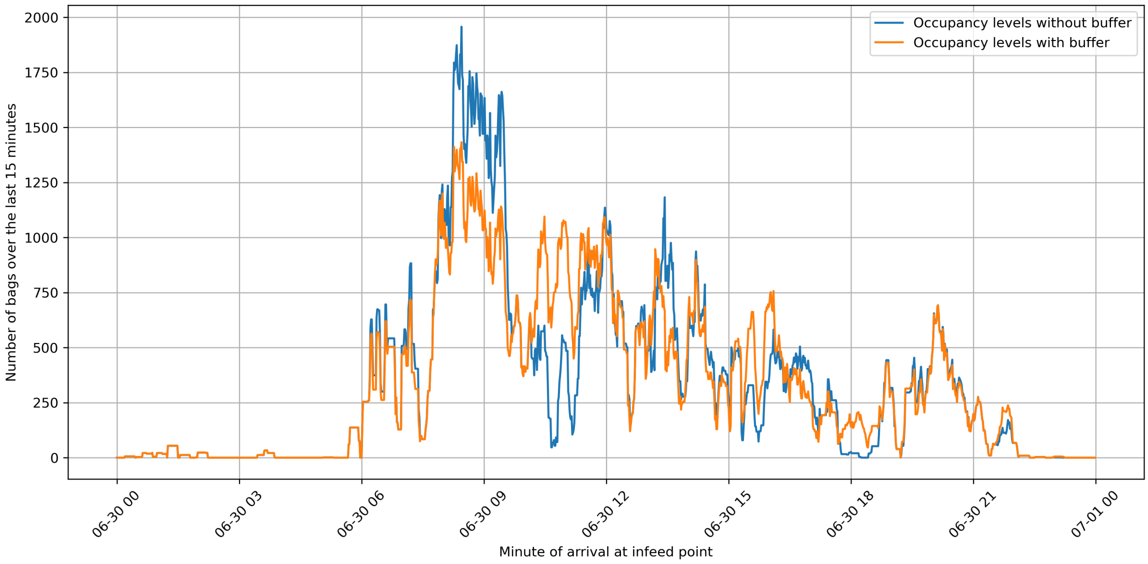


Figure B.31: Transfer infeed point occupancy levels for Configuration 16 and Base 1

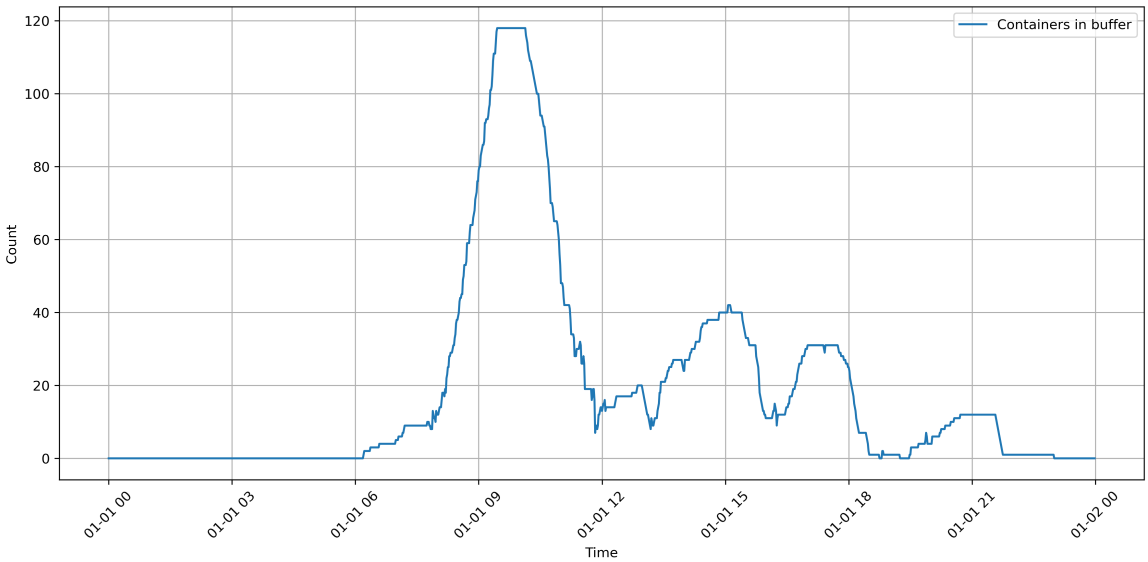


Figure B.32: Buffer occupancy levels for Configuration 16

Configuration 17

Configuration	Option	Alternative	Scenario	Peak of bags	Peak shaving in %	Peak in buffer	Buffered containers	AGVs
17	3: Combination	2: Optimized	2: Reduced capacity	1459	25.5%	115	251	31

Table B.33: Results of Configuration 17

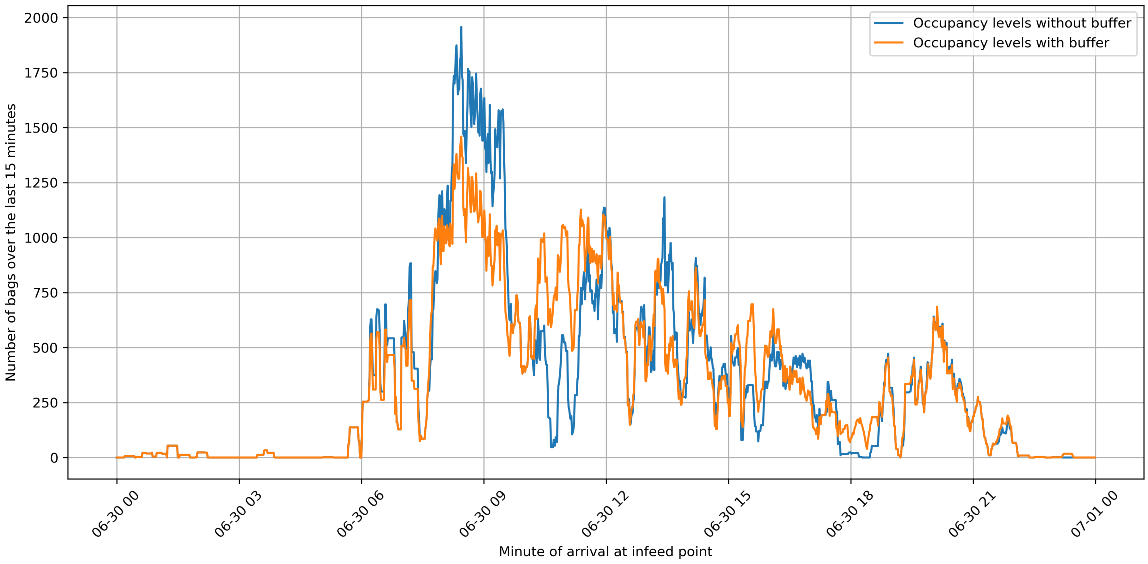


Figure B.33: Transfer infeed point occupancy levels for Configuration 17 and Base 2

