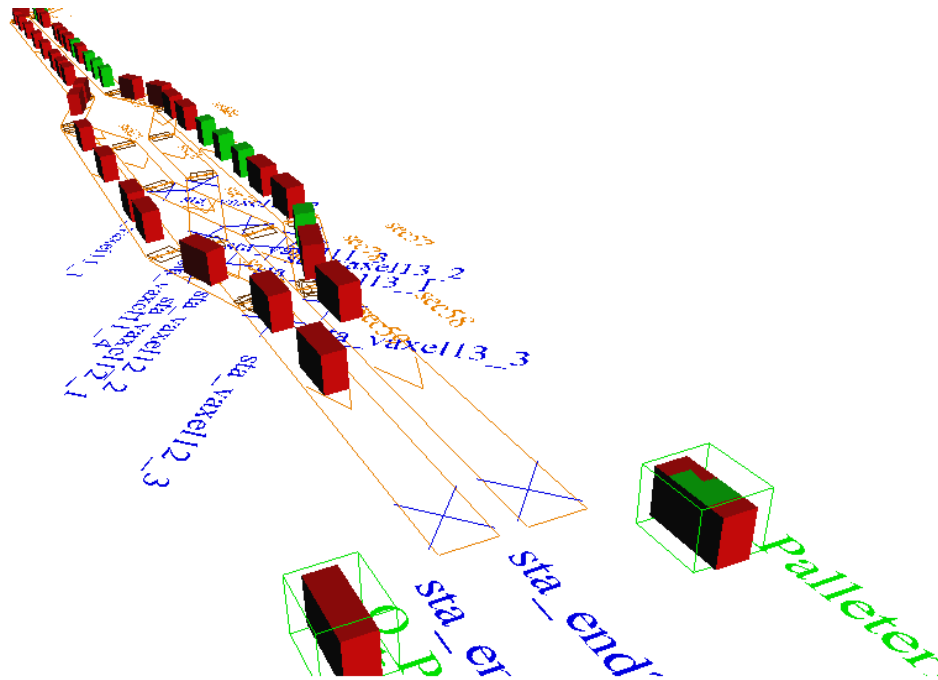




CHALMERS



Simuleringsprojekt för transportbandssystem

Examensarbete inom högskoleingenjörsprogrammet Maskinteknik

ERIK LILJA

MORGAN OLSSON

Institutionen för Industri- och Materialvetenskap (IMS)

CHALMERS TEKNISKA HÖGSKOLA

Examensarbete IMSX20

Göteborg, 2019

EXAMENSARBETE IMSX20

Simuleringsprojekt för transportbandssystem

*Examensarbete i högskoleingenjörsprogrammet
Maskinteknik*

ERIK LILJA
MORGAN OLSSON

Institutionen för Industri- och Materialvetenskap
CHALMERS TEKNISKA HÖGSKOLA
Göteborg, 2019

Simuleringsprojekt för transportbandssystem

Examensarbete inom högskoleingenjörsprogrammet Maskinteknik

ERIK LILJA

MORGAN OLSSON

© ERIK LILJA, 2019

© MORGAN OLSSON, 2019

Examinator: Björn Johansson, Industri- och Materialvetenskap

Handledare: Camilla Lundgren, Industri- och Materialvetenskap

Examensarbete 2019 Institutionen för Industri- och Materialvetenskap

Chalmers tekniska högskola

SE-412 96 Göteborg

Telefonnummer +46 31 772 1000

Förstasida:

Överblick på transportbandssystemet i Automod.

[Egen bild]

Göteborg, Sweden 2019

Simulation Project for a Conveyor System

*Bachelor's thesis in the Engineering programme
Mechanical Engineering*

ERIK LILJA

MORGAN OLSSON

Department of Industry and Material science
Chalmers University of Technology

Abstract

This project includes simulation and analyzation of a conveyor system in companies' factory. The company is in a stage where they need to improve their overall efficiency, since they have a lot of losses connected to production failure. The situation today is that the factory is running on a low percentage of efficiency, and thereby has to be running every day of the week, including weekends, to achieve their orders.

The company is aware of the problems concerning the efficiency on the palletizing systems that comes after the conveyor systems, these problems will be sorted later. Since the palletizing problem is known and will be sorted, it's crucial to know whether the conveyor system is able to keep up with its surrounding systems or if it will be a bottleneck. This is not possible to test in the real factory before the implementation of the new palletizing station. That's why simulation comes in handy, since it can test the conveyor system without the need to disturb or wait for the new palletizing system.

The project aimed to replicate the reality in a simulation, using the available in-parameters, to test how the conveyor system will behave with the new palletizing system. The simulation can also be used to test the conveyor system in different situations.

While creating the simulation the group used Banks method in structuring the project. Banks uses a flowchart to visualize the events and their order in the project. Banks includes 11 steps which were followed during the project.

For the simulation to be legitimate, all the essential information about the conveyor system had to be obtained. The surrounding influencing factors could be traced down to four sources, which affected different parts of the system. By inserting the obtained information in the simulation, it reached a 5 percent margin of error.

In conclusion, there are some areas in simulation that is more difficult than others. Primarily it is the collecting of essential data that is the biggest challenge. It is also of big importance to have foundational knowledge about Automod coding for the project to move along smoothly.

Simuleringsprojekt för transportbandssystem

*Examensarbete i högskoleingenjörsprogrammet
Maskinteknik*

ERIK LILJA

MORGAN OLSSON

Institutionen för Industri- och Materialvetenskap
Chalmers tekniska högskola

SAMMANFATTNING

I projektet ingår simulering och analysering av ett transportsystem i företagets fabrik. Företaget befinner sig just nu i ett stadium där de behöver förbättra sin produktionseffektivitet, eftersom de har stora förluster kopplade till produktionsstopp.

Då företaget är medveten om problematiken som orsakas av palleteringen så kommer detta åtgärdas. Eftersom problemen med palleteringen förväntas försvinna är det av företagets intresse att undersöka om det eventuellt är transportbandssystemen som kommer bli den nya flaskhalsen. Med hjälp av en simulering kan effektiviteten testas utan att störa produktionen och behöver inte heller vänta tills den nya palleteringen installerats.

Projektet syftar på att göra en simuleringsmodell av det verkliga transportbandssystemet, vilken är ihopkopplad med dess omkringliggande påverkande faktorer. Simuleringen kommer inte bara kunna testa hur transportbandssystemet fungerar ihop med den nya palleteringen. Den kommer även fungera som ett verktyg som företaget kan använda för att testa transportbandet i andra typer av situationer.

Transportsystemet innehåller två transportbandslinjer som är kopplade till växlar, banden transporterar laster från 11 olika förpackningsstationer. Projektet syftar till att replikera verkligheten i en simulering med hjälp av de in-parametrar som är tillgängliga från företaget.

När simuleringen skapades användes Banks metod för att strukturera projektet. Banks metod innebär att arbetet följer ett förutbestämt flödesschema för att visualisera händelserna och deras respektive ordning i projektet.

För att simuleringen ska vara legitim måste all essentiell information om transportbandssystemet finnas till hands. De omgivande påverkande faktorerna kunde spåras till fyra faktorer, som påverkade olika delar av systemet. Genom att införa den insamlade informationen i simuleringsmodellen uppnåddes en felmarginal på 5,1 procent.

Sammanfattningsvis kan det ses att vissa områden inom simulering är mer besvärliga än andra. Framförallt är det datainsamlingen som är problematisk. Dessutom kommer grundläggande kunskaper inom Automod programmering krävas, för att arbetet ska fortgå smidigt.

FÖRORD

Denna projektrapport är vårt examensarbete på högskoleingenjörsprogrammet i maskinteknik på Chalmers tekniska högskola i Göteborg. Arbetet är utfört av Morgan Olsson och Erik Lilja under vårterminen 2019, dess omfattning är 15 högskolepoäng.

Vi har under arbetets gång haft stor nytta av en mängd människor som är insatta i något av de områden som projektet omfattat. Framför allt har vår handledare på konsultföretaget bidragit mycket till detta projektet. handledaren arbetar inom simulering och är därmed mycket kunnig inom området. Handledarens kollega, arbetar också inom simulering på konsultbolaget och har hjälpt oss med en mängd tips under simuleringsarbetets gång.

Vi vill även tacka de anställda på företaget som hjälpt oss att få fram parametrar och övrig information om transportbandssystemet som simulerats. Framförallt är det vår kontaktperson på företaget som varit mycket behjälpliga från företagets håll.

