Utilization improvement of a sortation system for the parcel industry

van den Hoven, M.

Award date:
2019

EINDHOVEN UNIVERSITY OF TECHNOLOGY

Department of Industrial Engineering & Innovation Sciences
Operations Planning Accounting & Control

Utilization Improvement of a Sortation System for the Parcel Industry

Master's Thesis

Matthijs van den Hoven

First Supervisor	Prof.dr.ir. I.J.B.F. Adan
Second Supervisor	Dr. Oktay Karabağ
External supervisor	Ing. Jean-Pierre Rijkers

Eindhoven, Wednesday 25th September, 2019

Abstract

This thesis is the result of a master's thesis project at Vanderlande Industries in Veghel concerning a type of parcel sortation system that is typically located in distribution centers of parcel delivery companies. The sortation process starts by loading parcels through infeed zones onto a main loop that travels at a constant velocity. From the main loop, parcels are sorted into outfeed zones. This study aims to determine whether the utilization of the main loop can be increased while using at most three infeed zones.

The stochasticity of parcel dimensions and interarrival times are an important cause of utilization loss. The use of buffers was proposed such that a sequence of parcel arrivals on the main loop can be planned and released from buffers. The trajectory of parcels that are released from such a buffer was investigated before considering potential buffer designs. This trajectory consists of the acceleration zone, infeed zone, and main loop. When comparing the cycle times of each zone, it turned out that the current acceleration zone is not able to process parcels without causing slippage when it transports parcels at the same rate as the infeed zone. The cycle times of the main loop depend on whether a planned sequence contains more than one parcel from a particular infeed. Buffers must release parcels such that the planned sequence of arrivals can be processed by all crossorter zones.

Two types of buffers were considered. The first design requires parcels to decelerate, wait if necessary, be accelerated again at the right moment, and handed over to the next conveyor belt at a constant velocity. It turned out that completely decelerating and accelerating takes too much time and causes the cycle times of the buffers to be longer than that of the infeed zone for many parcel lengths. The second design is a buffer belt that can lower it's velocity to one of three configured lower velocities rather than stopping completely. A requirement of this type of buffer is that the windows between parcels before arriving at the buffer must be smaller than, but not deviate excessively from the windows needed after the buffer.

The initial results of an emulation study showed that the average utilization was 71.92%, while a system with buffers had an average utilization of 66.39%. The reason for this was that the infeed cycle times used to determine the theoretical utilization were taken from situations where the cycle time could be predictable, while the real cycle time may be lower on average, but less predictable. After further investigation, the unpredictability in real systems turned out to be caused by a gradual shift in the relative moment of acceleration on the infeed. For some parcel lengths, graphs showing this gradual shift indicate that a controller could be built that adjusts parcel windows based on the moment of acceleration. The utilization assuming such a controller is feasible was 73.71%. An additional improvement in theoretical utilization could be made using an algorithm for planning sequences that selects the parcel that can be inducted the earliest instead of using a Round-Robin algorithm. In this case, the utilization was 74.42%.

Future research directions should aim to find out whether these findings hold in situations including operational disturbances. Furthermore, the amount of slippage on the acceleration zone should be measured on real crossorters to confirm whether or not a redesign is necessary. Another research direction could be to determine whether it is possible to create a controller that alters the windows based on the relative acceleration moment.

Contents

Chapter 1

Introduction

Vanderlande is a market leader in logistic process automation at airports and in the parcel business. They have more than 5,500 employees worldwide and over 100 site-based teams stationed close to the customer. Around the world, 600 airports including 13 of the world's top 20 have baggage handling systems made by Vanderlande. These systems move 3.7 billion pieces of luggage around the world per year, which is about 10.1 million a day. Vanderlande's parcel systems sort nearly 39 million parcels each day for some of the world's leading parcel companies.

1.1 The Crossorter System

This study will focus on the crossorter system, which is used for parcel sorting. This graduation project aims to find alterations that increase the utilization of the main loop of the crossorter system while using three infeed zones. In this chapter, the system scope is given and the added value in comparison with previously conducted studies is reviewed. The second chapter contains an analysis of the current system concerning possible bottlenecks. In the third chapter, two alternative scenarios are proposed and discussed of which one is selected to determine the theoretical utilization. Chapter 4 compares the utilization of the alternative scenario to that of the current system. A crossorter consists of acceleration zones, infeed zones, outfeed zones, and a main loop that connects the infeed zones to the outfeed zones. Figure 1.1 shows an abstract top view of a crossorter with three of each of the aforementioned zones.

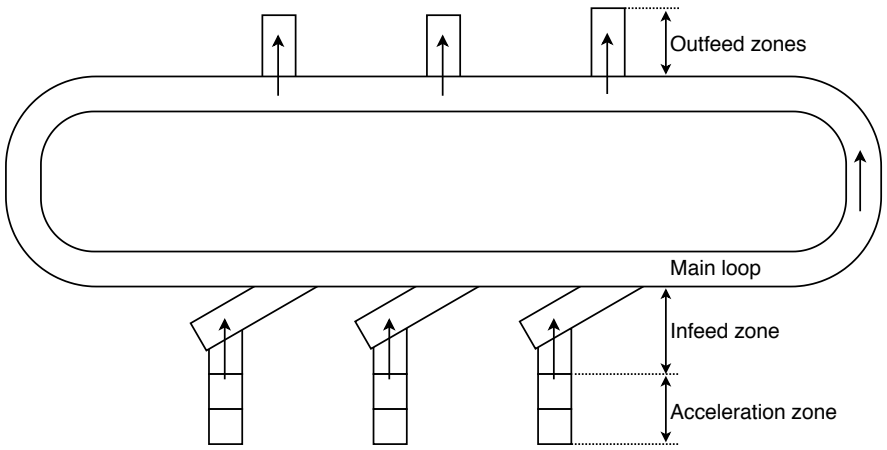

Figure 1.1: Schematic depiction of a crossorter system

The main loop consists of single or dual belt carriers on which respectively one or two parcels may be loaded. Figure 1.2 and Figure 1.3 show pictures of single belt and dual belt carriers respectively. The width of the carrier belts is 600 mm, 700 mm, or 800 mm, while the length of the belts can be either 700, 1200 or 1500mm, depending on what the crossorter type. If a parcel is too large for the width of the carrier, two or three carriers will be utilized for this parcel for dual and single belt carrier systems respectively.

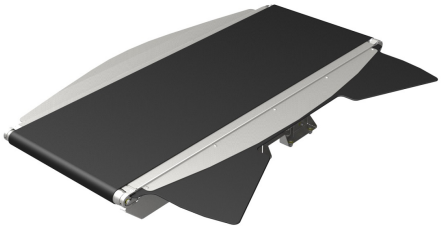

Figure 1.2: Single belt carrier

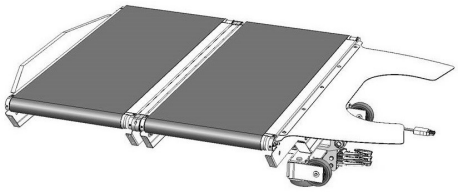

Figure 1.3: Dual belt carrier

Before parcels arrive at the acceleration zones, they typically come from multiple supply streams. At the start of these streams, there may be operators that unload parcels from trucks onto conveyors. The parcels are then conveyed towards the crossorter. Once the parcels are on the loop, a bar-code scanner determines to which outfeed a parcel must be transported. After the parcels are sorted they can be loaded onto trucks again and transported to their final destinations. The majority of crossorters work in such a way, so other loading methods such as manual placement of parcels in the infeed zone will not be included in the scope of this study.

The merging of parcels from several infeed zones onto the main loop is of crucial importance to the utilization, so a brief explanation of how this works is given. Once a parcel arrives at an infeed zone, it will be transported towards the main loop at a constant velocity until it is accelerated to a velocity which allows for a hand-over to the loop without slippage. Before the parcel can be accelerated, it passes a light grid that measures the length and width. Once the dimensions of the parcel are known, the acceleration trajectory of the parcel onto the main loop is calculated. The trajectory is limited by the time windows in which a parcel can be accelerated without colliding with another parcel on the loop. Once the trajectory is determined, the parcel is transported on a set of smaller conveyor belts called short belts that allow for rapid acceleration.

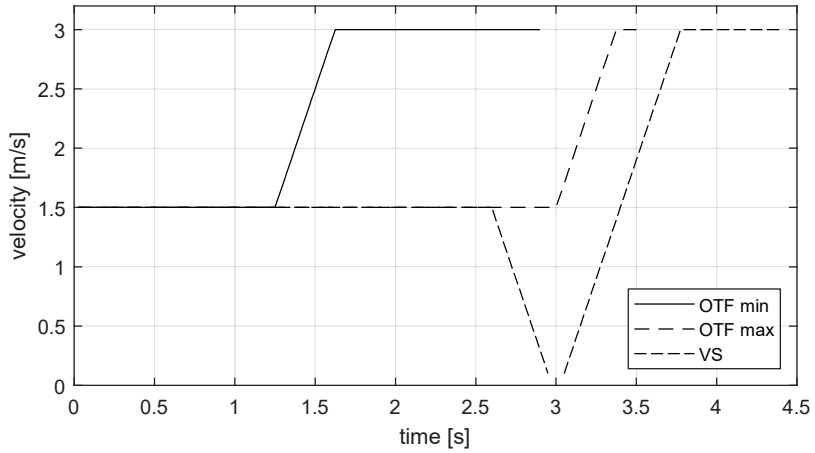

Figure 1.4: Comparison between velocity profile windows

Initially, the control system considers the time window called On-The-Fly (OTF). The window is defined as the time between OTF min and OTF max as shown in Figure 1.4. If a parcel is accelerated within this window the parcel will maintain its velocity for a certain amount of time before accelerating onto the loop. If no carriers were free in the OTF window, the infeed zone conveyor stops completely at the latest possible point and finds a carrier in what is called the Via-Stop (VS) time window. This window would practically be unlimited if the crossorter is running for an unlimited amount of time. Therefore this window ranges from the first possible location that can be reached after the parcel has stopped completely until the last carrier that will pass the infeed zone before the crossorter is shut off.

1.2 Previously Conducted Research

Several students had already researched the crossorter systems before the start of this graduation project. Peeters [2015] investigated the effects of alternative merge algorithms to minimize the imbalance between infeed zone. He used a crossorter model within the Automod software package for this. Further research was conducted by Meens [2017], who built and validated a simulation that was used for comparing different merge algorithms in Matlab Simulink instead of Automod. The study conducted by Struik [2018] focuses on improving the design capacity of a single infeed zone. This led to several proposed hardware changes that increase the throughput of an infeed zone. Summaries of the aforementioned studies are given in the following subsections followed by an overview of the differences between the studies. This provides the necessary background information for the problem description.

1.2.1 Balancing Control of Material Handling Systems

Peeters [2015] described the problem that comes with a FCFS merging algorithm: On average, parcels on infeed zone located further downstream will take longer to find an available carrier. This causes an imbalance between the throughput levels of the infeed zone. The goal was to design an algorithm that can balance the flow of parcels through the infeed zone. The system that was regarded during this study was a crossorter with pre-merges. For each infeeed zoness there were 4 pre-merge lanes feeding into the lane before the infeed zone. Figure 1.5 shows such a system with 3 infeed zone lanes and 12 pre-merge lanes. For the current project, crossorters without pre-merge lanes are also regarded, so the goal is to find improvements regardless of whether there are pre-merge lanes or not. Peeters [2015] created algorithms that work in case there are pre-merge lanes, so some results might not be relevant to the current project.

Peeters [2015] proposed two new types of algorithms: 1-merge and 2-merge. Both algorithms use waiting time to measure the imbalance and make decisions, but the 1-merge algorithm only controls the flow at the pre-merge level while the 2-merge algorithm controls the flow at the infeed zone level as well. The 1-merge algorithm is initialized by the arrival of a parcel at the pre-merge lanes. The 2-merge algorithm is a combination of two algorithms: one that initializes when a parcel arrives at the pre-merge, and one that initializes when a parcel arrives at the infeed zone. Regardless of where it initializes, the procedure behind the algorithm is the same.

The algorithms check whether the new parcel will collide with another parcel when it merges onto the next part of the system (the infeed zone lane, or the loop). If a collision is expected to occur, one of the parcels is delayed so that the collision is prevented. Which of the two parcels is delayed depends on the imbalance between the two pre-merge lanes. The imbalance is defined as the difference between the occurred waiting time of parcels that have passed a pre-merge lane. This includes the waiting time of the incoming parcels. If prioritizing one parcel creates a larger imbalance than prioritizing another parcel, the second parcel should be selected. The aforementioned algorithms were tested in simulations and their performance was assessed based on throughput and balance between infeed zone. The 2-merge algorithm outperformed the standard FCFS algorithm in terms of balance, throughput, and

utilization.

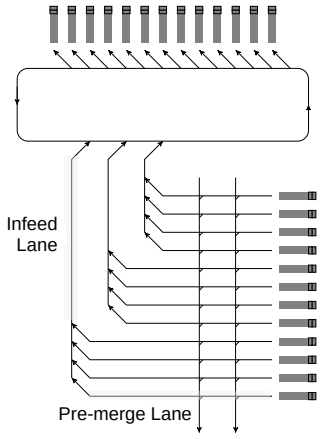

Figure 1.5: Crossorter system with pre-merge lanes, taken from Peeters [2015]

1.2.2 Model Based Design Approach for Merge Balancing

Meens [2017] created a simulation model in which it is possible to compare different merge algorithms. In this simulation he tested the First Come First Serve (FCFS), Round Robin (RR) and Round Robin Adjusted (RRA) algorithms. When the FCFS algorithm is used, a parcel is allocated to a carrier that is closest available to the infeed zone. When the RR algorithm is used, each carrier is reserved for a specific infeed zone. This algorithm comes with a problem: If, for example, a carrier is reserved for infeed zone 1 and it was unable to receive a parcel from that infeed zone, it is still unavailable for the other infeed zone because it is reserved for infeed zone 1. The RRA algorithm frees a carrier from its reservations as soon as it is clear that a parcel will not be allocated to that carrier, eliminating the previously explained problem.

The system that was regarded was a crossorter with seven infeed zone and no pre-merge area, as shown in Figure 1.6. The goal of that study was to find out which control algorithm and parameter setting decreases the imbalance while decreasing the utilization as little as possible. He concluded that FCFS provides the highest utilization while the RR and RRA provide a better balance between infeed zone, with the RRA algorithm slightly outperforming RR in terms of utilization.

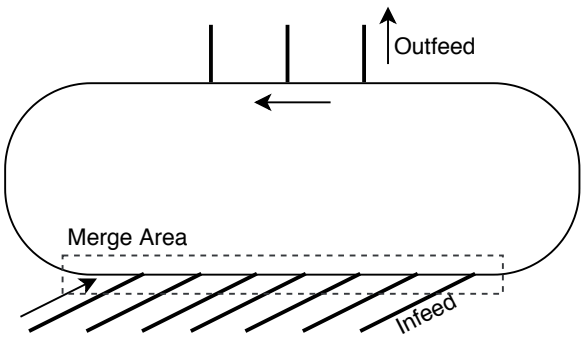

Figure 1.6: Crossorter System Considered by Meens [2017]

Besides testing several merge algorithms, Meens [2017] also investigated the possibility of including an additional velocity profile called Low Velocity (LV) that would create an overlap in between the VS

and OTF time windows visible in Figure 1.1. This velocity profile would be similar to the VS profile, but instead of stopping completely, the parcel slows down to a certain velocity after which it can be accelerated. In case the RR and RRA algorithms were used, the additional velocity profile caused a significant increase in utilization. These conclusions were drawn based on the situation when the system is peak loaded. When the inter-arrival time of parcels increases, the RR and RRA algorithms cause the utilization to decrease dramatically. The proposed solution for this would be an algorithm that could dynamically switch between FCFS and RRA depending on the inter-arrival time of parcels.

Another interesting conclusion from this study was that the starting position of the crossorter has a significant effect on the throughput when a deterministic arrival process is used. This effect was made clear by running multiple simulations with different starting positions. For each simulation run, the starting position was changed by 60mm. In total, 10 simulation runs were done, meaning that starting positions along the entire width of the carrier belt (600mm) were investigated. These tests resulted in large differences in throughput values for each infeed zone. When a stochastic arrival process was used, the differences were much less significant, and the throughput was higher on average.