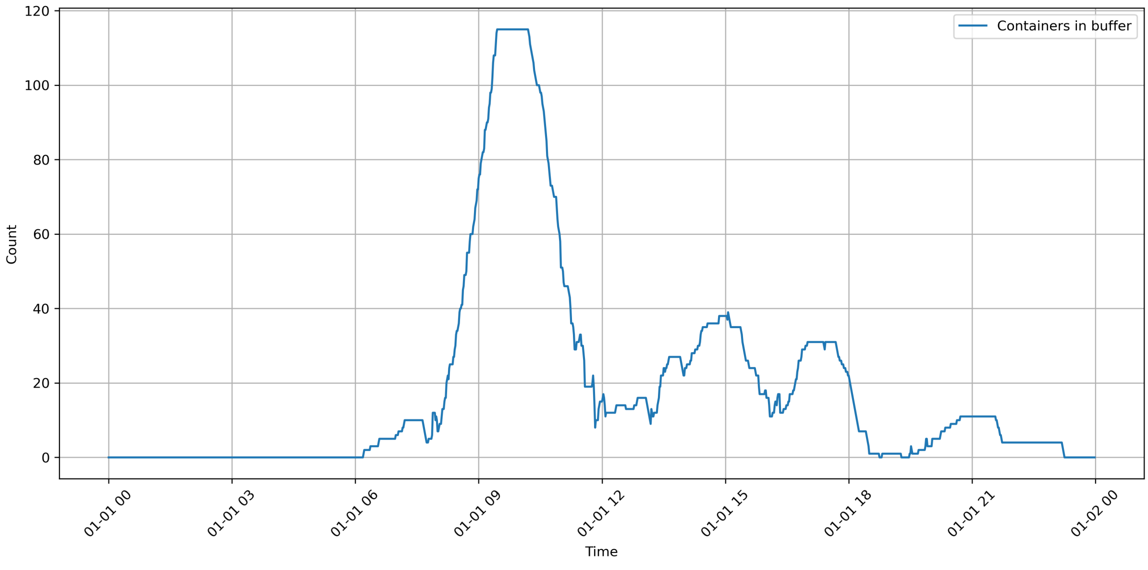


Figure B.34: Buffer occupancy levels for Configuration 17

Configuration 18

Configuration	Option	Alternative	Scenario	Peak of bags	Peak shaving in %	Peak in buffer	Buffered containers	AGVs
18	3: Combination	2: Optimized	3: Increased inflow	1663	27.2%	139	300	39

Table B.34: Results of Configuration 18

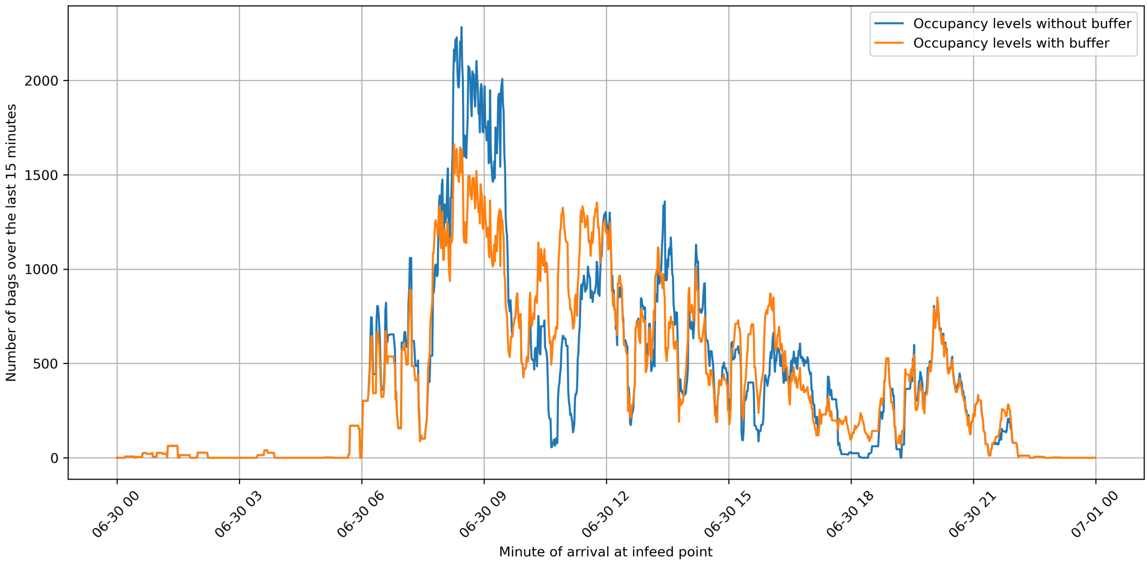


Figure B.35: Transfer infeed point occupancy levels for Configuration 18 and Base 3

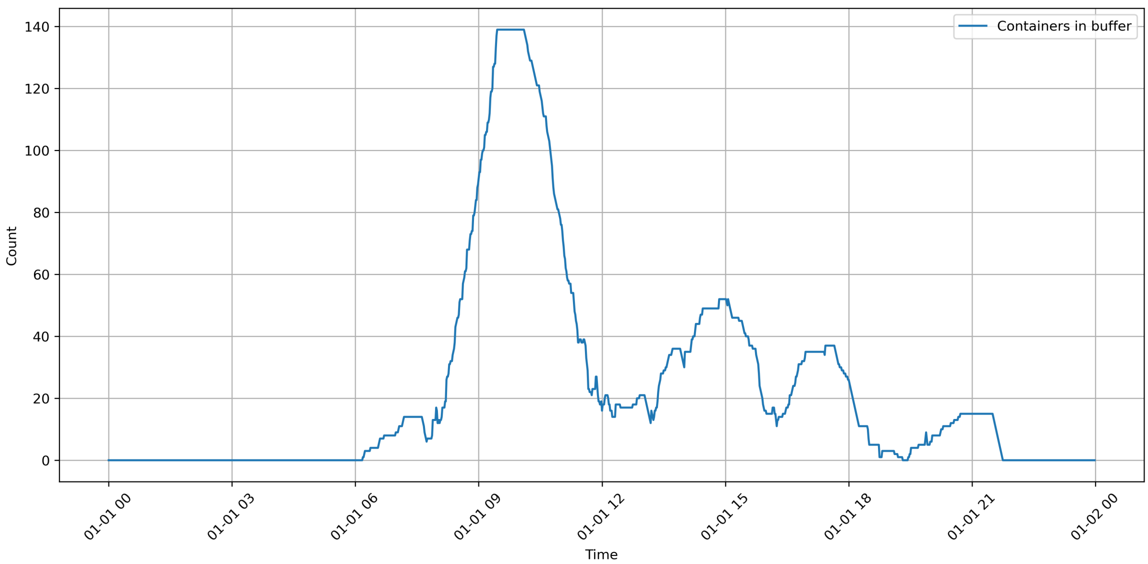


Figure B.36: Buffer occupancy levels for Configuration 18

References

Aviation Week Network. (2024). *Changi airport trialing new autonomous aurrigo uld dolly*. Retrieved February 9, 2024, from https://aviationweek.com/air-transport/airports-networks/changi-airport-trialing-new-autonomous-aurrigo-uld-dolly

Bagagekarretje met koffers [[Accessed 12-09-2024]]. (n.d.).

Baggage container [Accessed 12-09-2024]. (n.d.). https://qisexport.en.made-in-china.com/product/uBinCLSEhRrw/China-Air-Shipment-D-Container-Aircraft-Transport-Aviation-Container.html

Baggage Make-up and Unloading Systems | Beumer Group [[Accessed 12-09-2024]]. (n.d.).

Barth, T. C., Holm, J. T., Larsen, J. L., & Pisinger, D. (2021). Optimization of transfer baggage handling in a major transit airport. *Operations Research Forum*, *2*, 1–35.

Barth, T. C., & Larsen, J. T. H. J. L. (2013). A model for transfer baggage handling at airports. *Technical University of Denmark: Lyngby, Denmark*.

Bunnik, H. (2018, March). *Ism platform: Simulation rules for schiphol* (tech. rep.) (Internal Schiphol Report). INCONTROL Simulation Solutions.

Chua, K. H., Lim, Y. S., & Morris, S. (2016). Energy storage system for peak shaving. *International Journal of Energy Sector Management*, *10*(1), 3–18.

Clausen, T., & Pisinger, D. (2010). Dynamic routing of short transfer baggage. *DTU Management Engineering, Tech. Rep. 10*.

Darwazeh, D., Duquette, J., Gunay, B., Wilton, I., & Shillinglaw, S. (2022). Review of peak load management strategies in commercial buildings. *Sustainable Cities and Society*, *77*, 103493.

Dijkhuizen. (2024). Simulations at schipghol [Accessed on: 12-11-2024; Schiphol Airport department or directory location].

Disney, S. M., Towill, D. R., & Van de Velde, W. (2004). Variance amplification and the golden ratio in production and inventory control. *International Journal of Production Economics*, *90*(3), 295–309.

Duell, M. (n.d.). What happens to your bag at airports? Behind the scenes at Heathrow T5 — dailymail.co.uk [[Accessed 12-09-2024]].

Ekeocha, R., & Ushe, S. A. (2018). Application of queuing process in the optimization of baggage handling system of murtala muhammed international airport. *Int. J. Sci.*, *7*(5), 97–103.