På Chalmers är det Camilla Lundgren, vår handledare, och Björn Johansson, vår examinator, som hjälpt oss med handledning av projektet, framförallt med projektrapportens struktur och innehåll.

Innehåll

1. INLEDNING	1
1.1 Bakgrund	1
1.2 Syfte och mål	1
1.3 Frågeställning	2
1.4 Avgränsningar	2
2. TEORETISK REFERENSRAM	4
2.1 Simulering	4
2.1.1 Problemformulering & projektplan	5
2.1.2 Datainsamling	5
2.1.3 Bygga modell	6
2.1.4 Verifiering	6
2.1.5 Validering	7
2.1.6 Experimentdesign & analys	7
2.1.7 Dokumentera Resultat och implementering	7
2.2 Github	8
3. METOD	9
3.1 Problemformulering & projektplan	9
3.2 Datainsamling	9
3.3 Bygga modell	11
3.4 Verifiering	12
3.5 Validering	13
3.6 Experiment design och analys	13
3.7 Dokumentera resultat och implementering	14
3.8 Github	14
4. RESULTAT	15
4.1 Nulägesanalys	15
4.1.1 Systemets funktionalitet	15
4.1.2 Packmaskinerna	16
4.1.3 Transportbanden	17
4.2 Resultat av datainsamling	17
4.2.1 Packmaskinshaveriet	18
4.2.2 Transportbandshaveri	19
4.2.3 Passningshaveriet	19
4.2.4 Blockering orsakat av palletteringen	20
4.3 Resultat av modellen	20
4.4 Resultat av Interfacet	22

4.5 Validering	24
4.6 Kontroll av kapacitet	25
4.7 Återkoppling till frågeställningarna	26
5. DISKUSSION.....	28
5.1 Resultatdiskussion.....	28
5.2 Metoddiskussion.....	29
6. MILJÖ	30
7. SLUTSATS	31
KÄLLFÖRTECKNING	32

Ordlista

2D	- Tvådimensionell
3D	- Tredimensionell
Buffert	- Ett minilager som kompenserar för variation i produktionen
Conveyor	- Transportband
Loads	- Den last som åker igenom systemet och som blir behandlad genom att specifika värden, attributer, ges till den
Process	- I processer görs alla uppgifter som påverkar samtliga lådor i systemet
Branches	- När koden förgrenas till en ny lösning och detta ska separeras
MTTR	- Mean time to repair, den genomsnittliga tiden det tar att åtgärda ett fel
MTBF	- Mean time between failure, den genomsnittliga tiden det tar innan nästa fel uppstår, med start från då ett fel uppstår
MTTF	- Mean time to failure, tiden det tar från det att ett fel är åtgärdat tills det att ett nytt fel inträffar
GSWA	- Inlärningsprogram för Automod, "Getting Started With Automod"
Växel	- En del i systemet som bestämmer vilken väg varje enskild låda ska gå
Interface	- Gränssnittet fungerar som en styrenhet som kopplar all väsentliga indata till koden

1. INLEDNING

Inledningen beskriver projektets bakgrund, syfte, frågeställning samt dess avgränsningar.

1.1 Bakgrund

Effektivisering och balansering av produktionen är oerhört viktigt för att företag ska kunna vara konkurrenskraftiga. genom att uppnå detta kan ett företag få stora fördelar när det kommer till resurs och tidsallokering. Arbetet är baserat på att ge företaget en bild över möjligheterna för balansering och förbättringar genom att simulera delar av produktionen (Hågeryd, Björklund, & Lenner, 2002, s. 436-437).

Anledningen till att simulering blir allt vanligare är dess stora fördelar. Genom att testa idéer och koncept i ett simuleringsprogram istället för att i tidigt skede testa sig fram i verkligheten har visat sig vara gynnsamt. Misstag som görs i fysisk form kan vara fysiskt skadligt för de medverkande, ekonomiskt kostsamt och att det bidrar till onödig miljöpåverkan (Hågeryd et al., 2002, s. 437-439).

Simulering ger en indikation för vilka effekter man kan vänta sig om föreslagna idéer och koncept skulle implementeras i verkligheten, vilket kan vara gynnsamt för ledningen vid beslutsfattande. Vidare kan detta även användas för att förbereda övriga involverade på de planerade förändringarna som kommer att ske framöver, genom ett tydligt och visuellt medium (Heilala, 1999, s.15).

Företaget har ett problem huruvida en framtida investering i palleteringsområdet kommer påverka transportbandssystemet. Då är simulering ett lämpligt verktyg som kan användas för att lösa problemet. Då företaget ska vidareutveckla palleteringen finns behovet av att analysera transportbandssystemet, vilket är det föregående systemet till palleteringen. Företaget får då en insyn hur en uppgradering av palleteringen påverkar helheten.

1.2 Syfte och mål

Målet med arbetet är att simulera transportbandssystemet som går mellan samtliga packmaskiner och ut till palleteringsområdet. Detta görs ur ett makroperspektiv med Automod för att hitta eventuella flaskhalsar i den valda delen av transportbandssystemet. Transportbandssystemet valdes av anledningen att det är känt att palleteringen är problematisk, däremot har företaget inte fullständig koll på transportbandssystemets karaktär. För att arbetet ska vara lämpligt omfattande har fokuset avgränsats till att vara mellan paketeringsmaskinerna och palleteringsområdet. Slutligen skall en färdig modell, av det avgränsade transportbandssystemet, som representerar den verkliga fabrikslayouten tas fram i Automod.

Nedan är en punktlista av projektets mål:

- Kartlägga driftstopp och dess orsak för att kunna införa dem som verklighetstroga stopp i simuleringen.
- I förberedelse inför framtida investeringar vill företaget kunna ändra värden och ha en interaktiv simulering, som är användarvänlig.
- Modellen ska kontrollera transportbandssystemets maximala kapacitet och se om den blir en flaskhals efter framtida investeringar i palleteringsområdet.

Syftet med projektet är att få en bättre förståelse och ge ett helhetsperspektiv av transportbandssystemet. Med ytterligare förståelse om systemet kan mer välgrundade val tas av ansvarig personal för framtida investeringar. Då företag ofta förlorar värdefull information när personal slutar kan en del av denna kunskap säkras upp i simuleringen.

1.3 Frågeställning

Denna rapport ska besvara tre stycken frågeställningar. Dessa är listade nedan.

- Vilka driftstopp förekommer i transportbandssystemet och hur kan dessa simuleras på ett verklighetstroget sätt?
- Hur kan en simuleringsmodell utformas så att den på ett användarvänligt sätt blir användbar för företaget vid kommande investeringsbeslut gällande test av förbättringar?
- Kommer företagets transportbandssystem ha tillräcklig kapacitet för att inte bli en flaskhals efter kommande investeringar gjorts i palleteringsområdet?

1.4 Avgränsningar

Projektarbetet har begränsats till att simulera transportbandssystemet från packmaskinerna, där produkterna är färdigpackade i lådor, fram till palleteringsområdet. Följande förenklingar av verkligheten behövde göras på grund av bristfällig data och tid.

- Tiden som systemets körs i simuleringen har begränsats till en normal arbetsvecka, 95 timmar, med anledning av företagets omställningsschema som skall kunna skrivas in i systemet
- Palleteringsområdet simuleras ej, men eftersom dess kapacitet påverkar transportbandssystemet kommer detta ses som en kö som kan ta emot ett visst antal lådor

- Då en del information saknades fick det göras uppskattningar av vissa värden.
- Den stora mängd data som togs fram fick sammanställas till genomsnittliga värden, vilka fördelades med olika fördelningsfunktioner för att få simuleringen så verklighetstrogen som möjligt inom den tidsram som fanns
- Antaganden angående tillgängligheten på packmaskinerna gjordes utefter den data som fanns tillgänglig i företagets datasystem, då några exakta värden inte fanns tillgängligt
- En interface-struktur fanns tillgänglig, med hjälp av denna skapades Excel-sidor som kopplades samman och infördes i Automodkoden
- Föregående system, packmaskinerna, simuleras ej
- En teknisk miljöanalys görs ej av simuleringen

2. TEORETISK REFERENSRAM

Här förklaras den teoretiska bakgrunden till de verktyg och metoder som använts under projektet. Kapitlet skall ge en förståelse så att läsaren kan tyda resultatet samt analysen. Kapitlets rubricering följer banks teoretiska referensram.