1.2.3 Design Optimization of a Throw-Catch Infeed Zone

Struik [2018] focused on improving the throughput of a single infeed zone. The goal of the study was to find out how an infeed zone can be designed such that the throughput of the infeed zone is maximized. An important factor that constrains the throughput is the feasibility of parcel trajectories. A trajectory is infeasible when it means that the induct short belts must obtain speeds or accelerations that are not possible because a previous parcel was transported by the induct short belts not too long ago. The parcel could arrive at an induct short belt with a certain velocity while the induct short belt is running at a different velocity. This could cause the parcel to slip and result in an unsuccessful induct on the main loop. Earlier models did not include this trajectory and therefore do not generate the feasible profiles. Struik [2018] proposed a new model in which the infeed zone dynamics are modeled with sufficient detail to determine feasible profiles.

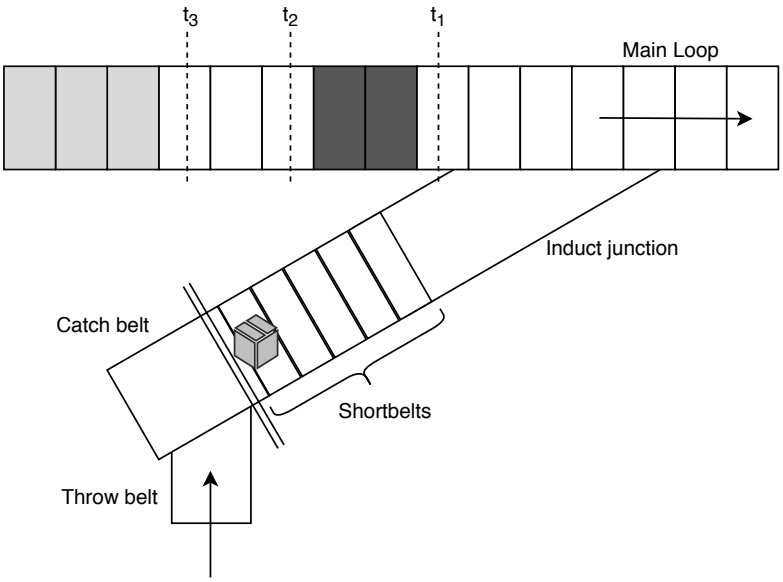

Figure 1.7: Overview of the infeed zone and its conveyor belts

A brief description is given about how the model works. A scheduling action is invoked when an entity arrives. The model calculates three times to define the time windows from Figure 1.1. First, the model

calculates the time it would take to reach the main loop if it were to accelerate immediately. Second, the time is calculated that it takes to reach the loop if it would accelerate at the latest possible point without stopping anywhere in the trajectory. Last, the time is calculated that it would take if the parcel completely stops at the latest possible point and then accelerates immediately after stopping. These times are denoted as t_1, t_2 and t_3 respectively, where the time in between t_1 and t_2 is the same as the OTF window in Figure 1.1 and from t_3 onward corresponds with the VS window. Once these windows are defined, a carrier can be selected. The carriers that the model can choose from are marked grey in Figure 1.7. Finally, the model selects the closest available carrier, as this will result in a faster induct time and therefore a higher throughput for a single infeed zone, and models the trajectory of the parcel on the infeed zone if the trajectory has previously been determined to be feasible.

Three types of alterations were made to experiment with different layouts: induct short belt conveyor, catch belt conveyor and control strategy alterations. Four different induct short belt lengths were can be used in the model: 150, 300, 450 and 600mm. In current systems, 600mm induct short belts are used. Struik [2018] concluded the throughput of the infeed zone increases when the length of the induct short belts is decreased. The sum of the length of all induct short belts should stay the same, so if smaller belts are used, the costs increase since each belt needs a separate motor.

The catch belt can also become a bottleneck in case a large ($\geq$ 500mm) product enters the infeed zone. The catch belt conveyor must come to a complete stop during the handover from the throw belt onto the catch belt to prevent the product from rotating excessively. Imagine a case where a large product is followed by another large product. The first product is transport to the catch belt, which has completely stopped. Then, the catch belt needs to accelerate and maintain the velocity of the next belt until the parcel is handed over to that belt.

Finally, the catch belt must decelerate again and come to a complete stop before it can receive the second large parcel. Decreasing the duration of these processes would increase the throughput. Two alterations were made to decrease the time the parcels spend on the catch belt. The first alteration is aligning the parcels on the throw belt as far to the right as possible. The result of this alteration is that the parcel is closer to the main loop and will, therefore, spend less time on the catch belt. The second alteration was dividing the belt into two sections. Once the parcel has been handed over to the second section, the first section can start decelerating and therefore receive a new large parcel sooner than it would if the catch belt had only one section.

The third alteration involves the control strategy. In real systems, the arrival of products is stochastic. The control system can deal with this stochastic input, but 100% utilization is impossible in this case. If however the arrival process would be synchronized with the carrier passing time, and if only small products that need one carrier are regarded, 100% utilization would be possible. Struik [2018] proposes the creation of a buffer before the infeed zone, from which parcels can be taken at a rate similar to which carriers pass by the infeed zone.

1.3 Problem Statement

Customers may feel dissatisfied with the performance of their crossorter when the main loop is not being fully utilized. Increasing the utilization of crossorters will likely increase customer satisfaction and has, therefore, become a high priority at Vanderlande. Moreover, Vanderlande aims to achieve higher utilization with fewer infeed zones because this would reduce the amount of floor space needed for a crossorter system. The goal of this project is, therefore, to determine how the utilization can be maximized while using three infeed zone. The scopes of the previously conducted research do not match the current goal, so additional research must be conducted.

1.3.1 Project Scope

In current crossorter systems, the flow of parcels starts once they are loaded from trucks onto the first conveyor belt in the system. Crossorters are made to customer specifications so the location of the first belt varies per customer. From there, parcels flow onto an infeed zone lane which is connected to one infeed zone. The interarrival times between parcels on this lane is determined by the rate at which parcels were fed into the system. Parcels are often blocked on the infeed zone when no free carriers are available, decreasing the utilization of the main loop. Buffers that release parcels while taking available carriers and the flow of parcels from other infeed zones into account could potentially solve this problem.

Figure 1.8 shows the system scope that is the focus of this study. It shows a section of the main loop where three infeed zone which each have acceleration zones consisting of two belts. Buffers have been included as a possible extension of the system.

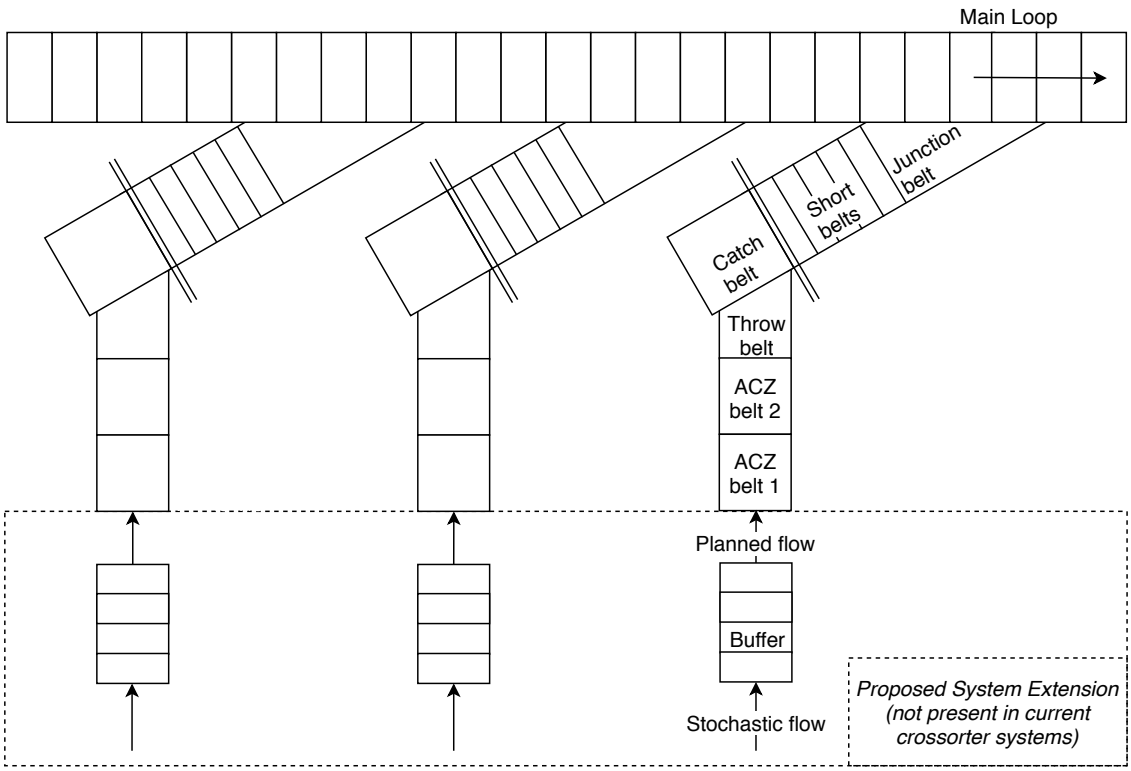

Figure 1.8: System Scope including buffers

1.3.2 Research Questions

Based on the system description from the previous section, the following research questions were be posed:

1. Is it possible to increase the utilization of a system with three infeed zones by using buffers?
2. How does the trajectory downstream of the buffer affect its design?
3. How should the buffer be designed?

1.4 Comparison Between Current Study and Previous Studies

The results of the study conducted by Peeters [2015] do not apply to the current research direction. That study focuses solely on balance between the infeed zone rather than carrier utilization. Only if a future improvement sacrifices balance for a higher utilization, it might be useful to implement some feature that improves the balance similar to the merge algorithm from that study.

Possible re-use of the simulation model was also assessed and deemed to be not useful because the simulations were performed in Automod. The reason for this is that it is difficult to design and test control algorithms in Automod or to make hardware modifications. Peeters [2015] also recommended that alternative software packages should be considered for future research, so with that in mind the decision was made to not use Automod for the current research project. An overview of the main differences between this thesis report and the previously conducted studies is given in Table 1.1.

Table 1.1: Comparison Between Current Study and Previous Studies

	Peeters [2015]	Meens [2017]	Struik [2018]	This Study
Number of infeed zones	3	7	1	3
Performance Measure	Imbalance	Utilization and Imbalance	Throughput	Utilization
Method	Creating new algorithms	Testing existing algorithms	Modeling design changes	Comparing emulation to theoretical utilization
Results	2-merge algorithm outperforms all other algorithms	RRA = best when arrival rate is high FCFS = best when arrival rate is low	Several hardware modifications that may increase throughput	System modifications required for using buffers
Proposed alterations	Implement 2-merge algorithm	Create a dynamic algorithm that changes between RRA and FCFS	Change the short belt length and catch belt design	Create buffers before the acceleration zones
Project goals	Designing an algorithm that improves balance between infeed zones	Testing existing algorithms that improve infeed zone balance	Increasing the throughput of a single infeed zone	Increasing the main loop utilization with three infeed zones

Chapter 2

Current System Analysis

The subject of this research and the scope of the system have been introduced in the previous chapter. The goal of the study is to find out how the utilization of crossorters can be increased with three infeed zones. The proposal is to use buffers located at an earlier point in the crossorter conveyor system. This chapter aims to investigate whether there are any limitations in the succeeding trajectory that will affect the design of the buffer.

After being released from a buffer, parcels must travel over the acceleration zone and infeed zone, shown in Figure 2.1, before they can merge onto the main loop. Each zone is limited concerning the rate at which parcels can be processed. These limitations are expressed as the cycle time of the corresponding zone. After the cycle time has passed, a new parcel may arrive at that zone. The infeed zone and acceleration zones have cycle time limitations concerning parcel length, while the main loop may be limited due to parcel width.

The following sections discuss how the limitations are defined and what they specifically are for each zone. Three types of crossorters, 1200HC, 1200 and 1500 have been investigated because they all have different carrier belt widths. The 1200HC crossorter also has a dual belt carrier system, as opposed to a single belt carrier system of the 1200 and 1500 crossorters. By having multiple crossorters included in this study the importance of the research increases because it encompasses a larger proportion of the company's portfolio. However, the final emulation study only includes the 1500 model because, at that time, this was the only properly functioning emulation model where three infeed zones could be used. The results of this emulation study may still provide sufficient implications because all crossorters have the same basic layout and control system.

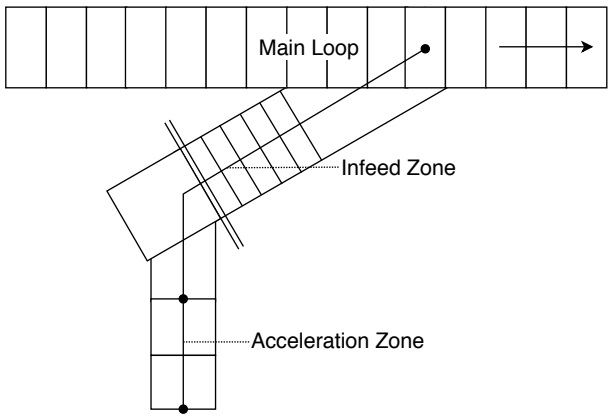

Figure 2.1: Schematic depiction of the trajectory over the current system

2.1 Infeed Zone

The cycle time of each infeed zone belt has been researched and validated at Vanderlande. The infeed zone belt that has the largest cycle time is most important because this determines the rate at which parcels may arrive such that parcels can be smoothly transported over the entire infeed zone. This cycle time was determined per parcel length and then rounded up to the nearest multiplication of the carrier belt interarrival time. The reason for rounding up based on carrier belt interarrival time is because parcels should arrive at the infeed zone at the same rate as the carriers. This would ensure that there is always a carrier available.

2.1.1 Cycle Time Definition

Cycle time can be defined as a value of n for easy comparison with the acceleration zone and main loop, which denotes that every n^{th} carrier belt that passes by an infeed zone can be used to induct a parcel onto the main loop. These values are obtained from Vanderlande documentation.

$$n = \frac{t_{CT,IFZ}}{t_{CT,carrier}} \tag{2.1}$$

where:

$t_{CT,IFZ}$ = Interarrival time of parcels from an infeed zone
$t_{CT,carrier}$ = Interarrival time of carriers

For example, consider a 1500 type crossorter and parcel length 1000 mm. This parcel length corresponds with the following cycle time, according to Vanderlande documentation:

$$t_{CT,IFZ} = 1.74$$

The 1500 crossorter has the following carrier cycle time:

$$t_{CT,carrier} = 0.32$$

Plugging these values into Equation 2.1 gives the following result:

$$n = \frac{1.74}{0.32} = 5.4375$$

This would then be rounded up to the nearest integer, which is $n = 6$ because only whole carrier belts can be used and n pertains to whole carriers belts. Furthermore, a parcel length range per value of n can be defined where each range contains the parcel lengths that would have cycle times that round up to the multiplication that corresponds with that value for n. An example of such ranges and corresponding values of n is given below. The following values are specific for the 1200HC and 1200 crossorters.

$$n = 3 \quad \text{for} \quad 150 \leq L_{parcel} \leq 300 \tag{2.2a}$$
$$n = 4 \quad \text{for} \quad 300 < L_{parcel} \leq 500 \tag{2.2b}$$
$$n = 6 \quad \text{for} \quad 500 < L_{parcel} \leq 1200 \tag{2.2c}$$

The 1500 crossorter has a different parcel length range for the same values of n. The reason for this is that this crossorter type uses an infeed zone that is mechanically different from the infeed zone used by the 1200HC and 1200.

$$n = 3 \quad \text{for} \quad 150 \leq L_{parcel} \leq 400 \tag{2.3a}$$
$$n = 4 \quad \text{for} \quad 400 < L_{parcel} \leq 800 \tag{2.3b}$$
$$n = 6 \quad \text{for} \quad 800 < L_{parcel} \leq 1500 \tag{2.3c}$$

Table 2.1 shows the cycle times for three crossorter types. The difference between these types is the carrier belt width. The carrier belt widths for the 1200HC, 1200 and 1500 are 600, 700 and 800 mm respectively. The main loop speed is constant, so changing the carrier belt width will affect the time it takes for a carrier belt to pass by. The main loop speed was set at 2500 mm/s to determine these values because the majority of crossorters are configured to this speed. The goal of Table 2.1 is to provide an overview of the cycle times per value of n such that they can be easily compared to the cycle times of other zones.

Table 2.1: Cycle time at the infeed zone for three crossorter types

	t_{CT} [s]		
	1200HC	1200	1500
n = 3	0.72	0.84	0.96
n = 4	0.96	1.12	1.28
n = 6	1.44	1.68	1.92

2.2 Acceleration Zone

The trajectory starts at the acceleration zone, which consists of two conveyor belts with variable velocities. The purpose of this zone is to accelerate parcels to the same velocity as the throw belt, which is the first section of the infeed zone where parcels arrive. The lengths and velocities of these belts vary, depending on the type of crossorter. The front of the parcel is located at the start of the trajectory, as shown in Figure 2.3, so the rest of the trajectory is following the way that the parcel head travels.