Fildes, R., Goodwin, P., & Lawrence, M. (2006). The design features of forecasting support systems and their effectiveness. *Decision Support Systems*, *42*(1), 351–361.

Frey, M. (2014). *Models and methods for optimizing baggage handling at airports* [Doctoral dissertation, Technische Universität München].

Frey, M., Kiermaier, F., & Kolisch, R. (2017). Optimizing inbound baggage handling at airports. *Transportation Science*, *51*(4), 1210–1225.

Gao, Z., Ji, W., Guo, L., Fan, X., & Wang, J. (2021). Thermo-economic analysis of the integrated bidirectional peak shaving system consisted by liquid air energy storage and combined cycle power plant. *Energy conversion and management*, *234*, 113945.

Gurgul, H., Lach, Ł., Suder, M., & Szpyt, K. (2023). Using trigonometric seasonal models in forecasting the size of withdrawals from automated teller machines. *Entrepreneurial Business and Economics Review*, *11*(3), 181–204.

Hanwu, M., & Juan, L. (2009). Process optimization research on baggage transit system of hub airports. *2009 International Conference on Information Management, Innovation Management and Industrial Engineering*, *3*, 58–61.

Huang, E., Liu, I., & Lin, J. T. (2018). Robust model for the assignment of outgoing flights on airport baggage unloading areas. *Transportation Research Part E: Logistics and Transportation Review*, *115*, 110–125.

Huang, E., Mital, P., Goetschalckx, M., & Wu, K. (2016). Optimal assignment of airport baggage unloading zones to outgoing flights. *Transportation research part E: logistics and transportation review*, *94*, 110–122.

International Air Transport Association (IATA). (2024). Global air travel demand continued its bounce back in 2023 [Accessed: 2024-09-04]. *IATA Pressroom*. https://www.iata.org/en/pressroom/2024-releases/2024-01-31-01/

International Civil Aviation Organization (ICAO). (2022). Latest air traffic forecasts illustrate encouraging recovery and higher growth in global air travel [Accessed: 2024-09-04]. *ICAO Newsroom*. https://www.icao.int/Newsroom/Pages/Latest-air-traffic-forecasts-illustrate-encouraging-recovery-and-higher-growth-in-global-air-travel.aspx

Lin, J. T., Liou, I., & Chiu, C.-C. (2015). A simulation-based analysis for inter release problem in airport baggage handling systems. *Industrial Engineering, Management Science and Applications 2015*, 115–124.

Littler, R., & Whitaker, D. (1997). Estimating staffing requirements at an airport terminal. *Journal of the Operational Research Society*, *48*(2), 124–131.

Monczka, R. M., Trent, R. J., & Handfield, R. B. (2002). *Purchasing and supply chain management*. South Western Educational Publishing.

Naim, I., Mahara, T., & Khan, S. A. (2020). Ranking of univariate forecasting techniques for seasonal time series using analytical hierarchy process. *International Journal of Industrial and Systems Engineering*, *35*(2), 196–215.

Pickering, K. (n.d.). Manchester Airport installs new baggage reclaim carousel - Passenger Terminal Today — passengerterminaltoday.com [[Accessed 12-09-2024]].

Schiphol. (2023). Schematic representation of bhs [Internal PowerPoint presentation].

Schiphol. (2024). Schiphol | An autonomous airport in 2050 — schiphol.nl [[Accessed 19-09-2024]].

Shchelkalin, V. (2014). Hybrid mathematical models and methods of time series forecasting taking into account external factors. *Eastern-European Journal of Enterprise Technologies*, *6*(4).

Silver, E. A., Pyke, D. F., & Thomas, D. J. (2016). *Inventory and production management in supply chains*. CRC press.

Syntetos, A. A., Boylan, J. E., & Disney, S. M. (2009). Forecasting for inventory planning: A 50-year review. *Journal of the Operational Research Society*, *60*(sup1), S149–S160.

Tziovani, L., Hadjidemetriou, L., Charalampous, C., Tziakouri, M., Timotheou, S., & Kyriakides, E. (2021). Energy management and control of a flywheel storage system for peak shaving applications. *IEEE Transactions on Smart Grid*, *12*(5), 4195–4207.

van der Grift, N. (2023, February). *Reducing peak stress on the baggage handling system* [Doctoral dissertation, Technical University of Delft] [Type Thesis].

Van Kampen, T. J., Van Donk, D. P., & Van Der Zee, D.-J. (2010). Safety stock or safety lead time: Coping with unreliability in demand and supply. *International Journal of Production Research*, *48*(24), 7463–7481.

Wang, T., Tian, X., & Wang, Y. (2020). Container slot allocation and dynamic pricing of time-sensitive cargoes considering port congestion and uncertain demand. *Transportation Research Part E: Logistics and Transportation Review*, *144*, 102149.

Yu, C., Xu, C., Li, Y., Yao, S., Bai, Y., Li, J., Wang, L., Wu, W., & Wang, Y. (2021). Time series analysis and forecasting of the hand-foot-mouth disease morbidity in china using an advanced exponential smoothing state space tbats model. *Infection and drug resistance*, 2809–2821.

Comparing Distribution Strategies to Achieve Peak Shaving at Occupancy Levels of the Baggage Handling System of an Airport

B.R.M. Nijdam
Technical University of Delft
Studentnumber: 4486846
Reportnumber: 2024.TIL.9011
Supervisors: C. Roos, A. Napoleone, J. Vleugel

December 9, 2024

Abstract

Amsterdam Airport Schiphol (AAS) faces operational challenges from fluctuating baggage volumes and limited infrastructure, necessitating innovative approaches to optimize baggage handling. This study focuses on strategic buffering of super cold transfer baggage to reduce peak occupancy at transfer infeed points. A simulation model, integrated with TBATS forecasting, evaluates three buffering strategies (fixed target, polynomial, hybrid) and two reintroduction strategies (early, optimized release) under normal, disrupted, and high-demand scenarios.

The results demonstrate peak occupancy reductions of up to 26.8%, with the polynomial buffering strategy and optimized release proving the most effective, though requiring additional AGV resources. Fixed target strategies offer a more resource-efficient alternative while maintaining substantial improvements in peak shaving. Early release strategies further reduce buffer congestion, optimizing operations for space-constrained environments.

This research provides actionable insights for AAS and similar airports, highlighting trade-offs between efficiency, resource requirements, and operational constraints. The findings contribute to scalable solutions for managing growing baggage volumes within existing infrastructure, enhancing system resilience and performance.

1. Introduction

Efficient baggage handling systems (BHS) are crucial for the smooth operation of hub airports, ensuring the timely transfer of passengers' baggage while minimizing errors and delays. At major transit hubs such as AAS, fluctuating baggage volumes during peak times pose significant operational challenges. These fluctuations can lead to imbalances in staffing, strain on sorting systems, and inefficiencies in temporary storage buffers, ultimately disrupting overall airport operations [1, 5].

AAS faces additional constraints due to its existing infrastructure and limited opportunities for expansion. With a fixed number of infeed points and no immediate possibilities for additional capacity, the airport must manage increasing baggage volumes within the current system. This makes optimizing the use of existing resources a critical priority. Addressing peak loads at transfer infeed points is particularly challenging, as these points often experience bottlenecks during periods of high demand.

One promising approach to mitigate these challenges is peak shaving, a strategy aimed at redistributing baggage handling workloads over time to reduce operational strain. Previous studies have explored optimization techniques such as dynamic routing, robust scheduling, and heuristic approaches to address variability in baggage flows [2, 4]. However, limited attention has been given to the strategic buffering and reintroduction of cold transfer baggage—a category of baggage with longer layover times that can be temporarily stored without compromising outbound flight schedules. This category represents a unique opportunity to alleviate peak loads, but research on dynamic and predictive strategies for its management remains scarce [7].

This study aims to address this gap by answering the research question: ***How to support Amsterdam Airport Schiphol to identify the best distribution strategy for super cold transfer baggage through a simulation model?*** To this end, a simulation model was developed to replicate the baggage journey from aircraft arrival to transfer infeed points, incorporating time series forecasting to dynamically inform buffering and release decisions.