2.1 Simulering

Simulering definieras som en imitering av en verklig händelse under ett angivet tidsintervall. Denna imitering kan användas för problemlösning och analysering av befintligt system eller för experimentering av nya koncept och idéer (Banks, 2004, s. 2).

Banks flödesschema består av 11 steg med återkoppling efter valideringen och verifiering. Steg görs i följande ordning:

1. Problemformulering
2. Projektplan
3. Datainsamling, bygge av modell
4. Programmering
5. Verifiering, återgår till steg 4 om kraven ej möts
6. Validering, återgår till steg 3 om kraven ej möts
7. Experimentdesign
8. Utföra experiment och analys
9. Dokumentera program och resultat
10. Fler experiment
11. Implementering

Banks flödesschema förklarar det händelseförlopp som följts under simuleringens uppbyggnad och genomförande. Vid både verifiering, validering och experimentering bör en utvärdering göras om man bör gå tillbaka till ett tidigare steg enligt modellen ovan. De steg som loopen omfattar är mer tidskrävande då de görs flera gånger. Alla steg används inte alltid utan projektets mål och avgränsningar avgör vad som är relevant (Banks, 2004, s. 43).

Inom simulering finns fyra tillvägagångssätt:

1. Händelseschema-metoden
2. Trefas-metoden
3. Aktivitetsskanning-metoden
4. Processinteraktions-metoden

Händelseschema-metoden bygger på att ett schema får rörelser att inträffa, som i ett tvättmaskinssystem där en händelse styrs av tvättsschemat. Trefas-metoden är ett gammalt system för att spara datorkraft som kör programmet i tre faser. Aktivitetsskanning-metoden avancerar tiden med en viss mängd och ser om en aktivitet aktiveras. Den metod som lämpar sig bäst för detta projekt är processinteraktions-metoden, vilket är när ett objekt färdas genom ett flöde och påverkas av olika händelser som kan påverka dess rörelse. Händelserna

kan vara exempelvis olika haverier eller stopp som gör att systemet påverkas och speglar den imperfekta verkligheten (Banks, 2004, s. 36).

2.1.1 Problemformulering & projektplan

I alla projekt behövs en problemformulering samt en projektplan som ska leda fram till svaret på denna formulering. Simuleringen som ska tas fram är ämnad för att på ett tydligt sätt, genom en enklare datorredovisning av modellen samt dess resultat, visa hur problemformuleringen har lösts. På så sätt kan alla inblandade parter förstå hur problemet lösts och få ut den viktiga informationen som önskas. Kärnan i detta är att ha en grundläggande plan för vad som skall undersökas och hur man ska gå tillväga (Banks, 2004, s. 18).

Genom att ställa frågorna innan modellen byggs, och därmed styra den åt rätt håll från början, minskar risken för ogjort arbete eller felaktigt resultat. Problemförståelse är nyckeln i detta stadium framförallt i områdena kring datainsamling och helhetsperspektiv. Kompromisser görs i projektplanen där tid vägs mot detaljnivå, även här måste helhetsperspektivet tas in i beräkningen (Hågeryd et al., 2002, s.442-443).

Projekt startas oftast med ett möte där de inblandade parterna diskuterar de mål som ska eftersträvas och hur dessa skall uppnås. Mål är viktigt då de är dessa som projektet sedan skall kämpa mot. Målen kan dock ändras under projektets gång om det skulle ske oförutsedda händelser.

2.1.2 Datainsamling

Informationen som simuleringen kräver samlas in från det befintliga systemet som skall simuleras. Det vanligaste felet inom simulering är att informationen som finns lagrad är otillräcklig eller felaktig, vilket är ofta förekommande på många företag, detta leder till att resultatet ej blir tillräckligt representativt av verkligheten (B. Johansson, personlig kommunikation, 6 februari 2019).

Data kan både finnas och behöva samlas, uppskattas eller skapas. Därmed efterföljs följande händelseförlopp vid datainsamling. Den insamlade datan analyseras utifrån punkt 1 till 5 i följande lista. Vid datainsamling går man igenom listan uppifrån neråt, om svaret är JA på en punkt är nästkommande steg onödiga, är svaret nej försätter man med nästa punkt i listan (Hågeryd et al., 2002, s. 444).

1. Finns datan? JA/NEJ
2. Kan datan samlas med mätning eller intervju? JA/NEJ
3. Kan datan uppskattas? JA/NEJ
4. Kan datan skapas? JA/NEJ
5. Förenkla modellen eller ändra mål? JA/NEJ

Datainsamlingen gällande hastigheter och avstånd sker generellt sett med hjälp av fysiska mätningar och CAD-ritningar. Ritningar kan vara både 2D och 3D, båda har för och nackdelar beroende på det projekt som används. 2D gör att modellen blir snabbare att modellera men ger inte samma översikt som en 3D ritning (Hågeryd et al., 2002, s. 440-441).

När data ej finns tillgängligt, får man uppskatta eller skapa data enligt modellen ovan. Då kan tidmätning, subjektiva antaganden och rimliga antaganden göras. I ett system som rör sig beroende av tid kan tidmätningar användas för många förenklingar (Banks, 2004, s. 25).

En stor mängd data kan sammanställas till matematiska fördelningsfunktioner. Fördelningsfunktionerna används för att beskriva haverier och stopp. Haveritider varierar och datan som samlas in är spridd, svårtolkad och ej linjär. Fördelningen väljs beroende på vilken indata som finns tillgängligt och hur systemet beter sig (Banks, 2004, s. 50).

2.1.3 Bygga modell

Simuleringsprogram gör det möjligt att testa förbättringsförslag och befintlig maxproduktivitet utan att störa den pågående produktionen.

Byggandet av modellen är den primära delen av arbetet, och sker löpandes jämsides med datainsamlingen. Modellens uppbyggnad sker i en simuleringsmjukvara. Simuleringsmjukvaror varierar i programmeringsorientering och riktar annars in sig på mer visuellt framställande av simuleringen (Konsult, personlig kommunikation, 13 maj 2019).

Modellbygget är i grunden ett sätt att bygga upp en virtuell värld med logik och matematik. Simuleringen byggs först upp med ett enkelt mål i åtanke, utan att bygga in komplexiteten. Det sker ofta genom uppritning av de visuella elementen. Vidare placeras enkla processer in i systemet. Senare läggs resurscyklar in vilket kontrollerar hur ofta driftstopp inträffar på packmaskiner, transportband och motorer (Banks, 2004, s. 44).

2.1.4 Verifiering

Under verifiering och validering bekräftar man att systemet stämmer överens med verkligheten. Verifiering görs löpande under arbetet, vilket innebär att datan kontrolleras under samtliga moment med den informationen som finns tillgänglig. För kodningen betyder detta en tydlig kodlogik samt en bra dokumentation över vad som gjorts. Kodningen bör även bli verifierad av mer än en person, detta för att garantera att eventuella misstag och fel upptäcks (Banks, 2004, s. 53-55).

Tillförlitligheten hos den data som projektet baserats på är av essentiell betydelse för att simuleringen ska vara av nytta. Därför bör alla parametrar verifieras med verkligheten. Företagen kan inte alltid garantera att parameterinformation stämmer eller ens finns, därför bör all information verifieras innan användning inom simulering (Skoogh & Johansson, 2008, s. 1).

2.1.5 Validering

Valideringen görs genom att kunniga personer inom simulering, samt någon som har god insikt i det specifika systemet som simuleras, granskar simuleringen och dess resultat. Makroperspektivet och helheten bör även analyseras för att se om systemet är rimligt. Inom simuleringsmjukvaran finns även ett visuellt verktyg där ologiska fel kan upptäckas i den modulerade simulationen. Här ser man produktionens flöde och hur dess laster rör sig. Genom att analysera rörelserna ur ett rimlighetsperspektiv kan man se om de stämmer överens med verkligheten (Banks, 2004, s. 53-55).

Vidare kan utdata analyseras och jämföras med verkligheten för att se om de stämmer överens. Detta inkluderar ett antal parametrar som representerar hur väl modellen simulerar verkligheten. Antalet entiteter som rör sig genom systemet till slutdestinationen ger ett mätvärde som är användbart vid valideringen (Sargent, 2011, s. 186-187).