The moment that a parcel accelerates on each belt can vary and it affects the duration of the cycle time. To evaluate this, the composition of the cycle times of each acceleration zone belt in case of accelerating late and early are evaluated per parcel length. This will give the minimum and maximum cycle times, indicating the range of feasible cycle times per parcel length. This is then compared with the cycle times of the infeed zone, which will show whether or not the acceleration zone is the bottleneck of the flow of parcels.

2.2.1 Belt Cycle Time Definition

The infeed zone and acceleration zone consist of multiple belts. Each belt that can accelerate has a cycle time that is defined as the time it takes for a conveyor belt to accelerate, hand over the parcel to the succeeding conveyor belt and decelerate the belt in question. Once the belt has decelerated, it can accept a new parcel without slipping. The following equation shows how the cycle time is calculated.

$$t_{CT} = t_{LV} + t_a + t_{HV} + t_d \tag{2.4}$$

where:

t_{CT} = Cycle time
t_{LV} = Time spent at low velocity
t_a = Time needed to accelerate from low to high velocity
t_{HV} = Time spent at high velocity
t_d = Time needed to decelerate from high to low velocity

Figure 2.2 illustrates the cycle time definition and shows an example of three conveyor belts where the middle belt's cycle time is evaluated. This belt accelerates parcels from v_{LV} to v_{HV}. After the cycle time has passed, the middle belt is ready to accept a new parcel.

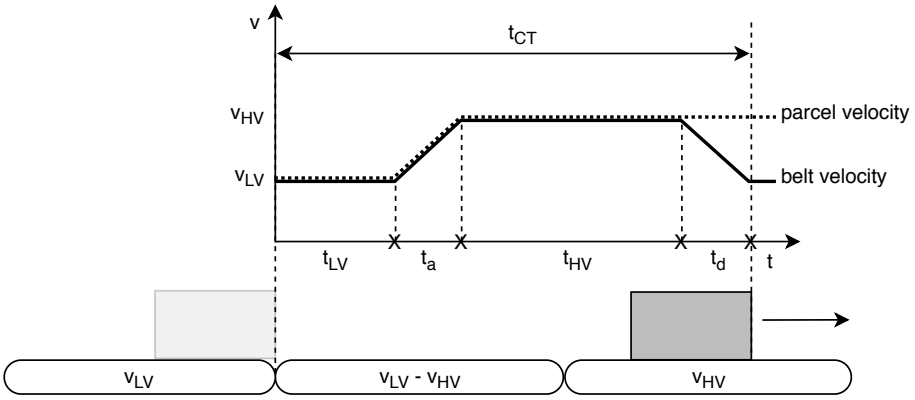

Figure 2.2: Cycle Time Definition

The following sections aim to discuss where the longest cycle time is located in the trajectory shown in Figure 2.1. This will be the bottleneck of the entire trajectory and therefore determine the rate at which parcels should be released by an upstream buffer.

2.2.2 Early Acceleration

If the acceleration starts as soon as possible, it will reach the required velocity sooner and spend more time at a higher velocity than if it would accelerate later. This gives the minimum cycle time of an ACZ (acceleration zone) belt. The trajectories of the parcel head, which is the front end of the parcel, that correspond with this behavior are shown in Figure 2.3 and Figure 2.4.

The acceleration starts immediately when the parcel tail, the rear end of the parcel, is located in between the two belts ACZ 1 and ACZ 2. After being accelerated to the acceptance velocity of the next belt, it still has to travel a distance of $x_{r,n} + L_{parcel}$. The distance $x_{r,n}$ depends on the acceleration belt length L_n, parcel length L_{parcel} and the distance needed for acceleration $x_{a,n}$ where n is the acceleration zone belt number.

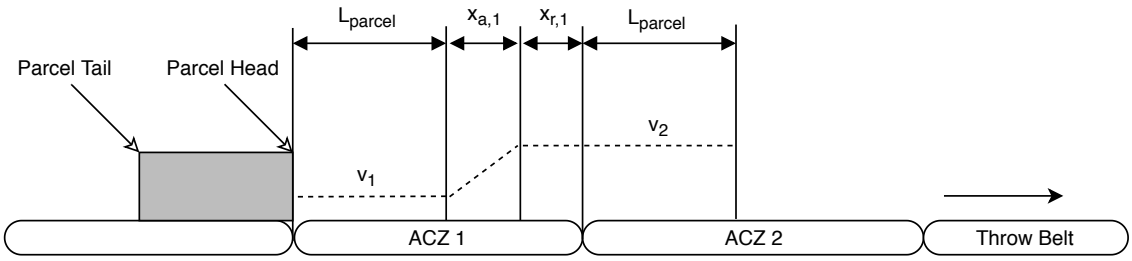

Figure 2.3: Velocity trajectory on the acceleration zone 1 belt in case of early acceleration

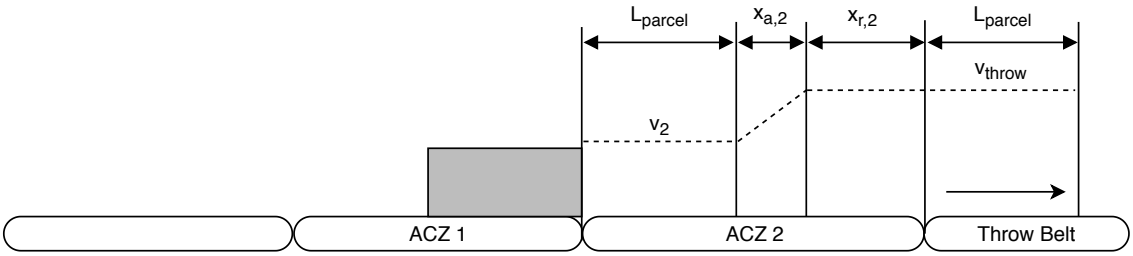

Figure 2.4: Velocity trajectory on the acceleration zone 2 belt in case of early acceleration

2.2.3 Late Acceleration

If the acceleration starts as late as possible, the maximum cycle time of the acceleration zone belt is reached. Figure 2.5 and Figure 2.6 show the trajectories of a parcel on acceleration zone 1 and 2 if it is accelerated as late as possible. In this case, the acceleration starts when the parcel head is still far enough away from the next belt to start accelerating and reach the required velocity.

Once a parcel has accelerated to the velocity of the next belt, it starts being handed over to the next belt. The duration of this hand-over depends on the parcel length because the tail of the parcel has to pass the point in between the two acceleration zone belts.

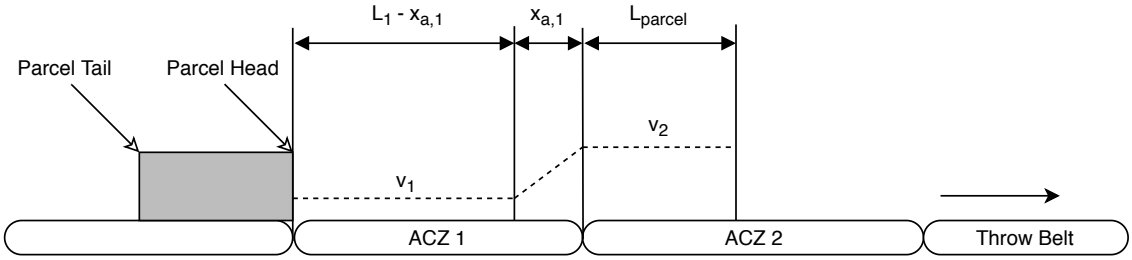

Figure 2.5: Velocity trajectory on the acceleration zone 1 belt in case of late acceleration

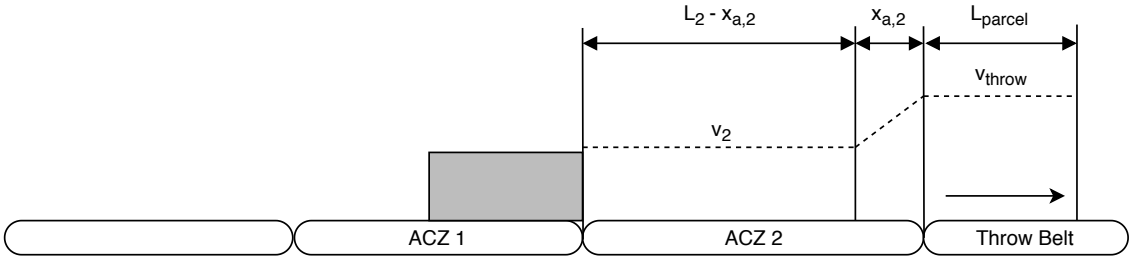

Figure 2.6: Velocity trajectory on the acceleration zone 2 belt in case of late acceleration

2.2.4 Cycle Time Equations

Early Acceleration

The following equations for the acceleration belt cycle time have been derived from Figure 2.3 and Figure 2.4. The terms of Equation 2.5 and Equation 2.6 correspond with the terms of Equation 2.4.

$$t_{CT,ACZ1} = \frac{L_{parcel}}{v_1} + t_{a,1} + \frac{x_{r,1} + L_{parcel}}{v_2} + t_{d,1} \tag{2.5}$$

$$t_{CT,ACZ2} = \frac{L_{parcel}}{v_2} + t_{a,2} + \frac{x_{r,2} + L_{parcel}}{v_{throw}} + t_{d,2} \tag{2.6}$$

Late Acceleration

The cycle time for the late acceleration of a parcel is based on Figure 2.5 and Figure 2.6 and defined as follows. Similarly to the case of early acceleration, the terms correspond with Equation 2.4.

$$t_{CT,ACZ1} = \frac{L_1 - x_{a,1}}{v_1} + t_{a,1} + \frac{L_{parcel}}{v_2} + t_{d,1} \tag{2.7}$$

$$t_{CT,ACZ2} = \frac{L_2 - x_{a,2}}{v_2} + t_{a,2} + \frac{L_{parcel}}{v_{throw}} + t_{d,2} \tag{2.8}$$

Terms

$$t_{a,1} = t_{d,1} = \frac{v_1 - v_2}{a} \tag{2.9}$$

$$t_{a,2} = t_{d,2} = \frac{v_2 - v_{throw}}{a} \tag{2.10}$$

$$x_{a,1} = v_1 t_{a,1} + \frac{1}{2} a t_{a,1}^2 \tag{2.11}$$

$$x_{a,2} = v_2 t_{a,2} + \frac{1}{2} a t_{a,2}^2 \tag{2.12}$$

$$x_{r,1} = L_1 - x_{a,1} - L_{parcel} \tag{2.13}$$

$$x_{r,2} = L_2 - x_{a,2} - L_{parcel} \tag{2.14}$$

where:

L_n	= Belt Length of acceleration zone n
v_n	= Acceptance velocity of acceleration zone belt n
$x_{a,n}$	= Acceleration distance on acceleration zone belt n
$x_{r,n}$	= Remaining distance on acceleration zone n after acceleration
a	= Acceleration
L_{parcel}	= Parcel Length
v_{throw}	= Acceptance velocity of the throw belt

2.2.5 Effect of the Acceleration Zone on Infeed Zone Capacity

The cycle times in case of early and late acceleration have been determined for each acceleration zone belt and three crossorter types. These values are then compared to the cycle times at the infeed zone, given in Table 2.1. If the cycle time of the acceleration zone belt is higher than the corresponding cycle times, meaning that the acceleration zone would bottleneck the parcel flow, the corresponding cell in Table 2.2 is marked grey. For example, consider a 1500 type crossorter where a parcel is inducted with length 300 mm. The smallest feasible cycle times on the acceleration zone 1 and acceleration zone 2 belts are 1.10 s and 1.12 s respectively. The cycle time of this parcel on the infeed zone would be 0.96 s, so the acceleration zone would be the bottleneck in this case. The two corresponding cells are marked grey in Table 2.2.

What this table shows is that the acceleration zone could be a considerable bottleneck for many parcel lengths, especially for smaller parcels. For the 1200HC and 1200 crossorters, these are parcels up to 300 mm long. For the 1500 crossorter, these are parcels up to 400 mm long. In real systems, a higher cycle time may be measured on the acceleration zone, but this would likely be without a smooth hand-over to the next conveyor belt. This might cause slippage and therefore a deviation from the planned trajectory to the main loop. The results from this section should be validated in a real system and the amount of slippage should be quantified. If there is a significant amount of slippage, the acceleration zone should be redesigned.

Table 2.2: Cycle times per acceleration zone belt and parcel length

	1200HC				1200				1500			
Parcel	AZC1		ACZ2		AZC1		ACZ2		AZC1		ACZ2	
Length	Early	Late	Early	Late	Early	Late	Early	Late	Early	Late	Early	Late
150	0.93	1.05	0.91	1.02	0.93	1.05	0.91	0.94	1.01	1.10	1.04	1.12
200	0.96	1.08	0.94	1.04	0.96	1.08	0.94	0.96	1.04	1.13	1.06	1.15
250	1.00	1.11	0.97	1.06	1.00	1.11	0.97	0.99	1.07	1.15	1.09	1.17
300	1.03	1.13	1.00	1.09	1.03	1.13	1.00	1.01	1.10	1.18	1.12	1.20
350	1.06	1.16	1.02	1.11	1.06	1.16	1.02	1.04	1.13	1.21	1.15	1.22
400	1.10	1.19	1.05	1.13	1.10	1.19	1.05	1.06	1.16	1.24	1.17	1.25
450	1.13	1.22	1.08	1.16	1.13	1.22	1.08	1.08	1.20	1.27	1.20	1.27
500	1.16	1.25	1.11	1.18	1.16	1.25	1.11	1.11	1.23	1.29	1.23	1.30
550	1.20	1.27	1.14	1.21	1.20	1.27	1.14	1.13	1.26	1.32	1.26	1.32
600	1.23	1.30	1.16	1.23	1.23	1.30	1.16	1.15	1.29	1.35	1.29	1.35
650	1.26	1.33	1.19	1.25	1.26	1.33	1.19	1.18	1.32	1.38	1.31	1.37
700	1.30	1.36	1.22	1.28	1.30	1.36	1.22	1.20	1.35	1.40	1.34	1.40
750	1.33	1.38	1.25	1.30	1.33	1.38	1.25	1.23	1.38	1.43	1.37	1.42
800	1.36	1.41	1.27	1.32	1.36	1.41	1.27	1.25	1.41	1.46	1.40	1.45
850	1.40	1.44	1.30	1.35	1.40	1.44	1.30	1.27	1.45	1.49	1.42	1.47
900	1.43	1.47	1.33	1.37	1.43	1.47	1.33	1.30	1.48	1.52	1.45	1.50
950	1.46	1.50	1.36	1.40	1.46	1.50	1.36	1.32	1.51	1.54	1.48	1.52
1000	1.50	1.52	1.39	1.42	1.50	1.52	1.39	1.34	1.54	1.57	1.51	1.55
1050	1.53	1.55	1.41	1.44	1.53	1.55	1.41	1.37	1.57	1.60	1.54	1.57
1100	1.56	1.58	1.44	1.47	1.56	1.58	1.44	1.39	1.60	1.63	1.56	1.60
1150	1.60	1.61	1.47	1.49	1.60	1.61	1.47	1.42	1.63	1.65	1.59	1.62
1200	1.63	1.63	1.50	1.52	1.63	1.63	1.50	1.44	1.66	1.68	1.62	1.65
1250	-	-	-	-	-	-	-	-	1.70	1.71	1.65	1.67
1300	-	-	-	-	-	-	-	-	1.73	1.74	1.67	1.70
1350	-	-	-	-	-	-	-	-	1.76	1.77	1.70	1.72
1400	-	-	-	-	-	-	-	-	1.79	1.79	1.73	1.75
1450	-	-	-	-	-	-	-	-	1.82	1.82	1.76	1.77
1500	-	-	-	-	-	-	-	-	1.85	1.85	1.79	1.80

2.3 Main Loop

The previous sections discussed the limitations of the acceleration infeed zones, which have different cycle times depending on the parcel length. This is different from the main loop, where the parcel width and the planned sequence determine the cycle time. Depending on the parcel width, multiple carrier belts may need to be reserved for a parcel. This may affect the moment when a parcel should be released from an upstream buffer.

If only the cycle times of the acceleration zone and infeed zone are used to determine release times from the buffer, the number of carrier belts needed on the loop is not considered. The result can be that a parcel cannot be inducted because there are not enough carrier belts available, it will be blocked on the infeed zone and the utilization of the main loop will be decreased. This section discusses the limitations of the main loop concerning specific parcel width ranges and concludes with the implications it has for the upstream buffer.