The findings provide actionable insights into balancing operational efficiency, resource allocation, and peak shaving effectiveness within the constraints of AAS's existing infrastructure. By identifying optimal buffering strategies for super cold transfer baggage, this research contributes to more resilient baggage handling systems and supports airports in managing growing demands without requiring costly infrastructure expansion.

2. Literature

Efficient BHS are a cornerstone of modern airport operations, facilitating the seamless transfer of baggage across check-in counters, transit flights, and reclaim areas. The rising complexity of airport operations, driven by increasing passenger volumes and stricter security requirements, necessitates continuous optimization of these systems to enhance efficiency and passenger satisfaction [2, 4].

Extensive research has been conducted on optimizing BHS processes, with a focus on routing, scheduling, and resource allocation. Barth et al. [1] proposed a fuzzy logic model to enhance adaptability in baggage handling operations under fluctuating conditions, demonstrating improvements in resource allocation and system efficiency. Similarly, Huang et al. [6] introduced robust optimization models for assigning baggage unloading zones to outgoing flights, emphasizing stability under operational uncertainties such as flight delays and varying baggage volumes. These studies highlight the importance of adaptability and robust decision-making in managing complex airport operations.

In the context of inbound baggage handling, Frey [3] employed hybrid heuristics, combining GRASP with path-relinking, to optimize the distribution of baggage at Munich Airport. This approach reduced peak loads at carousels and improved passenger waiting times, underscoring the potential of advanced optimization techniques in mitigating bottlenecks.

Peak shaving, a strategy commonly used in logistics and energy systems, focuses on redistributing workloads to manage capacity constraints effectively. In the aviation sector, Clausen and Pisinger [2] explored dynamic routing algorithms to manage short transfer baggage, improving reliability and reducing delays during peak periods. While these approaches are effective in managing routing and scheduling, limited attention has been given to strategies for buffering transfer baggage to achieve peak shaving at specific points in the BHS.

Cold transfer baggage, characterized by longer layover times, offers a unique opportunity for strategic buffering. Studies have shown that decentralized buffers can enhance operational efficiency by reducing transportation times and facilitating smoother reintegration of baggage into the system [7]. However, existing research often addresses buffering as a static process, neglecting the dynamic and variable nature of baggage inflow. Few studies have examined the impact of predictive modeling and adaptive buffering strategies on peak shaving.

Despite the advancements in BHS optimization, a significant gap exists in the literature regarding the use of distribution strategies for dynamically managing and cold buffered transfer baggage. While prior studies have explored general optimization techniques, they often overlook the potential of integrating predictive models with buffering strategies to address the fluctuating occupancy levels at transfer infeed points. Specifically, the role of cold transfer baggage as a targeted solution for peak shaving remains underexplored.

This study addresses the knowledge gap by developing and evaluating distribution strategies for buffering cold transfer baggage. The simulation model is specifically tailored to the operational context of AAS, ensuring its findings are directly applicable to mitigating peak loads at transfer infeed points. In addition, this research provides a foundation for the future integration of Automated Guided Vehicles (AGVs) by optimizing the timing and flow of buffered baggage. These contributions enhance operational efficiency, reduce system bottlenecks, and offer a scalable framework that can be adapted to other major hub airports.

3. Methodology

This research employs a systematic approach to investigate how distribution strategies for buffering cold transfer baggage can alleviate peak occupancy at transfer infeed points. By combining insights from expert interviews, a review of relevant literature, and innovative problem-solving, the methodology is tailored to the operational context of AAS. The research process consists of five interconnected phases: current state analysis, simulation development and validation, forecasting model creation and validation, distribution strategy design, and comparative evaluation.

The study begins with a thorough analysis of the current baggage handling processes at AAS. This analysis focuses on mapping the baggage journey, identifying bottlenecks, and understanding variability in transfer baggage flows. Insights from discussions with industry professionals and a review of optimization techniques from literature provided valuable perspectives, while historical baggage flow data over a 10-week period established baseline performance metrics and variability patterns. This foundational analysis informed the development of the simulation model.

The simulation model was developed to replicate the transfer baggage journey at AAS, covering the process from the arrival of an aircraft to the point where baggage is unloaded onto the quay of the transfer infeed points. This includes key operational steps such as unloading baggage from the aircraft, packing containers, and transporting them from the parking area to either a buffer or the designated infeed point. The transport operations are facilitated by AGVs, which handle two primary tasks: moving containers from the apron to the buffer and transporting them from the buffer to the infeed points. The simulation incorporates distinct baggage streams—ShoCon, cold transfer, and super cold transfer baggage—while accounting for operational constraints such as infeed capacities, buffer storage limits, and varying layover durations. Historical data from AAS was used to define input parameters and validate the simulation by comparing its outputs against key metrics such as occupancy levels at infeed points and throughput times. This validation confirmed the model's accuracy in representing real-world operations, providing a reliable platform for testing distribution strategies.

To support dynamic decision-making within the simulation, a time series forecasting model was developed to predict occupancy levels at transfer infeed points. This forecasting model employs techniques such as Seasonal ARIMA (SARIMA) and TBATS (Trigonometric, Box-Cox transformation, ARMA errors, Trend and Seasonal components), which are well-suited for capturing the seasonal and cyclical patterns inherent in baggage flows. The model uses historical occupancy data from the simulation as input, generating forecasts that inform the timing of baggage release from buffers. Validation of the forecasting model was conducted using holdout datasets from the 10-week data period, with performance metrics such as Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) confirming its predictive accuracy. By integrating this predictive capability into the simulation, the research enables proactive management of baggage flows based on forecasted occupancy levels.

Building on the validated simulation and forecasting models, six distribution strategies for buffered baggage were designed. These strategies range from simple fixed-time release methods to more advanced dynamic approaches guided by forecasted occupancy levels. Each strategy was tested within the simulation to evaluate its impact on peak shaving and operational efficiency. The testing phase included three distinct scenarios to assess the robustness of the strategies: normal operations with typical baggage volumes and full infeed availability, disrupted operations with one infeed point out of service, and high-demand conditions simulating a 20% increase in baggage volumes.

The effectiveness of the distribution strategies was evaluated using key performance metrics, including the percentage reduction in peak occupancy levels at transfer infeed points, the peak occupancy in the buffer, and the required AGV resources. This comparative analysis highlights the trade-offs between the strategies, providing actionable insights into their operational viability and effectiveness under different scenarios.

By combining expert knowledge, literature insights, historical data analysis, and innovative modeling techniques, this research presents a robust methodology to address peak occupancy challenges at AAS. The integration of simulation and forecasting ensures dynamic adaptability to fluctuating baggage volumes, while the evaluation of distribution strategies under various scenarios provides a comprehensive framework for optimizing baggage handling operations.

4. The Baggage Journey

The BHS is a complex network that facilitates the movement of baggage through various stages, ensuring timely delivery to its destination. At AAS, the BHS handles three primary baggage flows: check-in baggage, transfer baggage, and reclaim baggage. Each of these flows interacts with distinct airport processes, as visualized in Figure 1.

Check-in baggage enters the system landside, where it is processed through security checks and routed into storage or directly to make-up stations, where it is prepared for loading onto departing aircraft. Reclaim baggage flows in the opposite direction, as it is unloaded from arriving aircraft and routed landside to the reclaim area for passenger collection. Transfer baggage represents the most dynamic and complex flow, involving the movement of bags from one flight to another during layovers. Within transfer baggage, several streams are distinguished: tail-to-tail, ShoCon, cold transfer, and super cold transfer baggage. These streams differ in their processing requirements and layover durations.

The BHS at AAS incorporates multiple operational subsystems, including storage buffers, infeed stations, sorting systems, and make up areas. While the system is designed to handle average demand levels efficiently, peaks in baggage volumes caused by fluctuating flight schedules create operational bottlenecks. To address these challenges, this research proposes the addition of an external buffer for super cold transfer baggage. The buffer, indicated in Figure 1, is not yet operational but represents the critical component introduced and analyzed in this study to reduce peak occupancy at transfer infeed points. This figure schematically represents the baggage journey through the BHS, highlighting the interconnected nature of baggage handling processes.