2.1.6 Experimentdesign & analys

Under experimentdesignen så testades nya lösningar baserade på kunskapsbaserade antaganden. Dessa antaganden kommer från insamlad data eller personlig kunskap som samlats in under simuleringens uppbyggnad, genom samtal med kunniga personer eller ren intuition. När den experimentella designen senare testas så analyseras även resultatet, där rimligheten i antagandet bekräftas eller avvisas (Banks, 2004, s. 45).

2.1.7 Dokumentera Resultat och implementering

Dokumentation är nödvändigt för att senare kunna utveckla och expandera det system som byggs. För att systemet ska kunna användas igen måste kunskapen från projektet bevaras. Vidare är modifiering och ändringar under projektets gång lättare om kunskapen runt projektet finns, alla resultat och analytiska slutsatser som uppnås bör därför dokumenteras (Banks, 2004, s. 46).

Livscykeln av en modell påverkas av hur komplicerad den är att förstå för utomstående. Då gäller även om den kan uppdateras och ej är låst. Andra utomstående som vill ha information av modellen behöver ha en viss nivå av dokumentation för att kunna begripa kontexten. Indata, experimentdata och modelldata är alla viktiga att dokumentera och analysera (Hågeryd et al., 2002, s. 446-447).

Implementering av simuleringssystem ger en möjlighet att testa scenarion som annars hade varit för dyrt eller tidskrävande att göra i verkligheten. Simuleringen kan då liknas till en rapport om ett scenario man vill testa. Rapporten kan vara en grund till ett framtida beslut eller investeringar (Banks, 2004, s. 46).

2.2 Github

Github är en plattform för att dela kod med medarbetare så att samtliga deltagare kan arbeta på ett gemensamt arbete. Genom denna plattformen kan man arbeta med fler branches där experiment kan utföras utan att påverka den huvudsakliga koden. På så sätt kan olika lösningar testas samtidigt. Vidare undviks arbete med att föra över kod manuellt, det minskar framför allt risken för misstag, samt att det gör arbetet mycket mer effektivt (Chacon & Straub, 2019, s. 8).

3. METOD

Under metod presenteras hur projektet genomfördes samt vilka metoder som användes. Här följdes även Banks metodik med samma rubriceringar som i teorikapitlet.

3.1 Problemformulering & projektplan

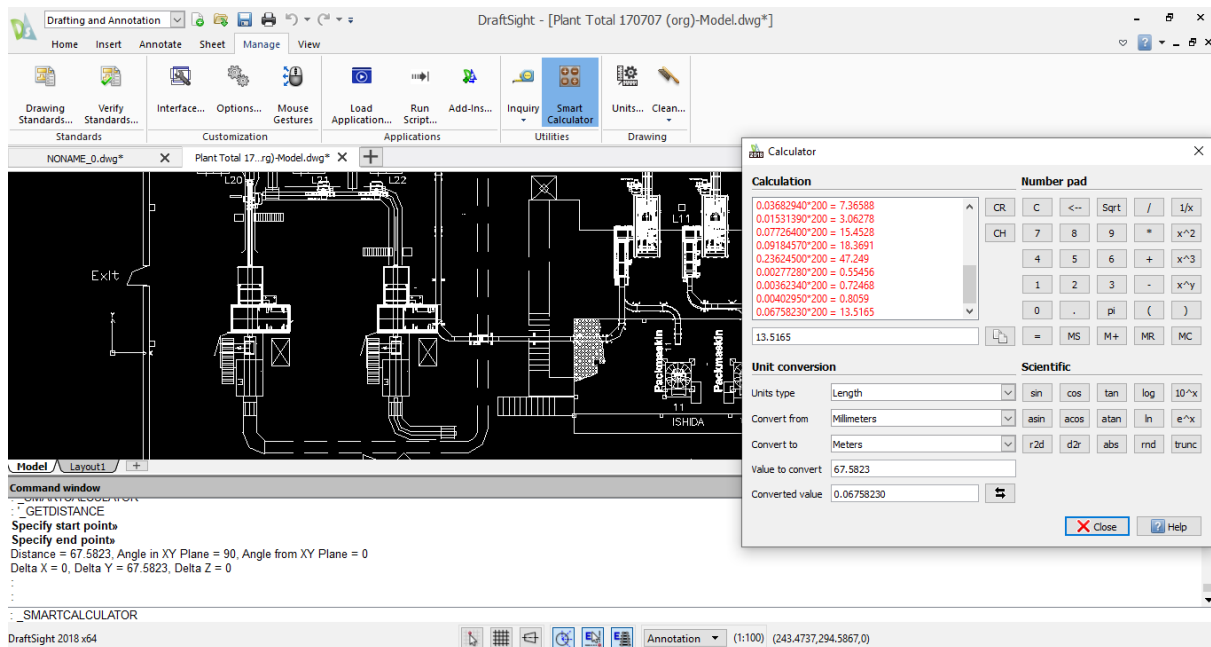
Projektet startades med ett inledande möte med representanter från samtliga involverade parter. På mötet deltog projektmedlemmarna, handledare från konsultföretaget samt en processingenjör och en projektledare från företaget. Under mötet sattes mål och avgränsningarna för projektet samt hur genomförandet skulle gå till praktiskt.

3.2 Datainsamling

Företaget besöktes en gång varje vecka med skiftande fokus, beroende på hur aktiv produktionen var vid varje enskilt tillfälle, samt på vilken data som var önskvärd vid respektive tillfälle.

Datainsamlingen skedde med flertalet metoder för de olika delarna av projektet. Till att börja med samlades allmän data in genom fabriksbesök och samtal med anställda på företaget för att få en helhetsbild av produktionen. Genom dessa samtal tillhandahölls befintlig information om systemet. Informationen erhöles i form av ritningar och dokument med listade kapaciteter och hastigheter för de berörda delarna av produktionen. En mängd fysiska mätningar av både tid och avstånd gjordes härnäst. Slutligen påträffades ett system som loggade samtliga driftstopp, anledningen till att företaget inte informerat om detta direkt var att de till en början inte trodde denna detaljerade information var nödvändig.

Den första informationen som erhöles var en digital 2D ritning över transportbandssystemet. Under besök på företaget mättes fysiska avstånd för att säkerställa att de ritningar som tilldelats var korrekta och överensstämmande med verkligheten. De fysiskt uppmätta avstånden jämfördes med ritningarna och de stämde väl överens med varandra. Med hjälp av de digitala ritningarna i programmet Draftsight kunde transportbandssystemets dimensioner mätas, även på de platser där en fysisk mätning hade varit svår att genomföra, se bilden nedan.



Figur 1: Mätning av transportbandet med 2D ritning.

Med hjälp av tidtagarur mättes tider på olika sträckor och växlar. Med hjälp av dessa tider kunde en mängd information fås fram, såsom effektiviteten och eventuell fördröjning på samtliga växlar samt hastigheten på transportbanden. Tack vare tidmätning på packmaskinerna kunde även takttiden för alla packmaskiner kontrollerats för att senare jämföras med de av företaget tilldelade värdena. Det fanns ingen information på ritningarna om var drivmotorer befann sig eller var banden skarvats, därmed kontrollerades också detta fysiskt i fabriken.

För att fånga beteendet vid driftstopp på transportbandet var det tvunget att påtvinga en sådan händelse. Genom att stoppa in en aluminiumstång mellan två stålpelare kunde ett hinder skapas, vilket gjorde att lådorna fastnade och därmed orsakade köbildning. Meningen med detta var att åstadkomma framförallt två saker, nämligen att se vilken/vilka sensorer som registrerade köbildningen samt hur transportsystemet stegvis stoppades vid köbildningen. Det var av stor vikt att beteendet vid driftstopp fångades så exakt som möjligt då denna information senare användes i kodningen, detta för att transportbandssystemet skulle bli så verklighetstroget som möjligt.

Mängden driftstopp på packmaskinerna, och dess omfattning, är något av det mest avgörande för hela systemets produktivitet. Därmed var det viktigt att samla in så mycket väsentlig och exakt information som möjligt om detta. Denna data samlades in genom en sammanställning av företagets egna data där de anställda loggade när fel inträffade, hur länge de varade och hur lång tid det tog innan problemen åtgärdats. Denna tabell har uteslutits ur rapporten på grund av dess storlek.