The reason that wider parcels may need multiple carrier belts is that they could overlap onto other carrier belts. A collision could occur when other parcels induct onto the carrier belts next to it. This problem is solved by reserving multiple carrier belts for parcels whose width exceeds a defined width threshold. The width threshold depends on the carrier belt width, which differs depending on the crossorter type. The 1200HC, 1200 and 1500 crossorters use carrier belts that have a width of 600, 700 and 800 mm respectively. According to Vanderlande documentation, the width thresholds that are defined for these carrier belt widths are 400, 500 and 600 mm.

2.3.1 Double and Triple Belt Reservation

The number of carrier belts needed for a wider parcel depends on the crossorter type. The 1200HC crossorters uses two carrier belts while the 1200 and 1500 crossorters use three belts. This limits the flow of parcels from a buffer. For example, if only wide parcels are supplied to a 1500 crossorter, every 9^{th} carrier can be used because triple belt reservation is needed. These could be wide parcels that are 450 mm in parcel length, meaning that according to the infeed zone capacity, it could utilize every 3^{rd} carrier. The capacity of the infeed zone is therefore limited by the main loop.

The difference in dual and triple belt is because the 1200 and 1500 crossorters use single belt carriers. If only two carrier belts are used in those systems, the center of mass might reside in between two belts during a curve, potentially shifting the position on the loop. With triple belt reservation, the center of mass will likely reside on the middle carrier belt, reducing the probability of slipping during curves. Figure 2.7 and Figure 2.8 show the top view of a loop section in case of double belt and triple belt reservation respectively.

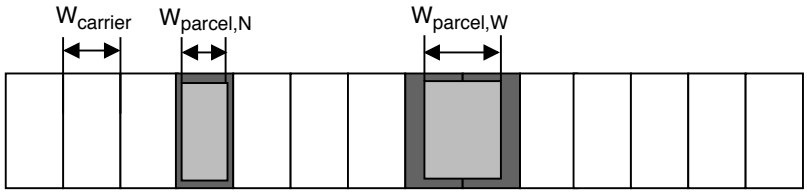

Figure 2.7: Schematic top view of loop section with double belt reservation

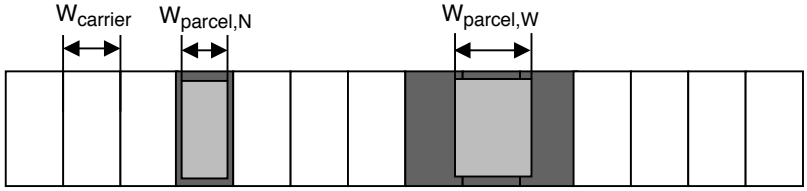

Figure 2.8: Schematic top view of loop section with triple belt reservation

2.3.2 Main Loop Cycle Time

The cycle time of the main loop is defined as the amount of time it takes before another parcel is inducted from a single infeed zone. Consider a crossorter with three infeed zones where each infeed zone will sequentially induct parcels according to a round-robin algorithm such that a pattern occurs on the main loop as shown in Figure 2.9. The origin of each parcel is denoted by the number 1, 2 or 3.

The cycle time of infeed zone 1 is considered in this example. Infeed zone 1 could induct parcels on the last two locations marked with dashed rectangles. The second dashed rectangle from the right is located 3 carrier belts away from the first one, so $n = 3$ in this case. The third one, however, is located further away because a parcel that exceeds the width threshold has been inducted by infeed zone 2, resulting in $n = 5$.

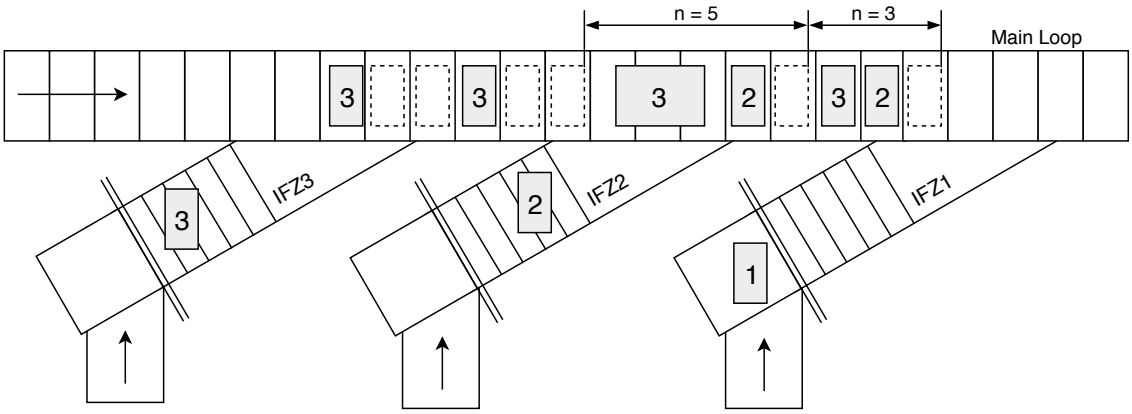

Figure 2.9: Top view of a 1500 type crossorter with three infeed zones

It is can be concluded that the cycle time at which an infeed can induct parcels onto the main loop is affected by the arrival order of parcels from other infeed zones. Table 2.3 shows an overview of all possible merge sequences considering a crossorter with three infeed zones. that sequentially induct parcels. Whether or not the parcel widths exceed the width threshold is denoted by "Narrow" or Wide". The two examples given in Figure 2.9 where n = 3 and n = 5 correspond respectively with sequences 1 and 3 or sequences 2 and 4, depending on whether the succeeding parcel from infeed zone 1 is Narrow or Wide.

Table 2.3: Cycle Time per variation of merge sequence for parcels coming from infeed zone 1

	Parcel merge sequence from infeed zone (IFZ)				Sorter Type					
					1200HC		1200		1500	
Sequence	IFZ 1→	IFZ 2→	IFZ 3→	IFZ 1	n	$t_{CT,1}$	n	$t_{CT,1}$	n	$t_{CT,1}$
1	Narrow	Narrow	Narrow	Narrow	3	0.72	3	0.84	3	0.96
2	Narrow	Narrow	Narrow	Wide	4	0.96	4	1.12	5	1.60
3	Narrow	Narrow	Wide	Narrow	4	0.96	4	1.12	5	1.60
4	Narrow	Narrow	Wide	Wide	5	1.20	5	1.40	7	2.24
5	Narrow	Wide	Narrow	Narrow	4	0.96	4	1.12	5	1.60
6	Narrow	Wide	Narrow	Wide	5	1.20	5	1.40	7	2.24
7	Narrow	Wide	Wide	Narrow	5	1.20	5	1.40	7	2.24
8	Narrow	Wide	Wide	Wide	6	1.44	6	1.68	9	2.88
9	Wide	Narrow	Narrow	Narrow	3	0.72	3	0.84	3	0.96
10	Wide	Narrow	Narrow	Wide	5	1.20	5	1.40	5	1.60
11	Wide	Narrow	Wide	Narrow	4	0.96	4	1.12	5	1.60
12	Wide	Narrow	Wide	Wide	5	1.20	5	1.40	7	2.24
13	Wide	Wide	Narrow	Narrow	4	0.96	4	1.12	5	1.60
14	Wide	Wide	Narrow	Wide	5	1.20	5	1.40	7	2.24
15	Wide	Wide	Wide	Narrow	5	1.20	5	1.40	7	2.24
16	Wide	Wide	Wide	Wide	6	1.44	6	1.68	9	2.88

What Table 2.3 shows is that there is no generic cycle time of the main loop per parcel length or width. Whether or not a sequence of parcel arrivals is feasible depends the capability of the crossorter to change the space between parcels before they arrive at the acceleration zone.

2.4 Discussion

The findings that were given in this chapter have some implications on the proposed system extension with buffers. Firstly, the findings regarding the acceleration zone imply that a redesign may be necessary. If slip occurs in the acceleration zone after a parcel is released from a buffer than that parcel might not be able to induct on the planned carrier belt. For now, it remains unclear whether or not slippage caused by the acceleration zone would significantly disrupt the flow parcels. The significance of slippage caused by this zone should be investigated before deciding if redesigning the acceleration zone is truly necessary.

Secondly, the findings regarding the cycle times of the acceleration zone, infeed zone, and main loop show that it is necessary to know both the length and width of parcels located in the trajectory before the acceleration zone. When the dimensions and locations of parcels are known before they arrive at the acceleration zone, their location on the loop can be planned. Based on this planned arrival, the required distance between these parcels can be determined. Whether or not this can be accomplished depends on the capability to adjust distances between parcels before the acceleration zone. A buffer could potentially provide this capability.

Chapter 3

Solution Design

The previous chapter addressed the parcel trajectory on the acceleration zone, the infeed zone, and the main loop. This was necessary because the solutions presented in this chapter need to take the cycle time limitations of that trajectory into account. The method of determining the capacity of each of these sections was explained and a final discussion was given to provide insight into the cycle time limitations and it's implications for designing a buffer.

In this chapter, two types of buffer designs are given and reviewed. First, the feasibility of a buffer where parcels decelerate and then wait to be accelerated is considered. This turned out to be infeasible for many parcel lengths because completely decelerating and then accelerating again causes the cycle time to exceed that of the infeed in many cases. An alternative buffer design is then given where parcels do not stop completely but are decelerated to one of several available lower velocities.

3.1 Buffer With Acceleration From Standstill

The buffer design considered in this section is one where parcels stand still and are accelerated according to a planned sequence on the main loop.

The buffer would have to consist of a conveyor belt that can accelerate a parcel to the velocity of the first acceleration zone belt before being handed over. This conveyor belt should be long enough to queue a sufficient number of parcels to plan sequences on the main loop. A crucial requirement is that the gap between parcels must be sufficiently large such that they can decelerate and accelerate before they reach the next belt. If the gaps are too small, parcels might reach the next belt before being accelerated to the required velocity, which could cause slippage.

This gap between parcels consists of the distances needed for deceleration x_d and acceleration x_a and an added 100 mm to ensure that the gap is never too small. According to Vanderlande documentation the trajectory before the buffer is able to create gaps with an accuracy of ± 100 mm. If the gap size of incoming parcels is set to $x_a + x_a + 100$, then the gap range of arriving parcels is as follows:

$$x_d + x_a \leq x_{gap} \leq x_d + x_a + 200$$

Figure 3.1 shows a schematic depiction of the required gaps between parcels. The result is that the window x_w between two parcel heads is such that the required cycle time is reached to achieve a planned sequence of arrivals on the main loop. The term 'window' is used in this case, because this corresponds with the cycle time definition shown in Figure 2.2.

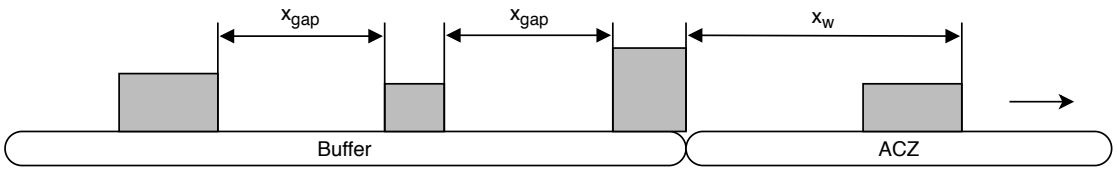

Figure 3.1: Buffer where parcels can stand still before being handed over to the acceleration zone

3.1.1 Cycle Time Definition

Considering the gap requirement of $x_d + x_a + 100$, the feasibility of the proposed buffer can be determined by comparing the cycle times of the buffer to the cycle time of the infeed. If for a specific parcel length the cycle time of the buffer is longer than that of the infeed, then the buffer would have a lower capacity than the infeed. This means that the buffer could potentially bottleneck the flow and decrease the main loop utilization. In those cases, the buffer design was considered infeasible for that specific parcel length. This cycle time is defined as follows:

$$t_{CT,Buffer} = t_d \quad + t_a \quad + t_{AZC1} \tag{3.1}$$

$$t_{CT,Buffer} = \frac{v_{AZC1}}{a} + \frac{v_{AZC1}}{a} + \frac{L_{parcel} + 100}{v_{ACZ1}} \tag{3.2}$$

where:

v_{ACZ1} = Velocity of the first belt in the acceleration zone
a = Acceleration
L_{parcel} = Parcel Length

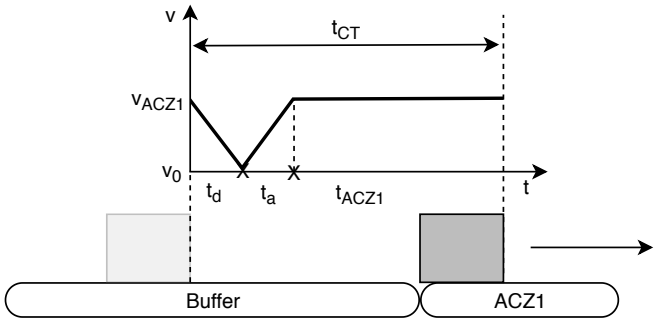

Figure 3.2: Required gaps between parcels to accelerate from standstill

3.1.2 Feasibility Analysis

The feasibility of a buffer as described in this section has been investigated for the 1200HC, 1200 and 1500 crossorter. This was done by comparing the cycle times of the buffer to that of the infeed. An important variable that determines the cycle time is the velocity of the first acceleration zone belt. This velocity is different per crossorter type. For the 1200HC and 1200 crossorter, this is 1500 mm/s and for the 1500 crossorter, it is 1600 mm/s.

After the cycle times per parcel length was determined, they were compared with the cycle times of the infeed for the same parcel length. The infeed cycle times are taken from Table 2.1. Figure 3.3, Figure 3.4 and Figure 3.5 show plots of the the infeed cycle times and buffer cycle times for the three crossorter types in question. The parcel lengths whose buffer cycle times were larger than the infeed cycle time have dark grey areas. This denotes that the buffer would be a bottleneck in the parcel flow. Figure 3.4 and Figure 3.5 show small dark grey areas in between light grey areas, this is due to the discrete increase of cycle time for a higher parcel length range.

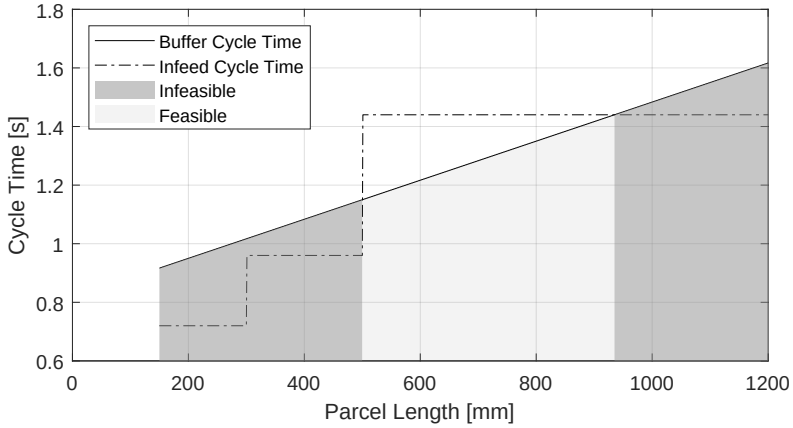

Figure 3.3: Crossorter Type 1200HC

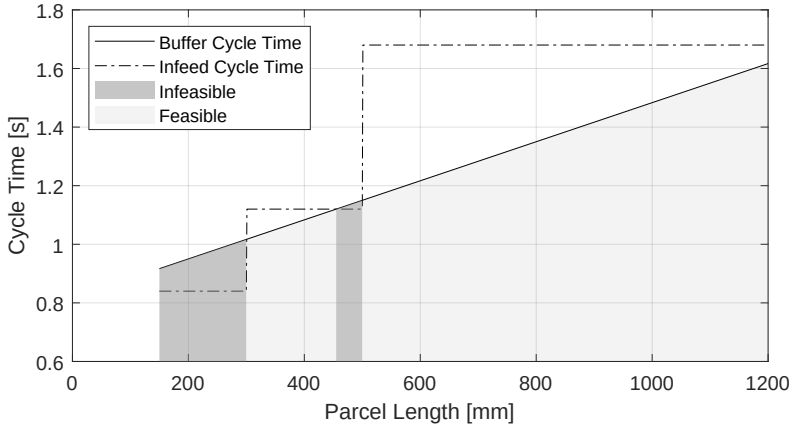

Figure 3.4: Crossorter Type 1200

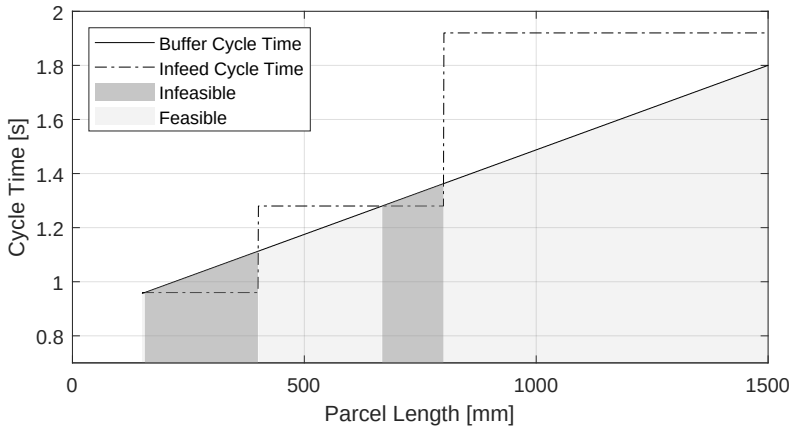

Figure 3.5: Crossorter Type 1500

All three figures show parcel lengths where the cycle time of the buffer exceeds that of the infeed. It can be concluded that introducing such a buffer to the system could potentially decrease the throughput of parcels. This would lower the main loop utilization and not contribute to the research problem, therefore other solutions must be considered.