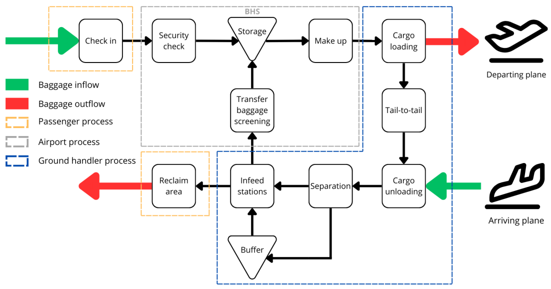


Figure 1: Baggage Handling System of Amsterdam Airport Schiphol

5. Data

The analysis and simulation in this study are based on a comprehensive dataset from AAS, covering 10 weeks of baggage handling operations. The dataset includes Baggage Source Messages (BSM) and scan data, capturing key timestamps, baggage volumes, and flight schedules. For each piece of baggage, detailed information is available about the time of arrival at a specific parking position and the time of departure from the departing parking position. These data provide insights into the flow of transfer baggage, including ShoCon, cold transfer, and super cold transfer baggage.

The dataset was preprocessed to ensure accuracy and reliability. Key preprocessing steps included cleaning invalid or incomplete records, filtering the data to focus exclusively on transfer baggage, and standardizing timestamps for use in the simulation. Based on the processed data, baseline metrics such as average occupancy levels at transfer infeed points, variability patterns, and peak periods were calculated.

This historical dataset not only served as the foundation for developing the simulation model but was also used to validate the forecasting models and evaluate the distribution strategies. By leveraging real-world operational data, this study ensures that its results are directly applicable to the context of AAS, enhancing the reliability and relevance of the proposed strategies.

6. Simulation

The simulation developed for this research serves as a critical tool to evaluate how distribution strategies for buffering cold transfer baggage can alleviate peak occupancy at transfer infeed points. Designed for the operational context of AAS, the simulation captures the complexity of the transfer baggage journey while incorporating key system constraints and dynamics. This section outlines the main elements of the simulation, as well as its validation and verification process.

The primary objective of the simulation is to model the journey of transfer baggage from aircraft arrival to the infeed points of the BHS. By incorporating historical data and system constraints, the simulation provides a platform to test distribution strategies that optimize the timing and allocation of buffered baggage to achieve peak shaving.

6.1. Transfer Baggage Journey

The simulation model distinguishing between two processes: the "cold process" for buffered baggage and the "hot process" for baggage that bypasses buffering. Figure 2 illustrates the sequential stages of this journey, including unloading, buffering, and transportation to infeed points. The timeline is divided into key milestones:

- t1: Time from aircraft arrival (AIBT) to the unloading of the first baggage item.
- t2: Time to unload and pack the last baggage item into a container.
- t3: Transport time for containers to reach either the buffer or infeed points.
- t4: Time containers spend in the buffer before being released to infeed points.
- t5: Transportation from the buffer or apron to the infeed point and subsequent processing.

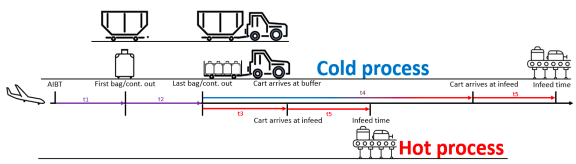


Figure 2: Transfer baggage journey from arrival to transfer infeed point [7]

The simulation distinguishes between flights based on their operational characteristics and the types of transfer baggage they carry. Baggage is categorized into four temperature classes—Tail-to-tail, ShoCon, cold transfer, and super cold transfer baggage—based on layover times. These classifications determine whether baggage can be directly processed (hot process) or buffered (cold process) before being reintroduced into the system.

The simulation differentiates between three baggage types: bulk carts, which can carry up to 30 bags; AKE containers, with a capacity of 38 bags per container; and AKH containers, designed to hold 28 bags per container. These baggage types play a significant role in shaping the packing, transportation, and buffering processes modeled in the simulation.

AAS is divided into multiple ramp clusters, which serve as operational zones for loading and unloading baggage. The simulation accounts for the spatial distribution of these clusters and their role in determining transportation times to buffers and infeed points.

The capacity constraints of infeed points and unloading quays are critical factors in the simulation. These capacities are modeled to reflect the real-world limits of the BHS at AAS, ensuring that peak periods accurately represent operational bottlenecks.

The buffer is a central feature of the simulation, designed to temporarily store super cold transfer baggage. AGVs are responsible for transporting containers between the apron, buffer, and infeed points. The simulation models AGV scheduling, ensuring efficient use of these resources under varying operational scenarios.

6.2. Verification and Validation

The simulation's verification and validation processes were essential to ensure its technical accuracy and ability to reflect real-world operations at AAS. The verification focused on ensuring the logical consistency of the model under various scenarios, while validation involved comparing the simulation output with actual historical data.

To verify the model, a series of tests were conducted to evaluate how the simulation responded to changes in input parameters. For example, key variables such as unloading times and travel times were systematically adjusted, and the resulting shifts in baggage infeed times were observed. These tests confirmed that the model responded predictably and logically to such changes, with delayed inputs appropriately shifting the simulated occupancy peaks to later intervals. The verification demonstrated that the simulation's underlying logic was robust and aligned with operational expectations.

Validation compared the simulated occupancy rates at transfer infeed points with historical data collected from AAS. This was done using both visual and statistical methods. Visual validation involved line plots that compared simulated and actual occupancy levels, both as averaged daily values and for a specific high-volume day, June 30, 2024. While the simulation successfully captured general trends and patterns, slight deviations were observed during peak periods. These discrepancies highlighted a consistent time misalignment, with simulated peaks occurring earlier than the actual peaks.

To address this issue, a delay of 1500 seconds was introduced into the simulation. This adjustment significantly improved the alignment between simulated and actual data, resulting in a lower RMSE and MAE. While this delay cannot be entirely explained empirically, discussions with simulation experts at AAS revealed that such unaccounted delays are common and often included in operational analyses to better reflect observed dynamics. This validation step underscored the necessity of including the delay to ensure the simulation closely mirrors real-world processes.

Statistical validation further supported the accuracy of the simulation. Metrics such as RMSE, MAE, and Mean Absolute Percentage Error (MAPE) quantified the discrepancies between simulated and actual data. The configuration with the 1500 seconds delay consistently produced the most accurate results across all metrics, confirming its suitability for use in subsequent simulations (Figure 3). This adjustment ensured that the model accurately replicated baggage flow patterns and system dynamics, particularly during peak periods.

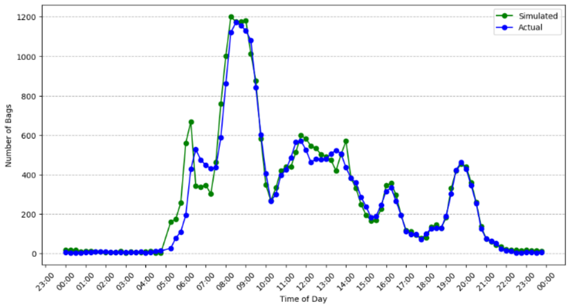


Figure 3: Simulated and Actual Average Baggage Occupancy per 15-Minute Interval

The combined verification and validation efforts confirmed that the simulation is a reliable platform for evaluating distribution strategies and testing various scenarios. By addressing operational nuances and aligning with historical data, the model is well-equipped to support decision-making processes aimed at optimizing baggage handling operations at hub airports.

7. Time Series Forecasting

To optimize the distribution of super cold transfer baggage and reduce peak occupancy at infeed points, this research incorporates time series forecasting models to predict occupancy levels. Forecasting provides a strategic basis for determining the optimal moments to release buffered baggage into the system, ensuring that the BHS operates efficiently and avoids unnecessary peaks. By leveraging historical data, the models enable proactive adjustments to baggage flow, supporting the goal of balanced system loads.

Two forecasting models, SARIMA and TBATS, were evaluated. These models were selected for their suitability in capturing the seasonal patterns inherent in baggage flows, such as daily and weekly fluctuations, which are critical to the operation of the BHS at AAS.