En problematisk del av transportbanssystemet var där transportbanden passerade ett hål genom en vägg, vilket ibland orsakade att lådor fastnade i hålet. Datan för passningsfel, som orsakade stoppen vid hålet i väggen, samlades in med hjälp av företagets processingenjör. Processingenjören hade gjort en studie på just detta fel, vilken omfattade dess frekvens och hur lång tid det tog innan felet åtgärdats. Informationen kom från processingenjören, anställd

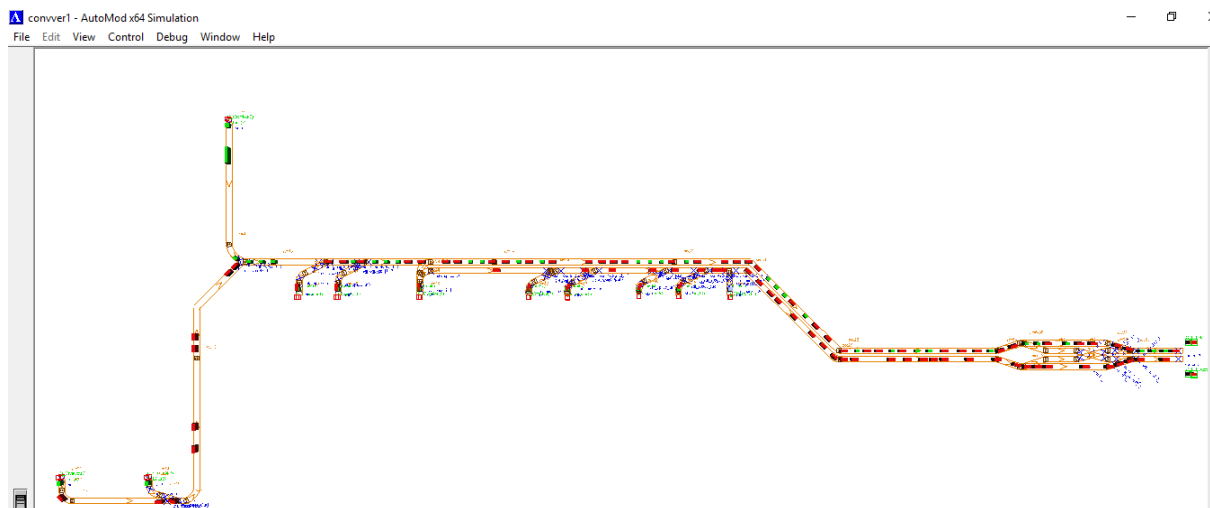
på företaget, med över 10 års erfarenhet inom området och informationens grundläggande delar sågs därmed som säkra.

Palleteringshaveriet granskades genom att fråga processingenjörerna, som ansvarade för förbättringen av palleteringen, om hur den påverkar systemet. De tog då fram förenklad data om hur mycket lådor varje transportband kunde ta emot och hur dessa påverkade systemet.

3.3 Bygga modell

För att kunna genomföra projektet användes ett inlärningsprogram, kallat GSWA, på så sätt erhöles den nödvändiga kunskapen som krävdes. I Automod gjordes majoriteten av arbetet, där inkluderades samtliga steg i simuleringen såsom uppbyggnaden av visuell modell, kodning och färdig simulering.

I Simuleringsmjukvaran finns olika system som är inriktade på olika områden. Vid arbete med transportband går man in i "conveyor system" i Automod. Där finns möjligheten att bygga upp ett transportbandssystem med hjälp av färdiga moduler, som man senare dimensionerar enligt önskemål. I "conveyor system" utförs ingen programmering utan enbart ritningen av transportbandssystemet samt utplacering av packmaskiner. En packmaskin är en fysisk plats i systemet som skapar lådor och skickar ut dem på transportbandssystemet. Dessa packmaskiner används senare under själva kodningen där systemets beteende formas och de används som referenser i koden för att programmera händelser.



Figur 2: Transportbandssystemet i Automod, sett ovanifrån.

Kodningen utfördes mestadels i Visual Studio där den logiska uppbyggnaden och struktureringen av koden är lättare än i Automods egna kodningsmjukvara på grund av språktillägg gjorda av konsultbolaget (Konsult, personlig kommunikation, 13 maj 2019).

Metodiken som följdes vid kodens uppbyggnad var följande:

1. En enkel modell byggdes upp där en produkt kan ta sig från punkt A till punkt B på det korrekt utritade visuella systemet.
2. I andra steget adderades korrekta parametrar där hastighet, växlar och påfarter undersöktes för att bekräfta dess exakthet gentemot verkligheten.
3. I steg tre placerades driftstörningar i systemet. Dessa driftstörningar kan delas in i fyra kategorier, packmaskinshaveri, passningshaveri, omställningar och haverier från efterkommande system dvs palleteringen, se bilaga 1.
4. Design av och sammankoppling med ett interface, som ska ge företaget möjligheten att experimentera inför framtida investeringar, inom satta gränser.

Under projektets gång har även felsökningar genomförts samt lämpliga åtgärder för att eliminera de identifierade problemen vidtagits. Återkoppling efter implementering av en förändring granskades, om den uppfyllde de satta målet, om målet ej nåddes så gjordes koden om.

3.4 Verifiering

Löpande under arbetets gång har arbetssättet, parametrar samt modellens uppbyggnad verifierats med hjälp av lämpliga experter inom området. De fysiska parametrarna har kontrollerats av två anställda processingenjörer på företaget. Modellen samt programmeringen har verifierats av anställda på konsultbolaget som arbetar inom just simulering med Automod.

Koden verifierades kontinuerligt av handledarna från konsultbolaget. Allt eftersom lösningar på koden togs fram kontrollerade handledarna kodens kvalitet. Lösningar som ej höll måttet byttes ut mot bättre alternativa lösningar som var mindre komplexa. Kodens effektivitet är mätbar genom längd och dess flexibilitet vid förändringar. Metodiken följdes under hela projektet för att garantera att slutprodukten höll måttet.

De parametrar som utgavs av företaget verifierades genom mätningar i det verkliga och befintliga systemet. Från företaget hade dimensioner, hastigheter och kapaciteter angivits, och dessa kunde verifieras på följande sätt. Dimensionerna mättes i verkligheten med måttband och jämfördes med de angivna dimensionerna, och det visade sig att de stämde överens. Hastigheter på de olika transportbanden kunde med enkla beräkningar fås fram genom att med tidtagarur mäta tiden som en låda transporterades på en sträcka, vilket i sin tur ledde till en verifierad hastighet.

Kapaciteten hos packmaskinerna fanns angiven i form av teoretiska värden, dessa kunde genom fysiska mätningar av respektive packmaskins taktid kontrolleras och därmed verifieras. Samtliga växlar kunde med hjälp av tidtagning ses som jämlika och dess kapacitetsförmåga detekteras, genom analysering av växlarnas beteende.

Handledare/expert	Problem som de hjälpt att verifiera
Processingenjör företaget	Passningshaveri
Processingenjör företaget	Palleteringshaveriet
Senior produktionsingenjör företaget	Packmaskinshaveri
Handledare konsultbolaget	Programmeringsproblem i Automod
Handledare konsultbolaget	Dataprocessering
Processingenjör företaget	Omställning
Handledare på Chalmers	Struktur av rapporten

Tabell 1: En sammanfattning av de personer och vad de hjälpt verifiera.

3.5 Validering

Inledningsvis så validerades systemet muntligt med såväl företaget som konsultbolaget. Då granskades helheten. Genom att jämföra utdatan av simuleringen med den verkliga datan från företaget, under en specifik tidsperiod, kunde det simulerade systemet valideras gentemot verkligheten. På så sätt fås en procentuell exakthet ut, vilken kan användas vid framtida simuleringar förutsatt att felmarginalen är inom rimliga gränser.

Fortsättningsvis utvärderades systemet visuellt jämfört med verkligheten för att se om rörelserna är liknande, och på så sätt få en mer verklighetstrogen simulering. Detta genom att regelbundet besöka transportbandssystemet för att själva se verkligheten samt att fråga de anställda hur systemet fungerar.

3.6 Experiment design och analys

Metoden för experimentdesign och analys gav företaget möjligheten att ändra olika parametrar, fördelningsfunktioner och startvärden. Med ett interface kan dessa värden ändras vilket ger simuleringen en större relevans, då en produktionslinje hela tiden förbättras. Målet med interfacet var att kunna, på ett användarvänligt sätt ge möjlighet för ändrade indata och tydliga utdata från simuleringen.

3.7 Dokumentera resultat och implementering

För att hålla koll på simuleringsmodellens utveckling under arbetets gång och lätt kunna följa dess uppbyggnad har kommentarer satts in i koden. Dessa kommentarer förklarar de olika stegen som tagits vid kodens uppbyggnad. Koden innehåller alltså en enkel form av förklarande instruktioner för varje händelse, se bilaga 1 för exempel.