Decreasing The Velocity of the First Acceleration Belt

It is possible to reduce the cycle time of the buffer by decreasing the velocity of the first acceleration zone belt. This would reduce the time needed for deceleration and acceleration, but it will increase the time needed to hand over the parcel to the first acceleration zone belt if the velocity is lowered excessively. Figure 3.6 shows how $t_{CT,Buffer}$ changes when lowering v_{ACZ1}. For this example, parcel lengths 150 mm and 450 mm were chosen. The infeed zone cycle time of 0.96 s for crossorter type 1500 and this parcel length range was also plotted for comparison, showing where the buffer cycle time would be lower than that of the infeed.

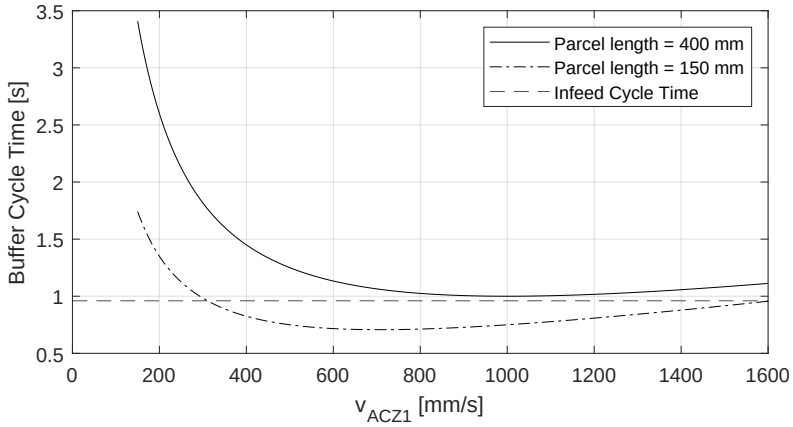

Figure 3.6: Effect of lowering v_{ACZ1} on $t_{CT,Buffer}$

As can been from Figure 3.6, lowering the velocity is not effective for all parcel lengths. In the case of $L_{parcel} = 400$ mm on a 1500 type crossorter, there isn't any lower v_{ACZ1} that would decrease the cycle time far enough. Considering all three considered crossorter types, there appears to be no lower v_{ACZ1} where the buffer cycle time is lower than the infeed cycle for all parcel lengths. Therefore, it can be concluded that another alternative solution is required. An overview of all possible parcel lengths and

lower velocities is shown in Appendix A. The cells of those tables in that appendix are marked grey in case the buffer cycle time exceeds the infeed cycle time for a particular combination of v_{ACZ1} and parcel length.

3.2 Buffer with Lower Velocities

Cycle time reduction could be achieved by allowing parcels to decelerate to a lower velocity rather than stopping completely. The deceleration and acceleration times would decrease, and since these are what makes the cycle buffer cycle time too long, it would reduce the cycle time. With the motor types in current Vanderlande systems, it is possible to use configure four different velocities, which would allow for deceleration to three lower velocities. The way that these velocities have been determined is presented in the following subsections. The velocities are denoted as v_4, v_3, v_2, and v_1 where v_4 is equal to the acceptance velocity of the first acceleration zone belt and the others are lower velocities.

An important requirement of this buffer is that is must be able to enlarge the windows of incoming parcels. Since only lower velocities are used, the initial window must always be smaller than the required window. The accuracy with which windows are created is ± 100 mm, so the initial window must be 100 mm smaller than the window based on parcel length $x_{w,n}$ where n is the cycle time according to parcel length from the infeed zone.

The inaccuracy causes an error, denoted as ε_w in Figure 3.7, which ranges between -100 and 100 mm. How this error can be corrected is addressed in Sections 3.2.1 and 3.2.2. Additionally, the buffer must be able to change the window if the planned position on the loop requires the window to be extended by $\Delta x_{w,\Delta n}$, which corresponds with a difference of n carriers compared to the planned position on the main loop. This is addressed in Section 3.2.3.

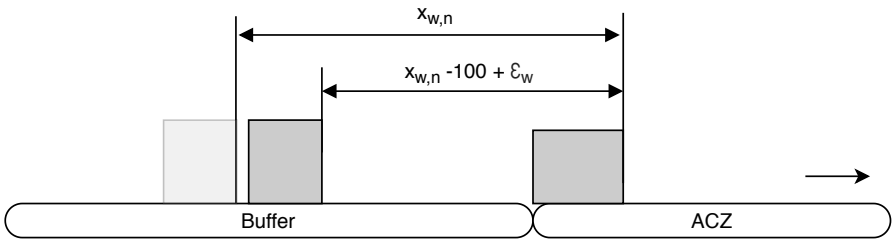

Figure 3.7: Correcting the window error caused by the preceding trajectory

3.2.1 Window Error Correction Using Lower Velocities

If the buffer cannot correct the error ε_w of a window, the required cycle time may not be feasible. The three lower velocities must be selected such that this error can always be corrected. This can be achieved by determining the window range for a given lower velocity of v_m where m = 1,2,3. The window range is the range between the minimum and maximum window for a given parcel size and cycle time. Figure 3.8 shows a schematic depiction of the definition of the minimum window $x_{w,min}$ and maximum window $x_{w,max}$. These windows depend on the time spent at low or high velocity respectively, which is shown in the graph in Figure 3.8.

The additional velocities should be chosen such that there is no gap between the minimum and maximum window ranges, forming a larger total window range. Figure 3.9 shows a graph of three window ranges corresponding with the three lower velocities. Note how the minimum windows of v_1 and v_2 align with the maximum windows of v_2 and v_3 respectively. This shows that it is possible to have a total range which consists of the three separate window ranges.

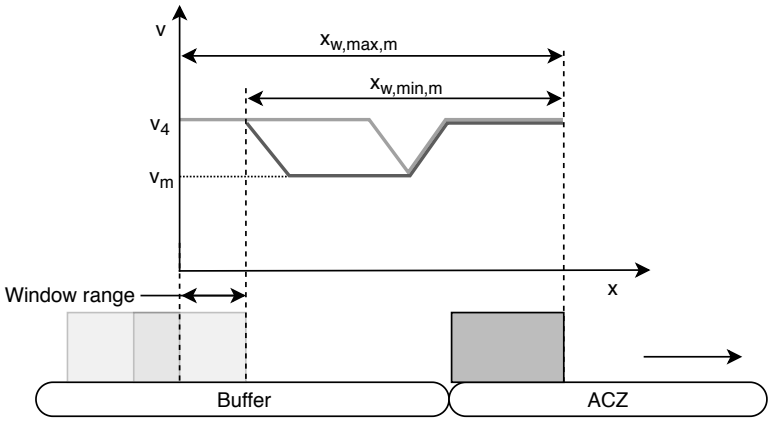

Figure 3.8: Schematic depiction of the window range resulting from using a lower velocity

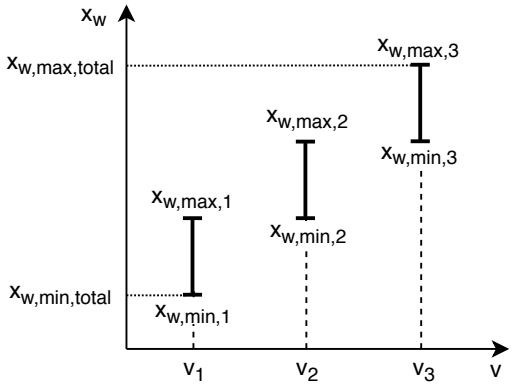

Figure 3.9: Total window range definition

The total range is the difference between the minimum window $x_{w,min,3}$ and $x_{w,max,1}$. The total range must be large enough to deal with the accuracy of ± 100 mm and the additional 100 mm window reduction created before the buffer. Moreover, the gap must still be large enough to make parcels come to a complete stop in case no carrier can be reserved. This gap has been discussed in Section 3.2.1, where it was determined that the gap should be $2x_a + 100$ mm, where x_a is the distance needed for acceleration or deceleration. A minimum window may be determined that is not usable because of the minimum gap needed for a complete standstill.

Minimum Window

The minimum of a given lower velocity and cycle time can be found with the equation for the cycle time, Equation 2.4. The parcel travels at v_m for as long as possible during the cycle time. This means that the distance at this velocity is shorter than the distance if it would travel at v_4, resulting in the minimum window. The corresponding cycle time equation is as follows:

$$t_{CT} = t_d + t_{LV} + t_a + t_{HV}$$
$$t_{CT} = \frac{v_4 - v_m}{a} + \frac{x_m}{v_m} + \frac{v_4 - v_m}{a} + \frac{L_{parcel}}{v_4}$$

The variable, in this case, is x_m, which is used to determine the minimum window with the following equation.

$$x_{w,min,m} = x_{d,m} + x_m + x_{a,m} + L_{parcel} \tag{3.3}$$

The terms from Equation 3.3 are obtained from the trajectory shown in Figure 3.10.

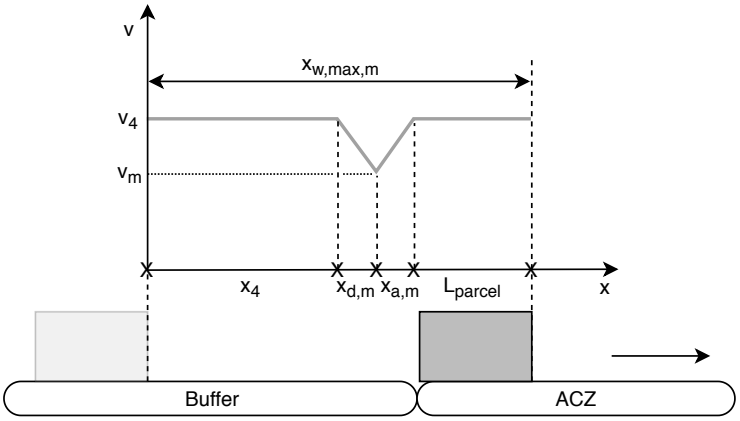

Figure 3.10: Maximum window definition

Maximum Window

The maximum of a given lower velocity and cycle time can be found similarly to the maximum window, with Equation 2.4. The time at low velocity L_{LV} is omitted because the maximum window is achieved if the parcel travels with a velocity of v_4 for as long as possible. In this case, the cycle time equation is as follows:

$$t_{CT} = t_{d,m} + t_{a,m} + t_{HV}$$
$$t_{CT} = \frac{v_4 - v_m}{a} + \frac{v_4 - v_m}{a} + \frac{x_4 + L_{parcel}}{v_4}$$

The only variable, in this case, is x_4, which is the distance traveled at high velocity. Once this value is determined, the maximum window can be determined with the following equation:

$$x_{w,min,m} = x_4 + x_{d,m} + x_{a,m} + L_{parcel} \tag{3.4}$$

This terms in Equation 3.4 are derived from Figure 3.11, which shows the parcel trajectory in case the window is $x_{w,min,m}$ between two parcels.

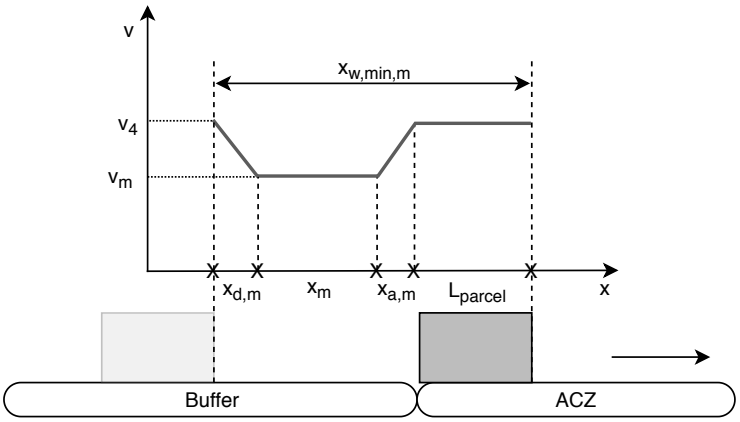

Figure 3.11: Minimum window definition

Acceleration From Standstill Constraint

As explained earlier in subsection 3.2.1, the window should be large enough to allow for the parcel to come to a standstill and then be accelerated again without slipping. As can be seen in Figure 3.12, this window consists of the distances needed for acceleration and deceleration, 100 mm to account for the potential window error and the parcel length.

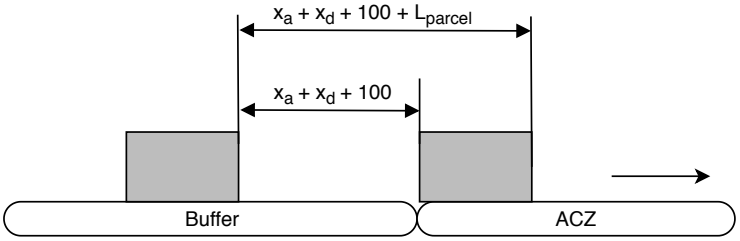

Figure 3.12: Acceleration From Standstill Gap and Window

If a calculated window $x_{w,max,m}$ or $x_{w,min,m}$ of a lower velocity v_m is smaller than the window needed for standstill, it should not be used when considering the total window range. If it would be used anyway, then in case a parcel must be queued on the buffer it might not be possible to stop a parcel and accelerate it to the right velocity before the parcel head reaches the next belt. Instead, the values of those windows should be set to value of the window needed for standstill. This way, the optimal lower velocities can be determined while ensuring that acceleration from standstill is still possible when needed.

3.2.2 Determining the Lower Velocities

The method for determining the optimal lower velocities was found by creating an example using an arbitrarily chosen crossorter type and parcel length. The crossorter 1500 and a parcel length of 200 mm were used in the example shown in this subsection. The choice of parcel length determines the

optimal velocities for that parcel length, so the effect on the window range for each parcel must be checked for size and possible gaps between individual ranges. First, a summary of all input variables is given.

$v_4 = 1600$ [mm/s]
$v_{sorter} = 2500$ [mm/s]
$W_{carrier} = 800$ [mm]
$a = 4000$ [mm/s^2]
$L_{parcel} = 200$ [mm]

The minimum and maximum windows for each v_3 have been determined using the given variables and Equation 3.4 and Equation 3.3. Following this, v_2 and v_1 were determined such that the minimum window using v_3 is equal to the maximum window using v_2 and minimum window using v_2 is equal to the maximum window using v_1. Figure 3.13 shows that the total window range is extended per each additional lower velocity. If only lower velocity v_3 could be used, then the window total range is the difference between $x_{w,max,3}$ and $x_{w,min,3}$. If an additional lower velocity v_2 is used, than the total window range is the difference between $x_{w,max,3}$ and $x_{w,min,2}$.