SARIMA was chosen for its effectiveness in capturing single seasonal patterns, particularly the daily cycle in baggage volumes. The model's components—autoregressive, differencing, and moving average—are extended with seasonal terms to handle repeated patterns over time. For this study, the seasonal period was set to 96 quarters (24 hours). A grid search determined the optimal parameters, and the model was validated using key metrics such as RMSE, MAE, and SMAPE (Symmetric Mean Absolute Percentage Error). SARIMA performed well, achieving an RMSE of 144.37, indicating that it effectively captured daily trends in baggage flows.

TBATS, a more flexible model, was employed to capture multiple seasonal patterns, such as the interplay between daily (96 quarters) and weekly (672 quarters) cycles. This capability makes it particularly suitable for forecasting in complex systems like the BHS, where both short-term and long-term seasonal trends affect operations. While TBATS had a slightly higher RMSE (151.02) and MAE (115.71) compared to SARIMA, its ability to model weekly seasonality provided a more comprehensive view of occupancy patterns.

The figure below demonstrates the performance of the TBATS model in forecasting baggage occupancy within the BHS. The blue line represents the training data, which only shows one dat of the 10 weeks of historical baggage flows, excluding the final day. This data was used to train the model, capturing both daily and weekly patterns in occupancy levels. The orange line depicts the actual test data from June 30, 2024, a day with high baggage volumes. The red line shows the TBATS model's predictions for this day, leveraging the learned patterns from the training data. The comparison highlights the model's ability to align closely with observed trends, particularly capturing the daily and weekly fluctuations in baggage flows. Although minor deviations occur, the predictions generally reflect the actual occupancy levels, demonstrating the model's effectiveness in providing actionable forecasts for optimizing baggage distribution strategies.

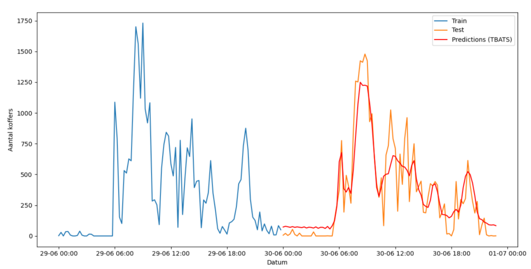


Figure 4: Train, Test, and Predictions using TBATS Model for 30-06-2024

Both models were validated using historical data and tested against the actual occupancy levels for June 30, 2024, a day with high baggage volumes and the last day of the available 10-week dataset. While SARIMA demonstrated slightly higher accuracy for daily forecasts, TBATS proved more robust for capturing weekly variations. Given the importance of weekly cycles in baggage handling operations, TBATS was chosen as the primary model for developing and evaluating distribution strategies.

The forecasting models form an integral part of the simulation framework, enabling dynamic adjustments to the timing of baggage releases from the buffer. By accurately predicting system occupancy, the research ensures that the distribution strategies are informed by data-driven insights, contributing to the effective management of baggage flows and reduction of peak loads at infeed points.

8. Distribution Strategies and Scenarios

This section outlines the distribution strategies developed for buffering super cold transfer baggage at AAS and evaluates their performance under different operational scenarios. The primary objective is to minimize peak occupancy at infeed points, ensuring smoother operations within the BHS. Two critical aspects are addressed: the timing of when baggage should be buffered and when it should be reintroduced into the system. Together, these elements enable effective management of baggage flows and workload distribution.

8.1. Strategic Timing for Buffering Baggage

Determining the optimal timing for buffering baggage is a crucial component of peak shaving. Three strategies were considered: fixed target, polynomial function, and a hybrid of both.

8.1.1 Fixed Target Value

The fixed target strategy employs a predefined threshold (600 bags per 15-minute interval) as the occupancy limit. When this threshold is exceeded, super cold baggage is buffered to alleviate strain on the infeed points. This approach is simple and reliable but does not account for natural fluctuations in baggage inflow caused by flight schedules, potentially resulting in unnecessary buffering during expected peaks.

8.1.2 Polynomial Target Value

A more adaptive approach is the polynomial function strategy, which uses a degree-9 polynomial to smooth predicted occupancy patterns over time. This dynamic target adapts to fluctuations in baggage volumes, allowing for more realistic and flexible buffering decisions. A minimum occupancy boundary of 150 is set to avoid excessive buffering during low-demand periods. By capturing daily trends and smoothing peak and trough variations, the polynomial function achieves a balance between operational flexibility and efficiency.

8.1.3 Hybrid Target Value

The hybrid strategy combines the fixed target and polynomial approaches, leveraging the stability of a fixed baseline with the adaptability of a dynamic curve. This method ensures consistent buffering during low-variability periods while accommodating higher volumes when necessary, creating a robust solution for managing occupancy at the infeed points.

Figure 5 illustrates the hybrid target value function, combining the fixed target and polynomial function to provide a dynamic yet stable occupancy threshold for buffering decisions.

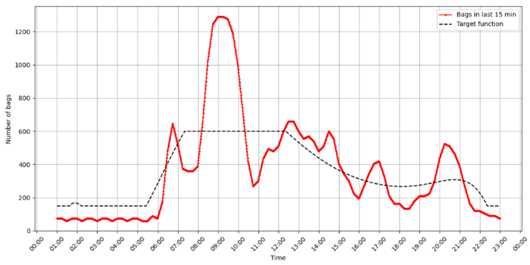


Figure 5: Target Value on Time Series Forecasting TBATS for 30-06-2024 with Hybrid Function

In summary, three options are proposed for determining when baggage should be buffered: a fixed target, a polynomial function, and a combination of both. These options provide varying levels of adaptability and precision, ensuring flexibility in addressing the fluctuating demands of the BHS.

8.2. Strategic Release of Buffered Baggage

The timing of reintroducing buffered baggage is critical to maintaining system stability and ensuring that no baggage misses its connecting flight. Two strategies were developed for this purpose: early release and optimized release.

8.2.1 Early Release

The early release strategy utilizes time series forecasts to determine the optimal moment for reintroducing buffered baggage. At each simulation step (indicated by the green line in Figure 6), the forecasted occupancy levels are consulted to identify the first time slot when the predicted occupancy falls below the target value. This ensures that baggage is reintroduced when the system can handle additional load without exceeding the occupancy threshold. If no such moment is found within the permissible window, the baggage is scheduled to return during the time with the lowest predicted occupancy. The blue line in Figure 6 represents the latest possible time by which the baggage must return to the system to ensure that no bags miss their connecting flights. This approach balances system load while adhering to operational constraints.

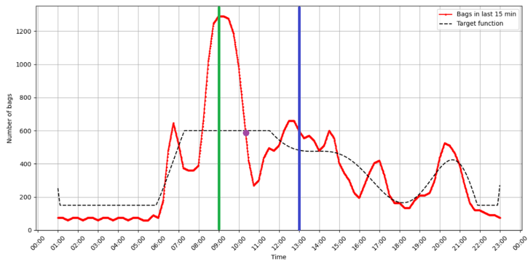


Figure 6: Early Reintroduction Strategy

The simulation continuously updates these decisions at each step, ensuring dynamic adaptability to changing system conditions.

8.2.2 Optimized Release

The optimized release strategy takes a broader view of the available reintroduction windows, prioritizing the reintroduction of buffered baggage at the moment when the predicted occupancy has the largest gap below the target value. This approach ensures that baggage is reintroduced during periods of minimal system load, further reducing the risk of creating new peaks at the infeed points.

At each simulation step, such as the moment indicated by the green line in Figure 7, the forecasting model is used to identify the optimal reintroduction time. Rather than selecting the first available moment when the occupancy falls below the target value, the model scans the allowable timeframe, up to the latest possible return time (indicated by the blue line), to find the moment with the maximum gap below the target. For example, if the forecast shows a significant drop at 10:45, even though a smaller dip occurs earlier at 10:15, the baggage will be scheduled for reintroduction at 10:45. This ensures that the system operates efficiently while avoiding unnecessary strain during higher occupancy periods.