Löpande har rapporten uppdaterats i jämn takt jämsides med det praktiska arbetet, vilket gjort att händelseförloppet lätt kan följas och förstås. Följande delar har varit i fokus, indata, experimentdata och modelldata, vilka har dokumenterats i Excel-dokument med tillhörande förklaringar.

Den sista delen av projektet medförde att företaget fick möjligheten att ändra in-parametrar genom ett interface. Interfacet styrde de parametrarna som företaget, konsultbolaget och projektgruppen kände var relevanta och realistiska inom ramen för projektet. Metodiskt granskades företagets data för att anpassa parametrarna, så att de fanns rum för senare ändringar från företagets sida.

3.8 Github

Metoden som användes för att säkra upp projektet från data-förluster var Github. Den använder en molntjänst för att under projektets gång dela olika versioner av koden som skrivs, bearbetas och slutligen används. Detta gör att man slipper tidskrävande manuella överföringar, vilket sparar tid och möjliggör även mer flexibla arbetsförhållanden. Koden lades upp på denna molntjänst med olika versioner så att mer än en kunde arbeta på koden. Rent praktiskt innebar detta att båda kunde jobba på olika problem, i samma kod. Github användes för att det är gratis och användbart.

4. RESULTAT

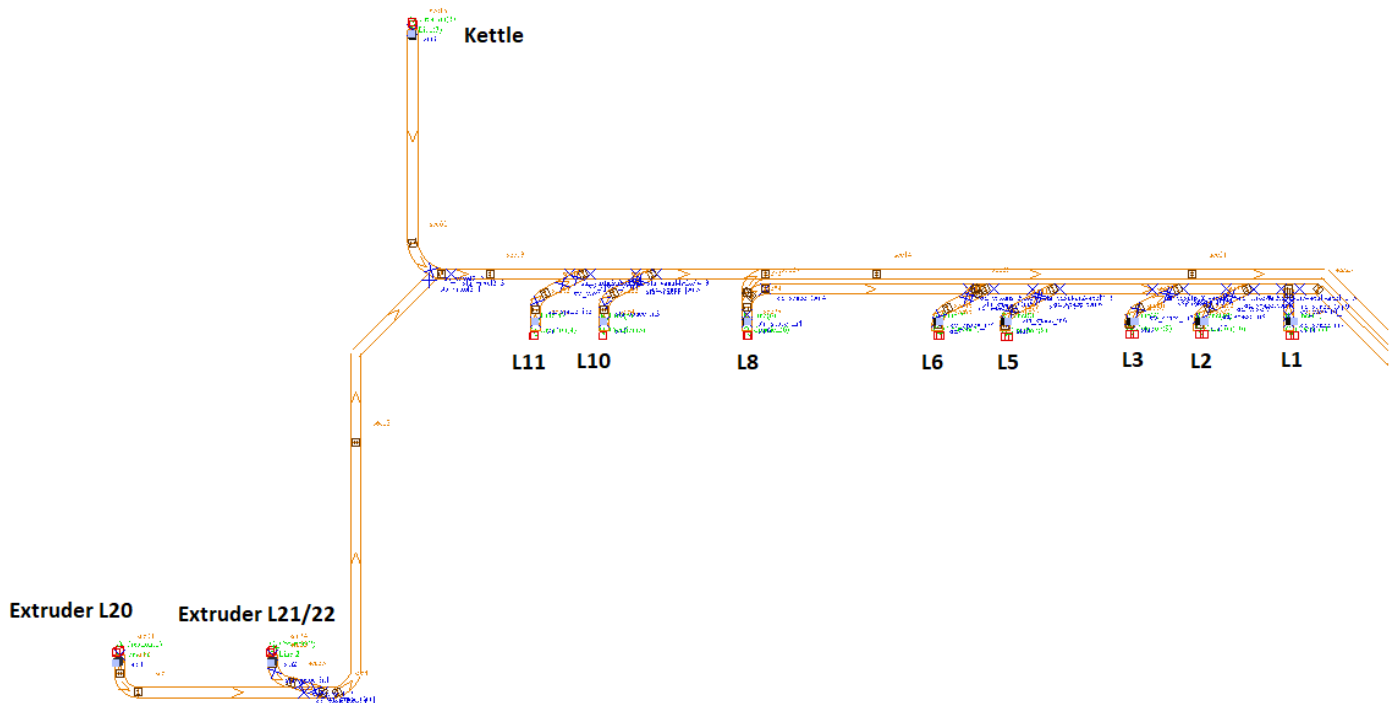
I resultatkapitlet presenteras resultaten som uppnåddes med metoderna som har beskrivits i metodkapitlet. Här presenteras även resultatet av nulägesanalysen, datainsamlingen, hur modellen byggdes, interfacet, valideringen samt kapacitetssimuleringen på bandet. Slutligen sammanfattas svaren till frågeställningarna utifrån övriga delar av resultatkapitlet.

4.1 Nulägesanalys

Här presenteras den grunddaten för simuleringen som togs fram men som ej finns representerat i resultatdelen. Kapitlet ger kompletterande information för att underlätta förståelsen för resultatet.

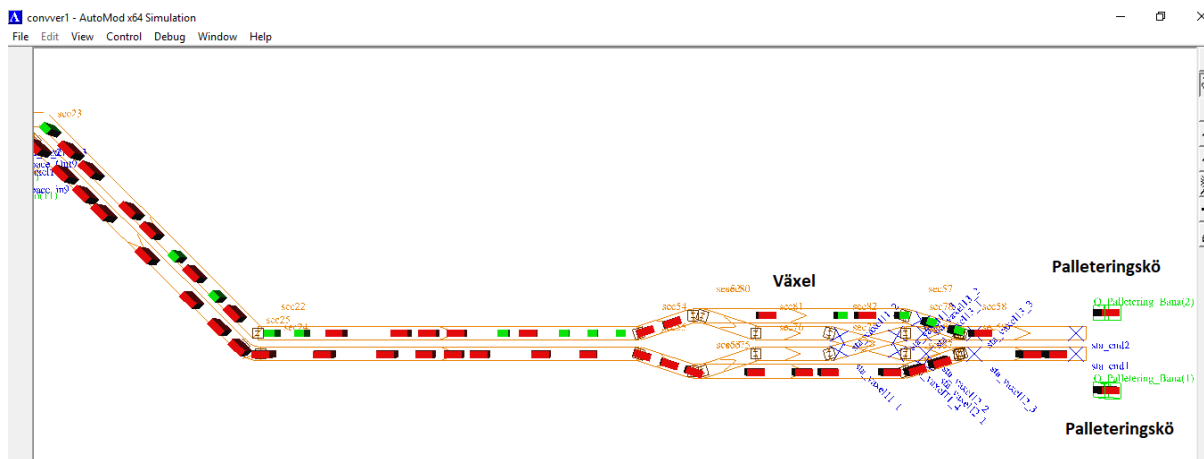
4.1.1 Systemets funktionalitet

Bilden nedan visar var olika packmaskiner befinner sig i simuleringen.



Figur 3: Del 1/2 av produktionen, översikt var de olika packmaskinerna ligger i systemet.

Systemets grundläggande funktion är att skicka lådor från punkt A till punkt B, punkt A är packmaskinerna som ses i figur 3 och punkt B är palleteringsköerna som ses i figur 4 nedan. Mellan dessa finns även en växel som syns i figur 4.



Figur 4: Del 2/2 av produktionen, här syns punkt B, palleteringskön.

4.1.2 Packmaskinerna

Här presenteras packmaskinernas matningshastigheter, dessa värden användes i simuleringen men dessa får inte yttras i rapporten på grund av sekretess.

Packmaskiner	Matningshastighet
Extruder L20	Sekretessbelagt
Extruder L21/L22	Sekretessbelagt
Kettle	Sekretessbelagt
L11	Sekretessbelagt
L10	Sekretessbelagt
L8	Sekretessbelagt
L6	Sekretessbelagt
L5	Sekretessbelagt
L3	Sekretessbelagt
L2	Sekretessbelagt
L1	Okänt*

Tabell 2: Matningshastigheter på packmaskiner *Denna packmaskin körs i nuläget ej ut på bandet, matningshastigheten är okänd.