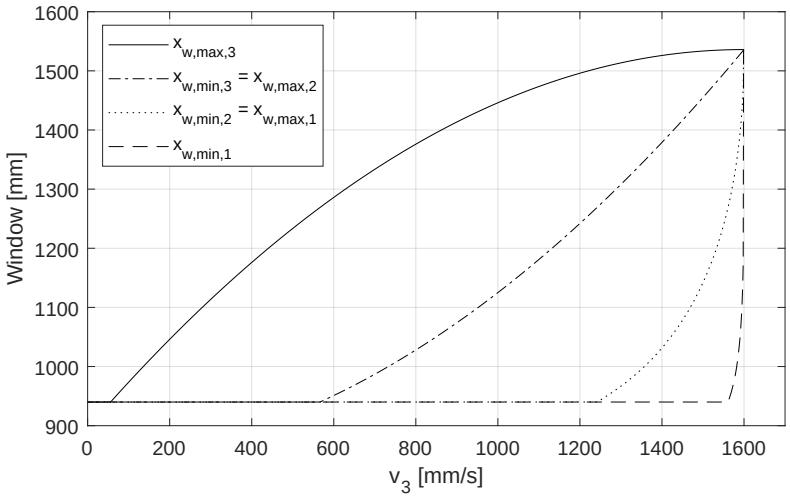

Figure 3.13: Window when using $L_{parcel} = 200$

Note that each window has a lower limit due to the acceleration from standstill constraint. For example, if $v_3 = 1000$ mm/s is selected to determine v_2 and v_1, then the minimum windows using v_2 and v_1 and the maximum window using v_1 are equal to 940 mm. If these windows were calculated using Equation 3.4 and Equation 3.3, they would have been lower than 940 mm. The equation used to determine this value, where x_d and x_a were determined using Equation 2.11, is as follows:

$$x_{w,constraint} = x_d + x_a + 100 + L_{parcel} = 940$$

The total window range has been plotted and is shown in Figure 3.14. The maximum total window range, which is marked with a circle, can be achieved by using the corresponding value of v_3, which in this case would be 1560 mm/s. The values that were found for v_2 and v_1 are 1237 and 561 mm/s respectively.

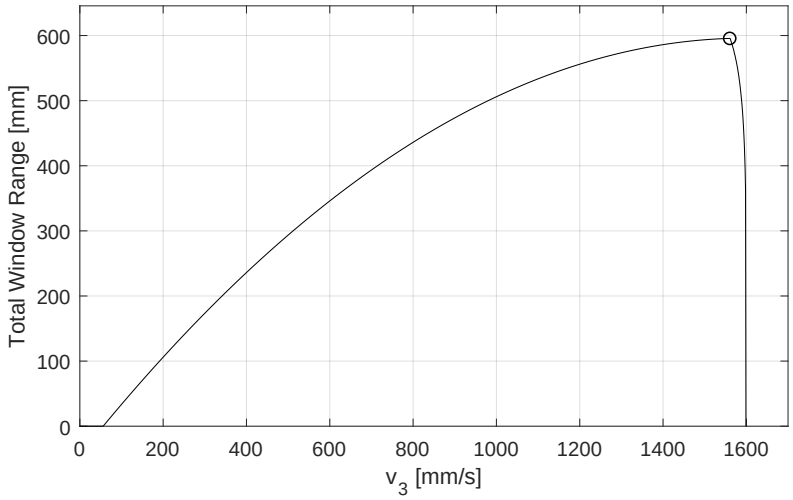

Figure 3.14: Total Window Range

Gaps Between Individual Window Correction Ranges

The values for the lower velocities v_1, v_2 and v_3 can then be used to check whether there are no gaps in between the window correction ranges for all other parcel lengths. If there is a gap in between these ranges, this section of the total window cannot be utilized to correct inaccurate windows. This check can be performed by subtracting the minimum window of lower velocity v_m from the maximum window of v_{m-1} where $m = \{2, 3\}$.

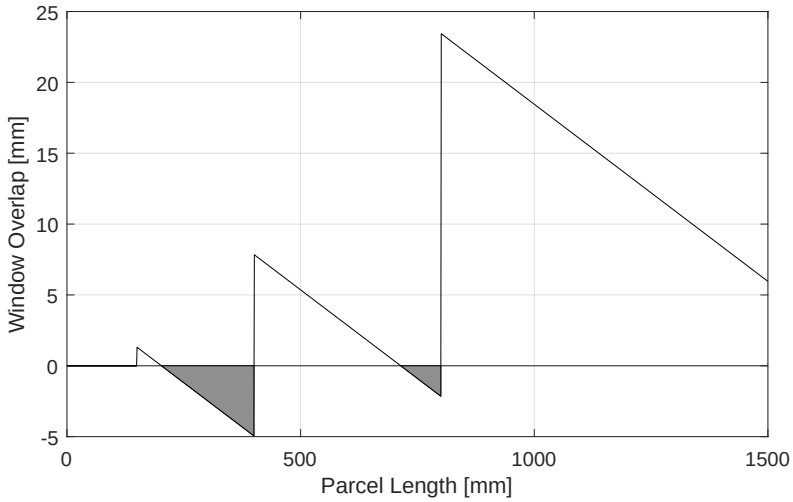

Figure 3.15: Minimum window v_3 - maximum window of v_2 when using $L_{parcel} = 200$

Figure 3.15 and Figure 3.16 show the window overlap for each parcel length in case the velocities are found using a parcel length of 200 mm. These figures show that at a parcel length of 200 mm, the window overlap is equal to 0 mm. The areas that are marked grey show where the overlap is negative, meaning that there is a gap between one or two window ranges. In both cases, parcel lengths larger than 200 mm and smaller than or equal to 400 mm have window range gaps. The same holds for parcel lengths larger than 713 mm and smaller than or equal to 800 mm. Therefore, it can be concluded that 200 mm is not the ideal initial parcel length that should be used to determine the set of lower velocities

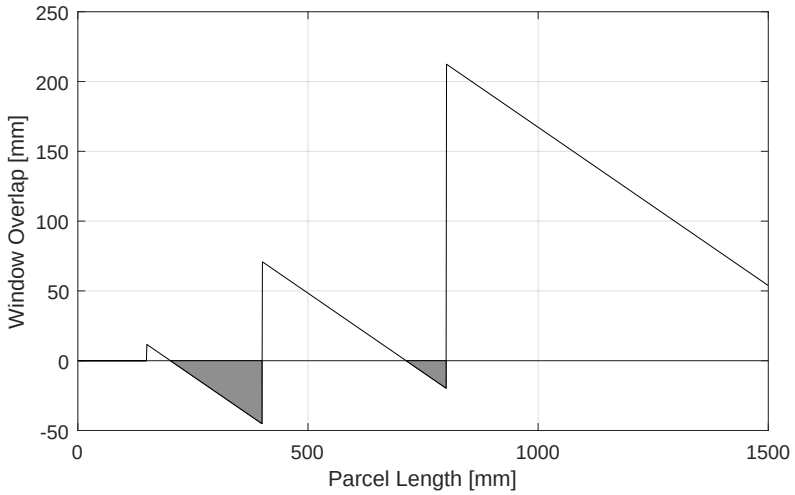

Figure 3.16: Minimum window v_2 - maximum window of v_1 when using $L_{parcel} = 200$

Ideally, the velocities are such that there are no window range gaps for any parcel length, so a different initial parcel length should be used to find adequate velocities. Both figures show that at parcel lengths 200 mm and 714 mm, the window overlap intersects with the parcel length axis and becomes negative. The intersection at 200 mm is a result of choosing a parcel length of 200 mm to find the velocities. Since the lowest minima in these figures are at 400 mm, having the intersection at this point would result in there being no window gaps for any parcel length. The results of using this parcel length can be seen in Figure 3.17 and Figure 3.18, which show that for all parcel lengths there are no gaps in between the window ranges.

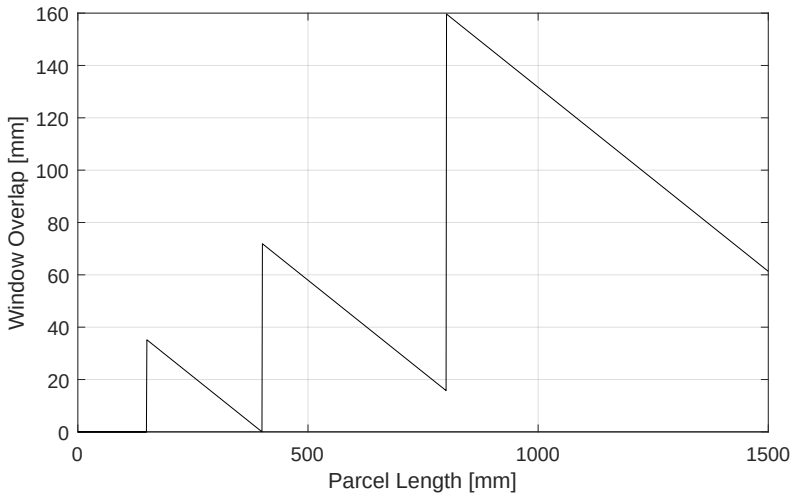

Figure 3.17: Minimum window v_3 - maximum window of v_2 when using $L_{parcel} = 400$

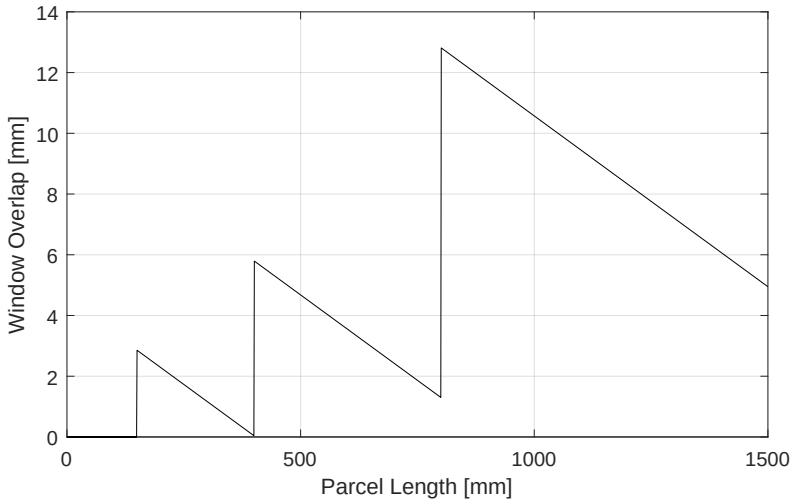

Figure 3.18: Minimum window v_2 - maximum window v_1 when using $L_{parcel} = 400$

Total Window Correction Range

As explained in the introduction of this section, the buffer should be able to correct the windows of incoming parcels with an accuracy of ± 100 mm. The total window range indicates whether this correction is possible with lower velocities. Figure 3.19 shows the total window range of all parcels in case the 1500 crossorter is considered.

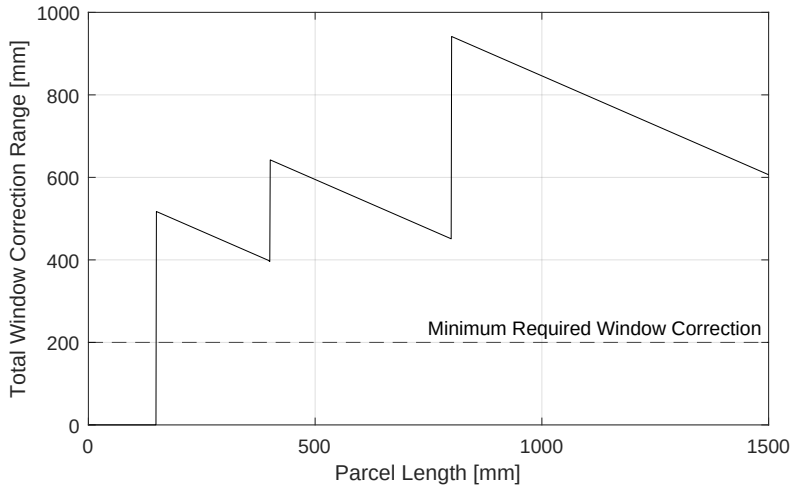

Figure 3.19: Total Window Range per parcel length when using $L_{parcel} = 400$

The results of a similar analysis that was performed for the 1200 and 1200HC crossorter are shown in Figure 3.20. Instead of using 400 mm, a parcel length of 300 mm was used to determine the velocities, because this is the point at which the window range is smallest. The 1200HC seems to have the least amount of total window range, which is due to it having the smallest carrier belt width. This figure also shows that for several parcel lengths the total window correction range exceeds the minimum required window correction.

These results indicate that the 1200 crossorter is not able to correct the error for each parcel length. This could be fixed by using additional lower velocities, extending the window correction range, or by reducing the window error ε_w.

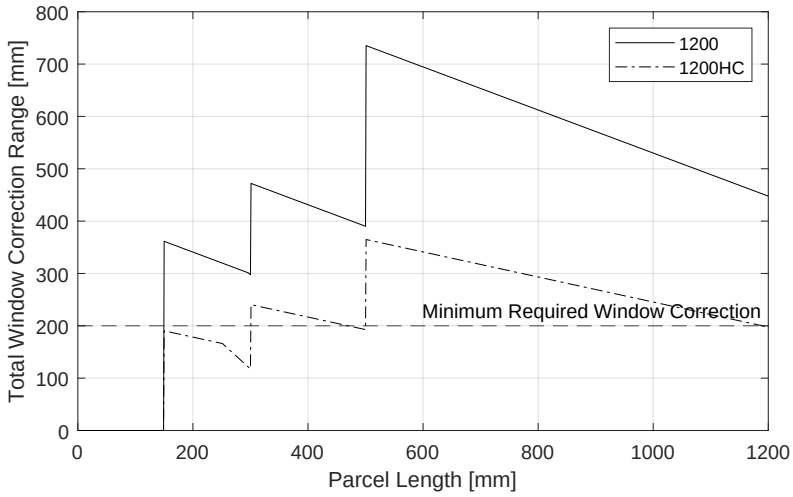

Figure 3.20: Total Window Range per parcel length when using $L_{parcel} = 300$

Optimal Lower Velocities per Sorter Type

From a parcel length of around 250 mm, the total window of the 1200HC starts decreasing faster. This is due to the minimum window using v_1 reaching the acceleration from standstill constraint, such as shown in Figure 3.13. From this point, the minimum window using v_1 remains at a constant value while the maximum window using v_3 keeps decreasing. The optimal velocities that were found as a result of using parcel lengths of 400 mm and 300 mm for the 1500 and 1200(HC) crossorters respectively are shown in Table 3.1.

Table 3.1: Optimal Velocities

	Lower Velocity [mm/s]		
	1200HC	1200	1500
v_3	1498	1490	1582
v_2	1436	1341	1375
v_1	1141	883	833

3.2.3 Adjusting Windows based on a Planned Sequence

The window correction ranges from the previous subsection are sufficient for correcting the error $\varepsilon_w = \pm 100$ mm for the 1500 and 1200 crossorter types. The 1200HC crossorter would require additional lower velocities or a smaller window error. Even though lower velocities can correct the window error ε_w, the question remains whether there it is also possible to change existing windows according to a planned sequence.

The amount of window space that needs to be adjusted depends on the difference between the initial window $x_{w,n} - 100 + \varepsilon_w$ and the window required to achieve the planned carrier location $x_{w,n+\Delta n}$ where Δn is the difference between the planned n on the main loop and the n from the windows before arriving at the buffer. The buffer should be able to adjust the window for any Δn, either by using lower velocities or by using acceleration from standstill. This is schematically depicted in Figure 3.21, which shows that the window adjustment can start as soon as the first of two arriving parcels has been handed over to the acceleration zone.

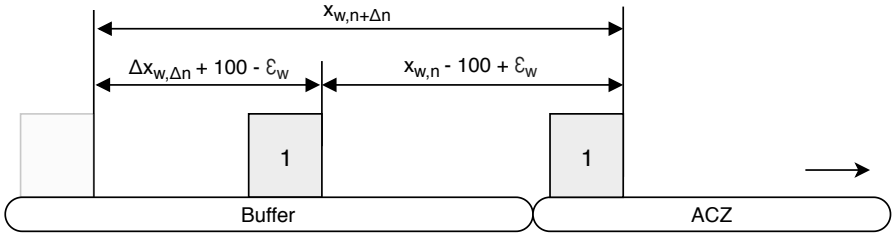

Figure 3.21: Two parcels heading to the buffer of infeed zone 1 with a window based on $x_{w,3}$

The terms in Figure 3.21 are defined as follows:

$$\Delta x_{w,\Delta n} = x_{w,n+\Delta n} - x_{w,n} \tag{3.5}$$

$$x_{w,n} = v_4 n t_{CT,n} \tag{3.6}$$

$$x_{w,n+\Delta n} = v_4 t_{CT,n+\Delta n} \tag{3.7}$$

where:

n	= Window expressed as n as supplied by the preceding trajectory
Δn	= Difference between supplied and required n
$x_{w,n}$	= Window size corresponding with n
$x_{w,n+\Delta n}$	= Window size corresponding with $n + \Delta n$
$\Delta x_{w,\Delta n}$	= Required window adjustment corresponding with Δn
v_4	= Acceptance Velocity of the First acceleration zone belt
$t_{CT,n}$	= Cycle Time from Table 2.3 corresponding with n
$t_{CT,n+\Delta n}$	= Cycle Time from Table 2.3 corresponding with $n + \Delta n$

The required window adjustment per value of Δn and crossorter type has been determined using these equations and is shown in Table 3.2. Since the ε_w ranges between -100 and 100 mm, the worst case would be when ε_w = -100 mm, resulting in the initial window being $x_{w,n} - 200$. In this case, the initial window is as small as possible, while the required window is as large as possible for given Δn. This means that a relatively large adjustment is required while there is less space in the initial window to accomplish that adjustment. If these window adjustments can be accomplished, it should also be possible to adjust the windows for different values for ε_w.