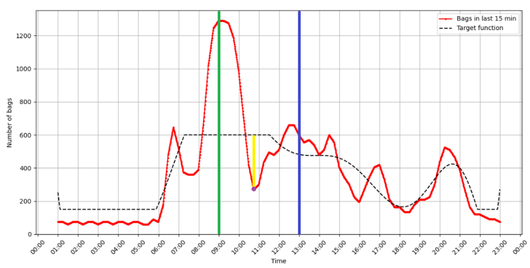


Figure 7: Optimized Reintroduction Strategy

This method leverages the flexibility of the forecasted occupancy data to maximize system capacity during low-demand periods. By dynamically scheduling reintroductions at the most opportune times, the optimized release strategy enhances the system's overall efficiency and minimizes operational bottlenecks.

8.3. Scenarios

To evaluate the effectiveness and robustness of the buffering and reintroduction strategies, three scenarios were developed, each reflecting different operational conditions and levels of system stress. The first scenario represents a baseline situation where all infeed points and baggage halls are fully operational. This scenario provides a controlled environment to assess how well the strategies maintain balanced occupancy levels under standard conditions, where baggage inflow follows historical patterns.

In the second scenario, one of the four infeed points, baggage hall Z, becomes unavailable due to maintenance or unforeseen breakdowns. This results in the redistribution of 8.8% of the baggage normally processed at this hall to the remaining infeed points, increasing their workload. This scenario simulates real-world challenges, such as reduced system capacity, and tests whether the strategies can efficiently manage resources and avoid bottlenecks despite this additional strain.

The third scenario introduces a 20% increase in baggage inflow, reflecting conditions such as denser flight schedules, delays, or seasonal surges. This heightened demand places significant pressure on the infeed points and the BHS as a whole. The increased volume tests the resilience of the strategies in maintaining balanced occupancy levels and ensuring smooth operations under heightened stress.

Together, these scenarios allow for a comprehensive evaluation of the buffering and reintroduction strategies under varying operational challenges, providing critical insights into their adaptability and performance in managing peak loads.

8.4. Configurations and Evaluation

Each of the three buffering strategies was paired with the two reintroduction strategies, resulting in six unique configurations. These configurations were tested across the three scenarios, generating 18 distinct combinations. Baseline scenarios without buffering were also included for comparison.

9. Results

This section evaluates the impact of various buffering strategies for super cold transfer baggage on peak shaving at transfer infeed points, based on simulations conducted for a single operational day (30-06-2024). Table 1 summarizes the results for 18 configurations, each combining different buffering options and release strategies under three scenarios: normal operations, reduced capacity, and increased baggage inflow.

9.1. Key Performance Indicators

To assess the configurations, several Key Performance Indicators (KPIs) were employed:

- Peak Occupancy: Maximum number of bags at infeed points in a 15-minute interval (lower values indicate effective peak shaving).
- Standard Deviation (STD): Variability in occupancy across the day, where lower values suggest more consistent flows.
- Mean Absolute Change (MAC): Average change in occupancy between consecutive minutes, reflecting flow stability.
- Coefficient of Variation (CoV): Normalized variability (STD/mean) for relative comparison.
- Peak Buffer Occupancy: Maximum number of containers in the buffer at any point, indicating reliance on buffering.
- Buffered Containers: Total containers buffered throughout the day.
- AGVs Required: Number of AGVs needed for container transport.

Configurations using a buffer demonstrated significant reductions in peak occupancy compared to baseline scenarios without buffering. The best-performing configurations achieved up to 26.8% peak shaving under normal conditions, with Configurations 10 and 16 (Polynomial and Combination buffering options paired with Optimized Release) reducing peak occupancy from 1958 to 1433 bags. In comparison, the Fixed Target buffering strategy achieved slightly less peak shaving (1454 bags) but required fewer AGVs.

The Polynomial and Combination options consistently required more AGVs (31–39) compared to the Fixed option (26–28). These strategies also exhibited slightly higher buffer peaks (118 containers) compared to the Fixed option (113 containers). Despite this, the Polynomial and Combination options provided smoother occupancy profiles, reflected in their lower STD, MAC, and CoV values.

9.2. Occupancy Levels and Buffer Utilization for Configuration 10

To illustrate the effectiveness of the proposed distribution strategies, the occupancy levels at the transfer infeed points and the buffer utilization are analyzed for Configuration 10, which combines the Polynomial Target strategy with the Optimized Release strategy under normal operating conditions.

Figure 8 compares the occupancy levels at the transfer infeed points for Configuration 10 against the baseline (Base 1), where no buffering is applied. The graph demonstrates a significant reduction in peak occupancy levels, highlighting the impact of buffering super cold transfer baggage. Configuration 10 achieves a 26.8% reduction in peak occupancy compared to the baseline, effectively smoothing the baggage flow over time.

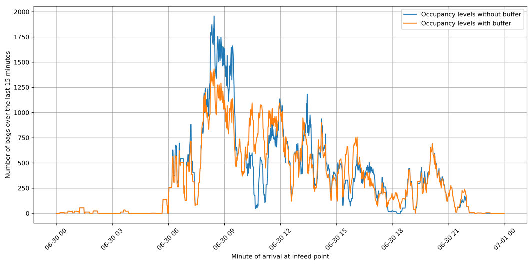


Figure 8: Transfer infeed point occupancy levels for Configuration 10 and Base 1

Figure 9 presents the buffer occupancy levels for Configuration 10, showcasing the dynamic use of the buffer throughout the day. The maximum buffer occupancy peaks at 118 containers.

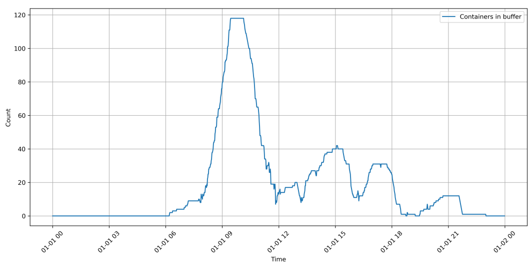


Figure 9: Buffer occupancy levels for Configuration 10

10. Discussion

This section examines the key findings of the research, discussing the implications of the simulation model, forecasting approach, and distribution strategies for buffering super cold transfer baggage at AAS. Limitations, trade-offs, and opportunities for further research are also addressed, with a focus on optimizing peak shaving at transfer infeed points.

10.1. Simulation Model and Assumptions

The simulation model captures the complexity of transfer baggage flows while introducing simplifications to maintain computational feasibility. Key assumptions, such as excluding odd-sized baggage and clustering ramp positions, streamlined the analysis but may slightly underestimate operational variability. Notably, a 1500-second delay was incorporated to align the simulation outputs with observed data, highlighting the importance of accounting for unmodeled delays in real-world processes. While these assumptions allow for a focused evaluation of buffering strategies, further refinement of the model could enhance its realism, particularly by incorporating operational disruptions and variability in transport times.

The decision to buffer only super cold transfer baggage reflects a conservative approach, prioritizing reliability over maximum peak shaving. This aligns with operational feedback from AAS and KLM but leaves potential gains from buffering cold transfer baggage unexplored. Future research could evaluate the controlled inclusion of this baggage category, particularly under scenarios with additional safety buffers in transport times.