Personalraster har blivit förenklade genom att ta bort en timme från varje 8 timmars pass. Förenklingen gör att simuleringstiden blir 95 timmar på en vecka.

4.1.3 Transportbanden

Här presenteras transportbandets hastighet och lådornas storlek.

Transportbandets hastighet:	0,4356 m/s
Dimension "Stor låda"	0,68*0,29m
Dimension "Liten låda"*	0,39*0,29m

Tabell 3: Här visas hastigheten och dimensionerna *De små lådorna är färgade grön och kommer endast från packmaskinen kettle.

4.2 Resultat av datainsamling

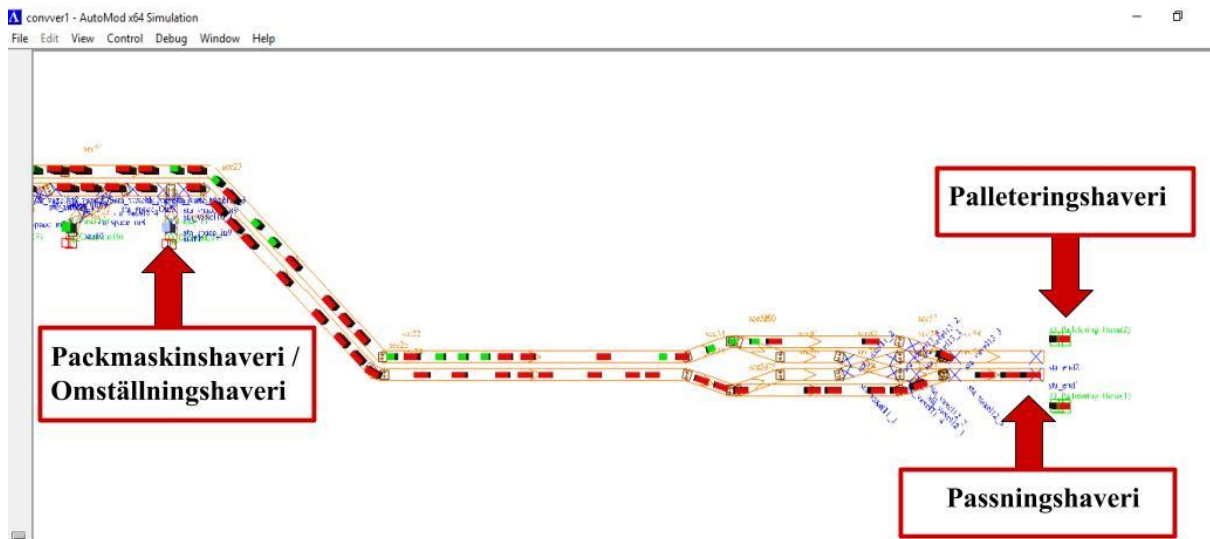
Under projektets gång samlades en hel del olika information in med hjälp av olika metoder beroende på vad som fanns tillgängligt, följande är det framtagna resultatet. Först presenteras informationen för olika driftstopp och sedan hur de infördes i simuleringen.

Under de inledande mötena med företaget kunde tre problem identifierats. Det första handlade om en grundläggande brist på kunskap om vem som hade ansvar och tillgång till den viktiga informationen inom företaget. Detta ledde till att mycket tid fick läggas på att lokalisera de rätta personerna. Det andra problemet var att själva informationen inte alltid var rätt på grund av felaktig inrapportering från de anställdas sida. Det tredje problemet var en grundläggande brist på information inom vissa delar av produktionen, vilka var essentiella för projektets validitet.

Ett grundläggande dokument, bestående av respektive packmaskins parametrar, användes för att få fram delar av den informationen som krävdes för att börja bygga upp systemets fundamentala delar. Informationen för haverier behövde införas, dessvärre saknades stora delar av denna information. För att få fram denna information användes datainsamlingsmetoder, vilka förklaras i figur 5. Allt som påverkade bandet negativt definierades som en kategori av haveri, detta inkluderade även händelsen då systemet blev blockerat av efterkommande palletering.

Haverierna delades in i olika delar som påverkade bandet på olika sätt. De haverier som identifierades var:

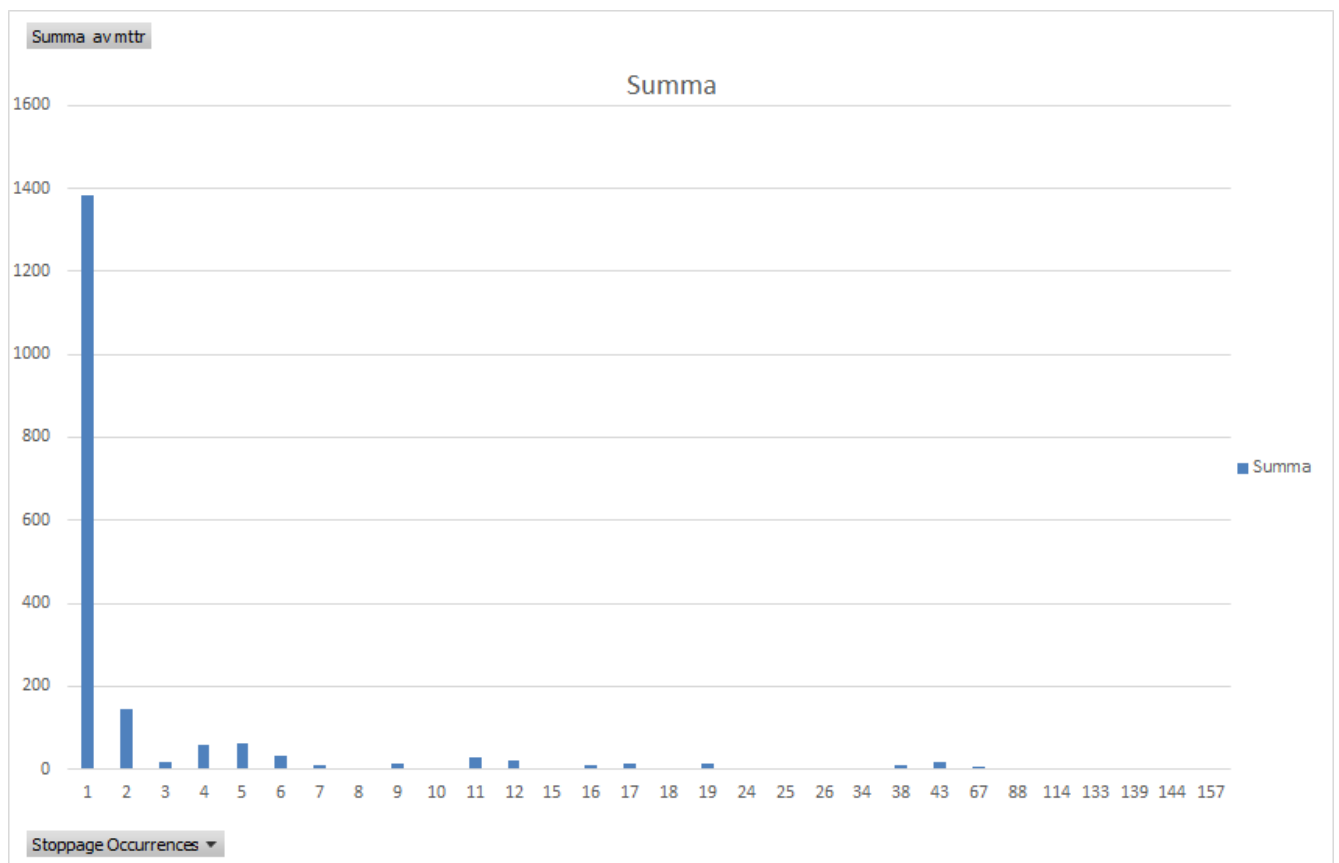
- Packmaskinshaveri på varje respektive packmaskin
- Transportbandshaveri
- Passningshaveri vid hålet i väggen
- Palleteringshaveri
- Omställningar



Figur 5: Visar var felen uppkommer i modellen.

4.2.1 Packmaskinshaveriet

Packmaskinshaveriets data fanns redan sammanställd och tillgänglig. Däremot var det inte möjligt att se de enskilda haverierna, vilket ledde till en viss mån av osäkerhet. Värdena viktades och sammanställdes till en fördelningskurva för att se hur mycket de olika felen påverkade systemet.



Figur 6: Exempel från packmaskin L2, tid för MTTR på x-axeln och antal stopp på y-axeln.