Table 3.2: Required window adjustment per Δn where ε_w = -100 mm

	$\Delta x_{w,\Delta n} + 100 - \varepsilon_w$		
Δn	1200HC	1200	1500
1	560	620	712
2	920	1040	1224
3	1280	1460	1736
4	1640	1880	2248
5	2000	2300	2760
6	2360	2720	3272

Window Adjustment Feasibility Using Lower Velocities

Since the total window correction range has already been determined in subsection 3.2.2, the feasibility of the window adjustments from Table 3.2 can be determined by checking if the total window correction range exceeds the required window adjustment. Figure 3.22 and Figure 3.23 show the required adjustments in case $\Delta n = 1$ plotted with the total window correction range per parcel length. These figures indicate that using lower velocities provides insufficient window adjustment capability for the majority of parcel lengths. The required adjustment in case $\Delta n = 2$ even exceeds the horizontal axis for all crossorter types, so it can be concluded that an alternative method is necessary.

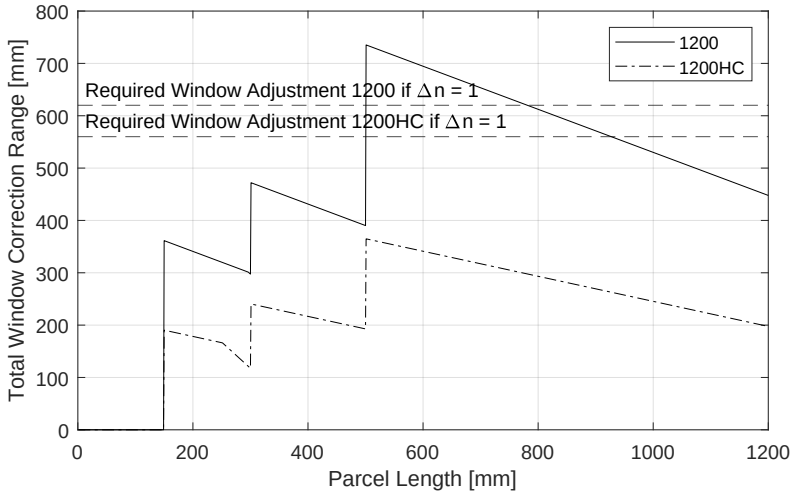

Figure 3.22: Required Window Adjustment vs Window Correction Range 1200(HC)

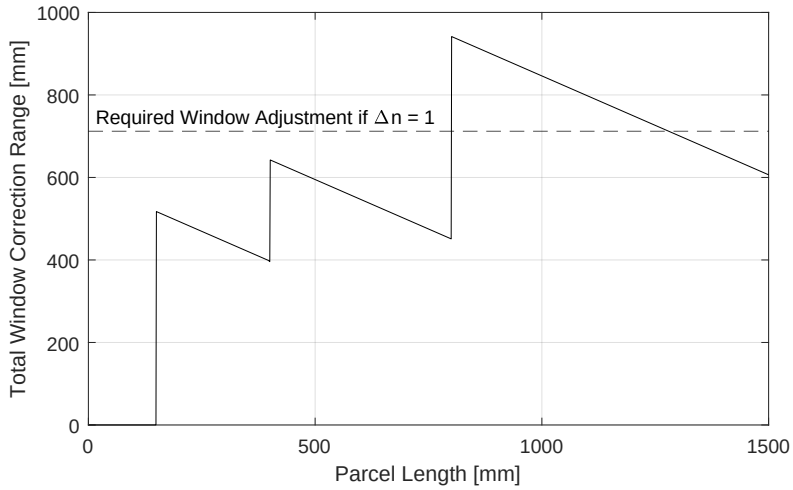

Figure 3.23: Required Window Adjustment vs Window Correction Range 1500

Window Correction feasibility using Acceleration from Standstill

An additional method is to use acceleration from standstill. The window adjustment capabilities were determined using the lowest velocity with the maximum amount of time spent at a lower velocity. Smaller adjustments are possible, but not larger ones. When using acceleration from standstill, the minimum window adjustment can be achieved by decelerating and accelerating again without waiting.

Any larger adjustment can be achieved simply by waiting, but a smaller adjustment is not possible. Therefore, it makes sense to determine the required window adjustment per Δn in case $\varepsilon_w = 100$ mm. In this case, the required window adjustment is as small as possible. If the adjustment can be achieved by using acceleration from standstill, it can also be achieved for all other values of ε_w.

Table 3.3: Required window adjustment per Δn where $\varepsilon_w = 100$ mm

	$\Delta x_{w,\Delta n} + 100 - \varepsilon_w$		
Δn	1200HC	1200	1500
1	360	420	512
2	720	840	1024
3	1080	1260	1536
4	1440	1680	2048
5	1800	2100	2560
6	2160	2520	3072

The minimum window adjustment is equal to $x_d + x_a$ because this would be the distance needed to decelerate and accelerate again. A larger window correction can be achieved by waiting after having decelerated before accelerating again, so the minimum window correction is equal $x_d + x_a$, but it can be enlarged simply by waiting. Table 3.4 shows an overview of the minimum correction windows per crossorter type.

Table 3.4: Minimum window adjustment using acceleration from standstill

	1200HC	1200	1500
$x_d + x_a$ [mm]	563	563	640

Comparing Table 3.4 to Table 3.3 shows that $\Delta x_{w,1} < x_d + x_a$ for all crossorter types. This means that if the required window corrections $\Delta x_{w,1}$, is smaller than the minimum correction window, meaning it is not feasible to correct the window. Using acceleration from standstill is only a suitable solution for gap correction if $\Delta n > 1$.

3.3 Discussion

Two methods for buffering parcels have been investigated in this chapter. The first method required each arriving parcel to stand still and wait, which would result in cycle times that are higher than the succeeding zones. This would create a bottleneck in the parcel flow, and would therefore not be a good option.

A second option would be to use lower velocities instead of standing still. This way, the cycle time could be maintained by correcting the windows between parcels. A requirement with this method is that the windows of arriving parcels must be controlled to be 100 mm smaller than their required window because of the two-sided ±100 error. By controlling the windows in such a way, they should vary between 0 and 200 mm smaller than the correct window. The motors in current crossorter systems allow for three additional velocities, which would be sufficient for the 1500 and 1200 crossorter, but not for the 1200HC crossorter because it is not able to correct windows by 200 mm with three lower velocities.

Windows may also need to be corrected if windows are supplied according to parcel length, but a planned sequence on the main loop requires a different window. It turned out that if the required correction is $\Delta n = 1$, it may not be feasible because the window correction with lower velocities

would be too small, and using acceleration from standstill would be too large. A solution would be to use motors that can use more than three velocities of which the additional velocities would be even lower such that larger windows can be created.

Even if it is possible to use different motors, it doesn't indicate whether or not a system with buffers would increase the utilization of current systems. The utilization as a result of using a planned sequence has therefore been determined and compared against the results from using Vanderlande emulation with the same windows between the parcels as they would have been supplied to the buffer. The next chapter contains the results of the comparison accompanied by an analysis and a discussion.

Chapter 4

Results and Analysis

In this chapter, the results of comparing a system with buffers and the current system are presented and discussed. The aim is to show whether or not the proposed solutions improve utilization. Vanderlande's emulation environment has been used to generate results that represent the current system. This environment uses the same control hardware and software where all other hardware, such as conveyor belts, are simulated. It is possible to provide the emulation system with input data consisting of a specific set of parcel dimensions, arrival order and window size per parcel. The same set of input data can be used to calculate the performance of the system with buffers, allowing for a comparison between the current and the proposed system.

4.1 Experimental Design

Only one crossorter type is considered in this chapter, which is the 1500 crossorter. The reason for this is because at the time of starting the emulation experiments, the only properly working crossorter model was of the 1500 crossorter. Furthermore, since all crossorters function similarly, the results from this chapter are likely to apply to any crossorter.

Rather than using only one parcel length, a parcel distribution has been used to create a random arrival of parcels, similar to how parcels would be supplied to a real system. The distribution that has been used is shown in Figure 4.1. In total, a set of 9670 parcels was considered, which is the amount needed to have at least one of each parcel length. The parcels used in this chapter all had widths that were 2/3 times the length. This is commonly used because many packages have this length to width ratio.

The utilization ρ is defined as the number of carrier belts that are utilized divided by the total amount of carrier belts, which is the sum of the utilized carrier belts N_u and the empty carrier belts N_e in between them. A carrier belt is considered to be utilized if it holds one narrow parcel or if it is one of the three carrier belts used for triple belt reservation.

$$\rho = \frac{N_u}{N_u + N_e} \tag{4.1}$$

The utilization may change over time, so the emulation data used to calculate the utilization is divided into multiple chronological sections. For each section, a value of ρ is determined and compared to the corresponding section from the proposed system with buffers.

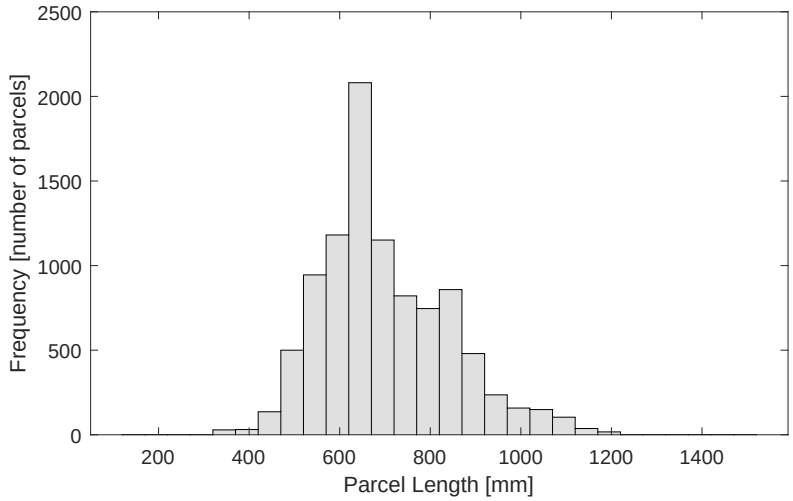

Figure 4.1: Parcel Distribution obtained from data from a 1500 crossorter

4.2 Initial Results

An emulation run was conducted with the parcel distribution shown in Figure 4.1. The arrival order of the parcels was randomized and sequentially divided into three separate sets corresponding with three separate infeed zones. The log files obtained from the emulation run show which parcel has been inducted on which carrier. If a carrier is not used, the number is not included in the log file. With this information, the utilization can be determined.

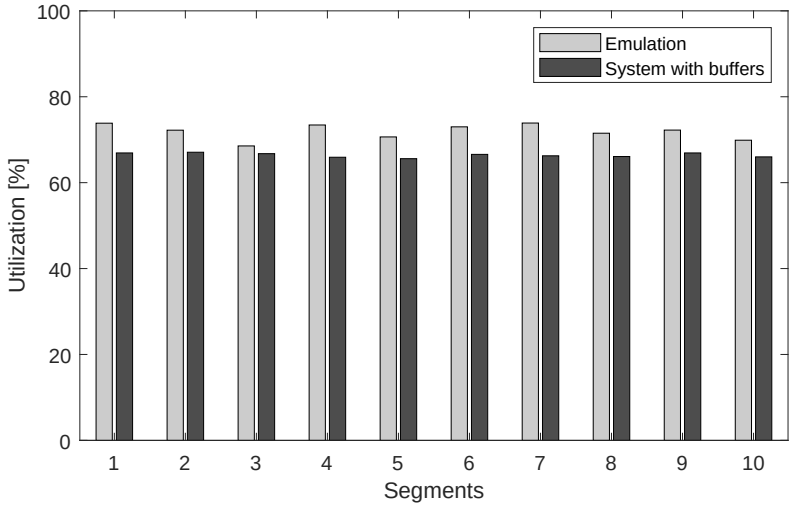

Figure 4.2: Utilization values from the initial test run

The utilization using a system with buffers was determined using an algorithm in Matlab that enters the parcel ID numbers into an empty matrix representing the main loop. The algorithm loops through the empty matrix and finds the next empty cell that can be used by an infeed while adhering to the infeed zone cycle time constraints from Table 2.1 and the main loop cycle time constraints from Table 2.3. The first parcel was inducted from infeed zone 1, the second from infeed zone 2, and the third from infeed zone 3. This round-robin pattern was maintained throughout the main loop to assure the throughput balance between the infeed zones.

The results of the emulation run and the system with buffers are shown in Figure 4.2. The average utilization of the emulation run ρ_e is 71.92% with a standard deviation of 1.78%, while the system with buffers has an average utilization of ρ_b 66.39% with a standard deviation of 0.50%. The utilization of the system with buffers is lower than the emulation results. It turned out that the infeed zone cycle time constraints taken from Table 2.1 do not reflect the highest possible capacity. According to Section 2.1 and regarding the 1500 crossorter, if $800 \leq L_{parcel} \leq 1500$ then $n = 6$, which means that every 6^{th} carrier can be utilized, corresponding with a cycle time of 1.96 s in Table 2.1. This suggests that a buffer should release a parcel every 1.96 s if only parcels in this length range are released.

4.2.1 Experiments with Lower Values of n

What happens in reality, however, is that sometimes a lower cycle time can be used, but eventually an arriving parcel is forced to use the cycle time from Table 2.1. An emulation experiment was conducted to determine why this occurs. First, the manual input option for emulation was used to visually determine if lowering the cycle time would consistently change the value of n. The parcel dimensions in Table 4.1 showed this behavior, so an emulation run was performed with 500 parcels per parcel length on one infeed. The value of n corresponding with the window that was supplied is shown at the top of Table 4.1. The percentages show how often a parcel could utilize every 3^{rd}, 4^{th}, 5^{th} or 6^{th} carrier belt.

Table 4.1: Test set for determining lower cycle times

		supplied $n = 3$		supplied $n = 4$		supplied $n = 5$	
Length	Width	$n = 3$	$n = 4$	$n = 4$	$n = 5$	$n = 5$	$n = 6$
400	267	73.1%	26.9%	-	-	-	-
450	300	61.0%	39.0%	-	-	-	-
500	333	51.8%	48.2%	-	-	-	-
550	367	46.8%	53.2%	-	-	-	-
600	400	41.0%	59.0%	-	-	-	-
650	433	35.0%	65.0%	-	-	-	-
800	533	0.2%	99.8%	-	-	-	-
850	567	-	-	90.4%	9.6%	-	-
900	600	-	-	71.7%	28.3%	-	-
900	600	-	-	-	-	98.8%	1.2%
950	633	-	-	-	-	91.4%	8.6%
1000	667	-	-	-	-	97.4%	2.6%
1050	700	-	-	-	-	90.0%	10.0%
1100	733	-	-	-	-	88.4%	11.6%
1150	767	-	-	-	-	75.6%	24.4%
1200	800	-	-	-	-	60.4%	39.6%
1250	833	-	-	-	-	52.9%	47.1%

The reason why the supplied n does not equal the n on the main loop is because the relative moment of acceleration on the infeed zone changes with each arriving parcel. Before a parcel is inducted onto the main loop, a moment of acceleration is determined that dictates where exactly a parcel will be inducted on the main loop. The feasible acceleration moments lie between accelerating early and late, which corresponds with t_1 and t_2 shown in Figure 4.3 and Figure 4.4.