Conf.	**Option**	**Alternative**	**Scenario**	**Peak**	**STD**	**MAC**	**CoV**	**Peak buff**	**Buff. cons**	**AGVs**
10	2: Polynomial	2: Optimized release	1: Normal	1433	350.48	26.55	1.04	118	251	31
16	3: Combination	2: Optimized release	1: Normal	1433	350.48	26.55	1.04	118	251	31
2	1: Fixed target	1: Early release	2: Reduced cap.	1454	348.81	26.99	1.04	114	205	26
4	1: Fixed target	2: Optimized release	1: Normal	1454	347.95	26.94	1.03	116	199	28
11	2: Polynomial	2: Optimized release	2: Reduced cap.	1459	349.25	26.77	1.04	115	251	31
17	3: Combination	2: Optimized release	2: Reduced cap.	1459	349.25	26.77	1.04	115	251	31
1	1: Fixed target	1: Early release	1: Normal	1462	349.90	26.25	1.04	113	199	27
5	1: Fixed target	2: Optimized release	2: Reduced cap.	1464	350.19	27.23	1.04	117	205	28
8	2: Polynomial	1: Early release	2: Reduced cap.	1484	345.25	26.57	1.03	105	259	32
14	3: Combination	1: Early release	2: Reduced cap.	1484	345.25	26.57	1.03	105	259	32
7	2: Polynomial	1: Early release	1: Normal	1492	346.89	25.82	1.03	104	257	33
13	3: Combination	1: Early release	1: Normal	1492	346.89	25.82	1.03	104	257	33
12	2: Polynomial	2: Optimized release	3: Incr. inflow	1663	417.77	30.98	1.03	139	300	39
18	3: Combination	2: Optimized release	3: Incr. inflow	1663	417.77	30.98	1.03	139	300	39
3	1: Fixed target	1: Early release	3: Incr. inflow	1734	411.03	29.64	1.02	138	288	34
6	1: Fixed target	2: Optimized release	3: Incr. inflow	1736	414.84	30.49	1.03	142	288	35
9	2: Polynomial	1: Early release	3: Incr. inflow	1819	413.07	30.15	1.02	127	311	40
15	3: Combination	1: Early release	3: Incr. inflow	1819	413.07	30.15	1.02	127	311	40
Base 1	0. No buffer	0. No buffer	1: Normal	1958	404.83	28.82	1.20	0	0	0
Base 2	0. No buffer	0. No buffer	2: Reduced cap.	1958	404.15	29.4	1.20	0	0	0
Base 3	0. No buffer	0. No buffer	3: Incr. inflow	2283	485.38	33.88	1.20	0	0	0

Table 1: Full results of all configurations in peak shaving order

10.2. Forecasting Model Performance

The SARIMA and TBATS models effectively predicted occupancy levels, enabling data-driven buffering decisions. TBATS' ability to capture weekly seasonality proved advantageous for managing baggage flows in complex systems. However, the forecasting accuracy could be further improved by leveraging larger datasets and exploring advanced models to better capture non-linear and abrupt changes in baggage flows. Ensuring data quality and consistency remains critical, as gaps or inconsistencies in historical data could introduce biases into predictions.

10.3. Effectiveness of Distribution Strategies

The distribution strategies demonstrated significant potential for peak shaving, with the Polynomial and Combination strategies paired with the Optimized Release approach achieving reductions of up to 26.8% in peak occupancy. These strategies, while more resource-intensive, provided smoother occupancy profiles, as reflected in lower standard deviation and variability metrics. The Fixed Target strategy, though less effective in peak shaving, offered a resource-efficient alternative, particularly in scenarios with limited AGV availability.

AGV requirements emerged as a critical factor, with Polynomial and Combination strategies demanding significantly more AGVs than the Fixed Target approach. This underscores the importance of balancing operational efficiency with resource constraints, particularly in space-constrained environments like airside operations at AAS.

10.4. Implications and Practical Applications

The study highlights the potential for buffering strategies to mitigate peak occupancy and improve operational stability in BHSs. By integrating predictive models with dynamic buffering decisions, the simulation offers a foundation for real-time decision support systems. Such systems could dynamically adjust buffering strategies based on live occupancy predictions, ensuring efficient resource utilization and minimal bottlenecks.

Practical implementation challenges, such as AGV traffic management and infrastructure investments for buffer expansion, must be addressed. These findings provide valuable input for cost-benefit analyses, helping airports evaluate trade-offs between capital expenditure, operational costs, and efficiency gains.

10.5. Limitations and Future Research

While the simulation and forecasting models effectively captured key dynamics, several limitations remain. The exclusion of operational disruptions, equipment constraints, and personnel shifts limits the generalizability of the findings. Incorporating these variables in future studies could enhance the robustness of the model and test the adaptability of buffering strategies under broader scenarios.

Additionally, optimizing the Fixed Target threshold and refining the Polynomial Target function could lead to even more effective peak shaving. Future research could explore machine learning-based approaches to develop adaptive target functions that better align with real-world inflow patterns.

11. Conclusion

This study explored the use of simulation and forecasting models to identify the most effective distribution strategies for buffering super cold transfer baggage at AAS. By analyzing the performance of various strategies under different operational scenarios, this research provides insights into how peak shaving can enhance the efficiency and resilience of BHS at hub airports.

The findings demonstrate that peak shaving at transfer infeed points, achieved through the strategic buffering and reintroduction of super cold transfer baggage, significantly reduces operational bottlenecks. A combination of simulation and time series forecasting enabled the evaluation of six unique strategies, each addressing specific trade-offs between peak occupancy reduction, AGV requirements, and buffer capacity management.

The most effective strategy, utilizing a polynomial target function combined with an optimized release mechanism, achieved a 26.8% reduction in peak occupancy. This strategy dynamically adjusted buffering decisions based on forecasted occupancy, ensuring efficient redistribution of baggage flows. However, the high resource demand, particularly in terms of AGVs, underscores the trade-off between peak shaving efficiency and operational resource requirements. In contrast, the fixed target strategy achieved slightly less peak shaving (25.7%) but required fewer AGVs, offering a more resource-efficient alternative for airports with constrained capacity.

Additionally, the early release strategy consistently maintained the lowest buffer occupancy levels, making it particularly advantageous for airports with limited buffer space. While this approach achieved slightly lower peak shaving at infeed points, it reduced congestion within the buffer and ensured smoother operations during periods of high baggage inflow.

The results highlight that the choice of a distribution strategy depends on the specific operational priorities of AAS. Strategies that prioritize peak shaving may require higher investments in AGVs and operational resources, while resource-efficient alternatives can still deliver substantial improvements. By integrating these strategies into its

BHS operations, AAS can better manage fluctuating baggage volumes and improve overall system performance.

This research provides a robust framework for supporting AAS in selecting and implementing the most suitable distribution strategies for peak shaving, enabling smoother baggage flows and greater resilience in its operations.

12. Recommendations

This study highlights several opportunities to further optimize peak shaving strategies at AAS. Future research should explore expanding the criteria for buffered baggage to include cold transfer baggage with layover times of two to three hours, potentially enhancing peak shaving effectiveness. Additionally, advanced forecasting models, such as machine learning or deep learning, could improve occupancy predictions, enabling more precise and adaptive buffering strategies.

Introducing greater flexibility in infeed point assignments, such as secondary or tertiary preferences, could help balance loads across the system, mitigating localized bottlenecks. Furthermore, analyzing the downstream impacts of buffering on other BHS subsystems, such as security screening and sorting, could lead to more integrated system-wide peak shaving strategies.

A comprehensive cost-benefit analysis is recommended to evaluate the financial implications of various strategies, considering investments in infrastructure and AGV deployment against potential savings in staffing and efficiency. Developing real-time decision support systems that integrate forecasting with dynamic resource allocation could further enhance operational adaptability.

Finally, testing strategies under additional scenarios, such as weather disruptions or equipment failures, would provide insights into their robustness and practical feasibility. These recommendations offer a pathway to refine peak shaving approaches, ensuring their scalability and alignment with AAS's operational and strategic goals.

References

[1] Torben C Barth, Janus Timler Holm, Jakob Lindorff Larsen, and David Pisinger. Optimization of transfer baggage handling in a major transit airport. In *Operations Research Forum*, volume 2, pages 1–35. Springer, 2021.

[2] Tommy Clausen and David Pisinger. Dynamic routing of short transfer baggage. *DTU Management Engineering, Tech. Rep. 10*, 2010.

[3] Markus Frey. *Models and methods for optimizing baggage handling at airports.* PhD thesis, Technische Universität München, 2014.

[4] Markus Frey, Ferdinand Kiermaier, and Rainer Kolisch. Optimizing inbound baggage handling at airports. *Transportation Science*, 51(4):1210–1225, 2017.

[5] Ma Hanwu and Li Juan. Process optimization research on baggage transit system of hub airports. In *2009 International Conference on Information Management, Innovation Management and Industrial Engineering*, volume 3, pages 58–61. IEEE, 2009.

[6] Edward Huang, Irene Liu, and James T Lin. Robust model for the assignment of outgoing flights on airport baggage unloading areas. *Transportation Research Part E: Logistics and Transportation Review*, 115:110–125, 2018.

[7] Niek van der Grift. *Reducing peak stress on the Baggage Handling System.* PhD thesis, Technical University of Delft, Delft, February 2023.