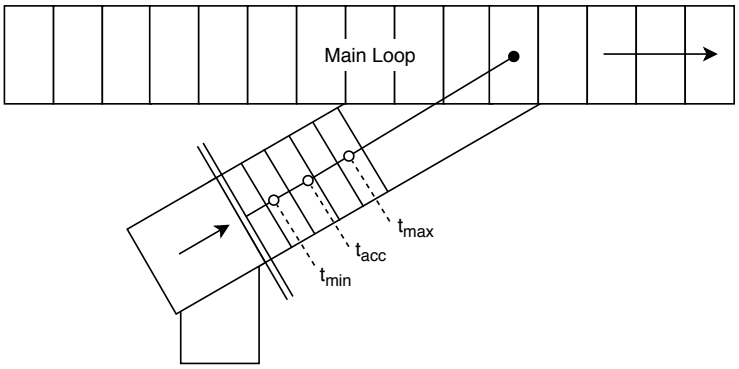

Figure 4.3: Schematic depiction of the infeed zone

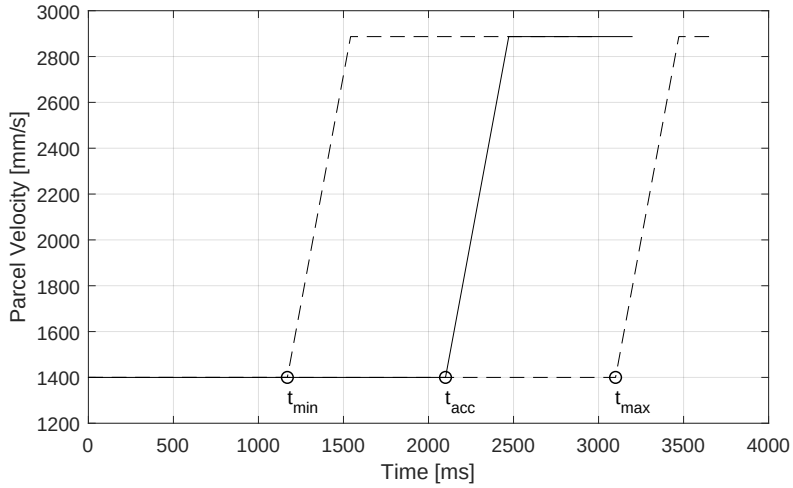

Figure 4.4: Parcel Acceleration Range

In the emulation experiments that were conducted, parcels were supplied at fixed windows, which were calculated based on the crossorter speed. The windows are therefore expected to be perfectly synchronized with the main loop. For some combinations of parcel lengths and n, the slight imperfection causes the moment when a parcel accelerates t_{acc} to gradually shift from t_{max} to t_{min}. Once t_{acc} comes close to t_{min}, the moment of acceleration will jump to a value close to t_{max} and start gradually shifting again towards t_{min}.

This effect is visible in Figure 4.5, which shows that the relative acceleration moment $t_{min} - t_{acc}$ decreases several times after which it immediately shifts back to a higher value. The value of n is also shown in this figure, which shows that after each time $t_{min} - t_{acc}$ is at it's lowest point a parcel is inducted with $n = 6$ instead of $n = 5$. Because this is something that is generated while a crossorter is operating, it should be feasible to build a control system that adjusts the windows between parcels arriving at the infeed zone based on the value of $t_{min} - t_{acc}$.

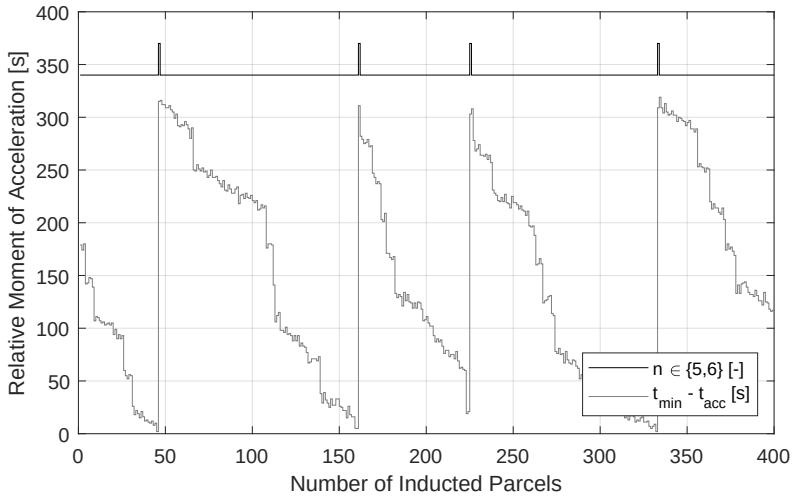

Figure 4.5: $L_{parcel} = 900$ and $n = 5$

4.3 Main loop Utilization With Lower Values of n

If a controller is made that can adjust the parcel windows based on the relative acceleration moment, some parcels could be inducted with lower values of n. In current systems, the values are as follows:

$$\begin{aligned} n = 3 \quad &\text{for} \quad 150 \leq L_{parcel} \leq 400 \\ n = 4 \quad &\text{for} \quad 400 < L_{parcel} \leq 800 \\ n = 6 \quad &\text{for} \quad 800 < L_{parcel} \leq 1500 \end{aligned}$$

Similar figures to Figure 4.5 have been made for each test scenario in Table 4.1, which are shown in Appendix B. In some of these figures, the relative moment of acceleration seems to change more erratically than shown in Figure 4.5 for example. When considering the figures which show a clear gradual decline of $t_{min} - t_{acc}$ a new set parcel length ranges can be determined per value of n.

$$\begin{aligned} n = 3 \quad &\text{for} \quad 150 \leq L_{parcel} \leq 400 \\ n = 4 \quad &\text{for} \quad 400 < L_{parcel} \leq 850 \\ n = 5 \quad &\text{for} \quad 850 < L_{parcel} \leq 1050 \\ n = 6 \quad &\text{for} \quad 1050 < L_{parcel} \leq 1500 \end{aligned}$$

The main loop utilization with these alternative length ranges and values for n can be seen in Figure 4.6. The utilization $\rho_e = 71.92\%$ with a standard deviation of 1.78%, while $\rho_b = 73.71\%$ with a standard deviation of 0.42%. Compared to the previously used values for n, it shows an increase in utilization.

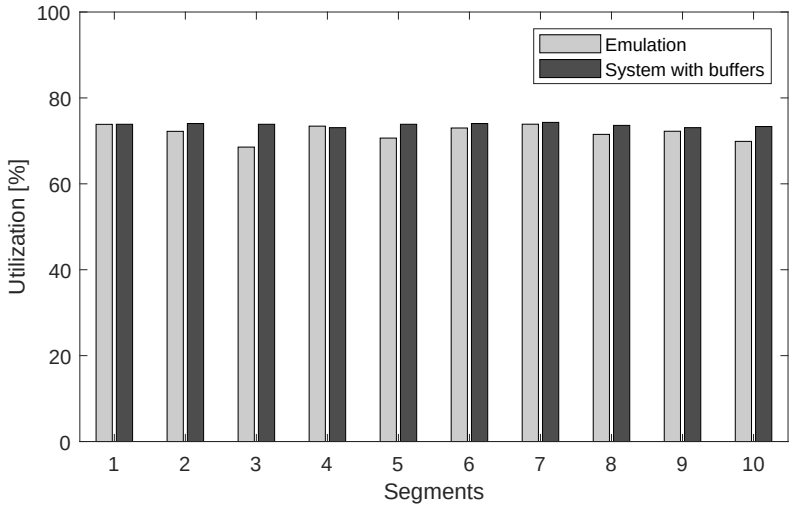

Figure 4.6: Utilization values with updated length ranges and values for n

4.3.1 Alternative Merge Algorithm

Even though the utilization is improved by using lower values of n, there turned out to be another option to increase the utilization even further. When taking a closer look at the matrix representing the main loop, it turned out that infeed zones were missing merge opportunities in some cases. Consider for example a sequence of parcels sequentially coming from infeed zones 1, 2, 3 and 1 where the first parcel from infeed zone 2 is a wide parcel, while the parcels from infeed zone 1 and 3 are narrow. The first parcel from infeed zone 2 causes a delay on infeed zone 2, which means that the next parcel must be inducted at $n = 6$. This leaves two empty carrier belts on the main loop, shown in Figure 4.7. The utilization, in this case, is lower than what is theoretically possible.

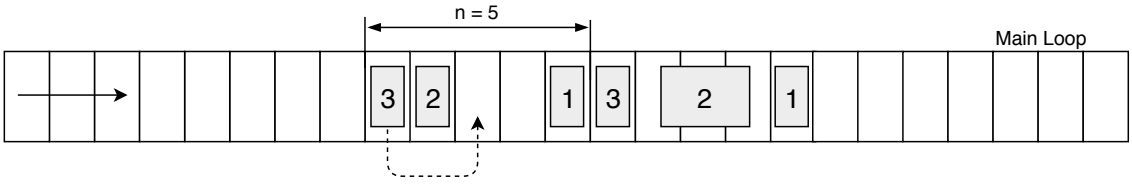

Figure 4.7: Schematic Depiction of the Main Loop using a Round-Robin Algorithm

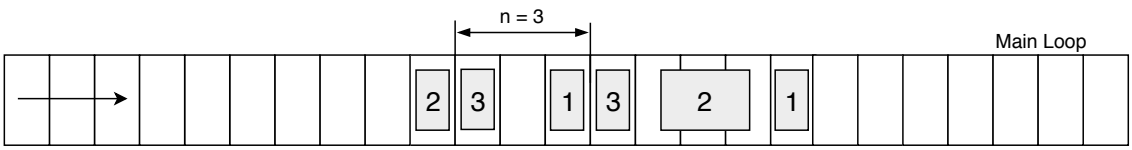

Figure 4.8: Schematic Depiction of the Main Loop using the Alternative Algorithm

The second parcel coming from the third infeed zone could have been inducted on the latter of the two empty carriers, as shown in Figure 4.8. This could be achieved by regarding parcels in all buffers,

rather than the one that follows a round-robin sequence. For example, after having allocated the second parcel from infeed zone 1, the parcels in the buffer of infeed zone 3 should be considered. The results of using this algorithm are shown in Figure 4.9. The utilization, in this case, is $\rho_b = 74.42\%$ with a standard deviation of 0.39%.

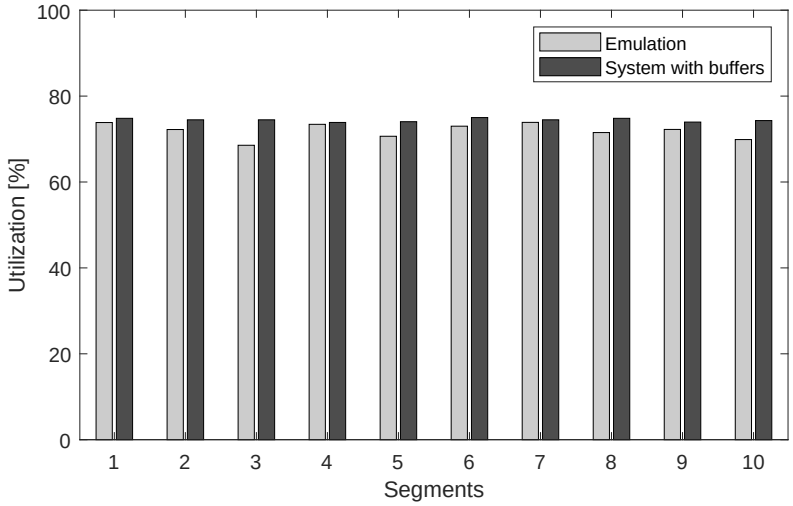

Figure 4.9: Utilization values with updated length ranges and values for n

4.4 Discussion

An overview of the results is shown in Table 4.2. Scenario 1 is using a Round-Robin algorithm, scenario 2 uses lower values of n and scenario 3 uses the alternative algorithm while also using lower values of n. It can be concluded the utilization of a system with lower values of n and the alternative algorithm outperforms the other scenarios and the results from emulation. The theoretical utilization results all have lower standard deviations, which is probably because these are determined under the assumption that blockages never occur. A decrease in standard deviation can also be seen as the theoretical mean increases. If the utilization would be 100%, then it would have to be 100% for all 10 segments, resulting in a standard deviation of 0%, so the decrease is probably because it approaches 100%.

Table 4.2: Overview of emulation results and theoretical utilization

	Utilization [%]	
	Mean	Std. Dev.
Emulation	71.92	1.78
Scenario 1	66.39	0.50
Scenario 2	73.71	0.42
Scenario 3	74.42	0.39

Chapter 5

Conclusions & Recommendations

Conclusions

This research aimed to identify how the utilization of crossorters can be increased while using three infeed zones. Current systems use more than three infeed zones to increase the utilization, resulting in a higher total cost compared to a system with three infeed zones. A system extension was proposed with buffers that allow parcels to be queued or slowed down to control the arrival at the infeed zones.

The trajectory that a parcel follows after being released from a buffer was analyzed by determining and comparing the cycle times of each zone. Two important findings were derived from this analysis. First, it turned out that the current acceleration zone design does not allow for parcel hand-over to the infeed zone without slippage. The second finding was that a planned release sequence from buffers on multiple infeed zones determines the required windows before parcels arrive at the acceleration zone. This is different than the current system which creates windows based on parcel length.

Following the analysis, two buffer designs were considered. The first design is one where each parcel must stand still before being accelerated. The second design uses lower velocities instead of standing still to create the required window. The first design would create a bottleneck in the flow because in many cases the cycle time is lower than that of the infeed zone. This design would be able to create windows based on when a parcel starts accelerating, while the second design requires the inaccuracy of windows supplied to the buffer to be corrected. The analysis of the second design showed that window corrections are not always feasible. If relatively large corrections are required, using acceleration from standstill would be sufficient for correcting the window, but if the required correction corresponds with $\Delta n = 1$, it is often not feasible.

This research may not have provided the full technical design needed to make buffers feasible, but it has investigated the requirements for using buffers and provides directions for future research. Regardless of this, it was still possible to determine the theoretical utilization of a system with buffers. The initial results of comparing the emulation model at Vanderlande with the theoretical utilization showed that the utilization of the current system was $\rho_e = 71.92\%$ with a standard deviation of 1.78%, while the proposed system had a utilization of $\rho_b = 66.39\%$ with a standard deviation of 0.50%. This was an unexpected result because the utilization is lower with buffers.

An important cause of the lower utilization was that the infeed zone constraints used to calculate the theoretical utilization did not reflect the constraints of real infeed zones. After running a set of emulation experiments, the results showed that lower values of n were possible, but that occasionally a higher value of n occurs. A gradual shift in the relative acceleration moment on the infeed zone's short belts causes the value of n to occasionally increase. This shift occurs when the windows of parcels arrive at the infeed are slightly too large compared to the sorter speed. If the windows of arriving parcels

can be controlled based on the relative acceleration moment, different parcel length ranges for lower values of n are feasible. Alternative parcel length ranges per value of n were determined based on the assumption that the windows of arriving parcels can be controlled by using the relative acceleration moment. In this case, the utilization would be $\rho_b = 73.71\%$ with a standard deviation of 0.42%.

Further investigation showed that planning a sequence of arrivals according to a round-robin algorithm is leaves empty spaces that could have been utilized by parcels waiting on other infeed zones. An alternative algorithm was made where the next parcel in the sequence was selected based on which one could be inducted the earliest. This resulted in a utilization of $\rho_b = 74.42\%$ with a standard deviation of 0.39%.

Recommendations

The theoretical utilization values are only valid under the assumption that there is no slippage once parcels are released from the buffer and that parcel blockages never occur on the infeed zones. In reality, this is often not the case, so further research may be needed to investigate whether these findings apply in situations with operational disturbances. This could be done by means of a simulation experiment in a model-based design environment that models the physical trajectory of parcels from the buffer to the main loop.

The analysis of the acceleration zone showed that the current design may cause parcels to slip during the hand-over between conveyor belts. In many cases, the cycle times of the acceleration zone's conveyor belts are too long. The amount of slippage that occurs on an acceleration zone at high levels of throughput should be measured on a real crossorter, since slippage is not present in an emulation environment. If the amount of slippage that is measured would affect the feasibility of using buffers, then the acceleration zone must be redesigned.

A possible redesign of the acceleration zone could be done by using shorter belts in the acceleration zone. This would decrease the cycle time and therefore allow for hand-over without slippage. However, if the belt lengths become shorter than the length required to accelerate large parcels, then slippage might still occur. A different control algorithm would have to be designed concurrently that allows larger parcels to accelerate using multiple belts such that slippage will not occur.

The second buffer design indicates that it is not possible to sufficiently change the window size. It should be investigated whether a different type of motor can be used that allows for even lower velocities. The required correction for $\Delta n = 1$ should be used to determine which lower velocity is needed.

Further research should be done concerning the feasibility of a control system that uses the relative moment of acceleration to increase main loop utilization, regardless of using buffers. The effect of this can be seen in current systems, which implies that it may be possible to improve the utilization without adding any hardware.

Bibliography

J. Meens. *Model-Based Design Approach for Merge Balancing*. Master's thesis, Eindhoven University of Technology, the Netherlands, 2017. 3, 4, 8

K. Peeters. *Balancing Control of Material Handling Systems*. Master's thesis, Eindhoven University of Technology, the Netherlands, 2015. 3, 4, 8

M. Struik. *Design Optimisation of a Throw-Catch Infeed Using a Model-Based Design Approach*. Master's thesis, Eindhoven University of Technology, the Netherlands, 2018. 3, 5, 6, 8

Appendix A

Buffer Cycle Time Tables

This appendix contains confidential information on the Vanderlande throw-catch infeed. It is only available at Vanderlande and has not been made public.

Appendix B

Relative Acceleration Graphs

This appendix contains confidential information on the Vanderlande throw-catch infeed. It is only available at Vanderlande and has not been made public.