På figur 6 syns det att majoriteten av felen är små fel på 1 min och att det är några stora fel som drar upp genomsnittet. Här gjordes en förenkling genom att en triangulär fördelning användes för att beskriva kurvan. Dessvärre medförde detta nackdelen att de stora och extrema värdena inte fångades. Värdena på MTTF och MTTR togs fram, och var då något för höga på grund av extremvärdena, som kan ses i figur 6. Datan gav ett genomsnitt i MTTR och dessa tider fångar ej det dynamiska beteendet som egentligen består av fler små fel. Triangulär fördelning valdes då det var det bästa alternativet utefter den tid som fanns tillgänglig.

Packmaskiner	MTTF	MTTR
Extruder L20	49 min	12,6 min
Extruder L21/L22	25 min	12,6 min
Kettle	22 min	5 min
L11	24 min	1 min
L10	20 min	1,2 min
L8	21 min	6 min
L6	15 min	4,7 min
L5	21 min	9,6 min
L3	19 min	7,2 min
L2	10 min	4,8 min
L1	Okänt	Okänt

Tabell 4: MTTF och MTTR för alla packmaskiner.

4.2.2 Transportbandshaveri

Någon validerad data fanns inte på transportbandens eventuella haverier. Efter konsultation med företaget gjordes valet att försumma denna typ av fel, då transportbanden och dess tillhörande motorer var av hög kvalitet och medgav stor tillförlitlighet. Därmed resulterade de ytterst sällan i något verkligt haveri.

4.2.3 Passningshaveriet

Resultatet från passningshaveriet i väggen gav ett MTTF och ett MTTR som i sin tur tilldelades en triangulär fördelningsfunktion. Beteendet orsakat av sensorerna vid passningshaveriet gjorde att hela bandet successivt stängdes av bakåt i transportbandssystemet när ett fel inträffat.

Passningshaveriet	MTTF	MTTR
Undre Transportband	600 min	5 min
Övre Transportband	267 min	5 min

Tabell 5: MTTF och MTTR på passningshaveriet.

4.2.4 Blockering orsakat av palleteringen

Blockering orsakat av palleteringen gjorde att packmaskinerna inte kunde fortsätta att producera lådor, då palleterings produktivitet var lägre än föregående packmaskin. Då de två transportbanden i dagens läge inte hänger ihop så planerades det efter att olika värden skulle behöva tas fram för respektive transportband. Grafiskt sett ser systemets utvärden ut som en sinuskurva då systemet klättrar upp mot maxproduktion och blir därefter blockerat, vilket leder till en nedåtgående kurva, som på grund av en tröghet i systemet långsamt återhämtar sig. Detta orsakar en dominoeffekt som sprider sig bakåt genom alla produktionens delar.

Lådor/min som palletering på bana 1 (nedre) klarar:	Lådor/min som palletering på bana 2 (övre) klarar:
5st	12st

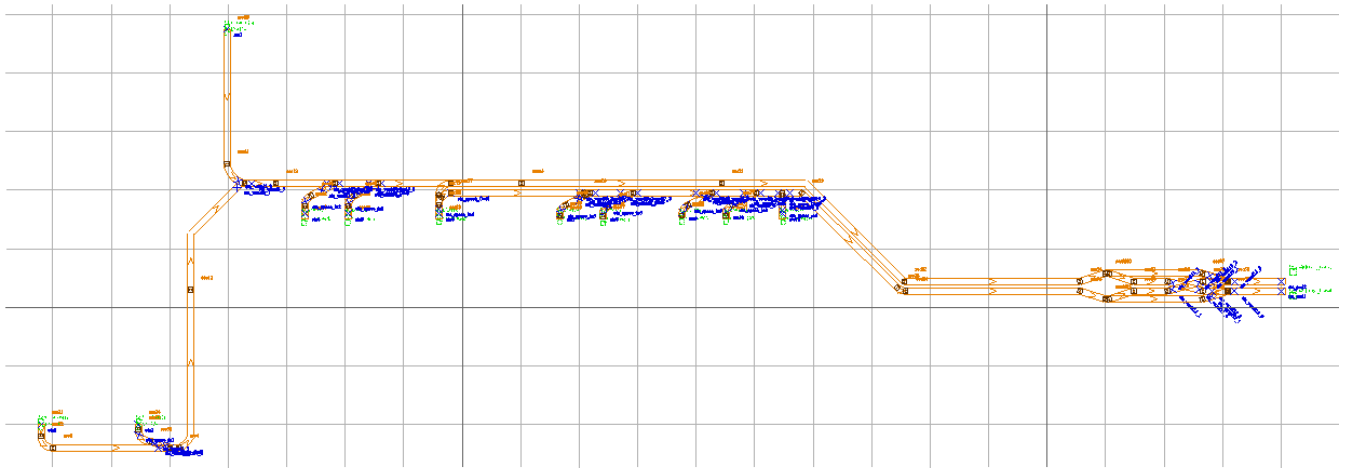
Tabell 6: Värdena som bana 1 och bana 2 har i interfacet.

Blockeringen orsakas av att palleteringen aktiverar samma sensorer som de vid ett passningsfel, detta stänger ner hela bandet successivt. Det finns en viss tröghet innan systemet är uppe i full gång igen efter ett stopp.

4.3 Resultat av modellen

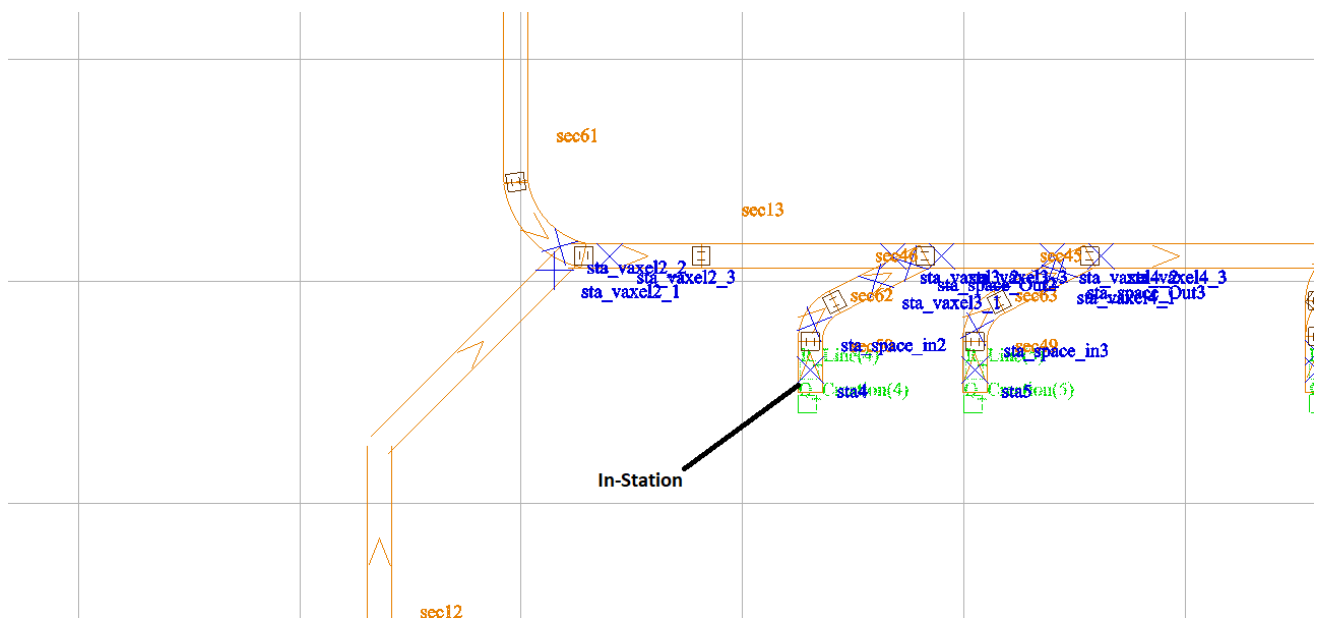
Här förklaras resultatet av uppbyggnaden av modellen och hur den fungerar. Hur de olika processerna är uppbyggda i koden hittas i bilaga 1.

Modellen byggdes i Automod med packmaskiner som skickar in lådor till transportbandssystemet. Dessa lådor passerar manuella växlar innan de kommer ut på transportbandssystemet för att sedan färdas till sin slutdestination. Figur 7 visar var de olika delarna är placerade.



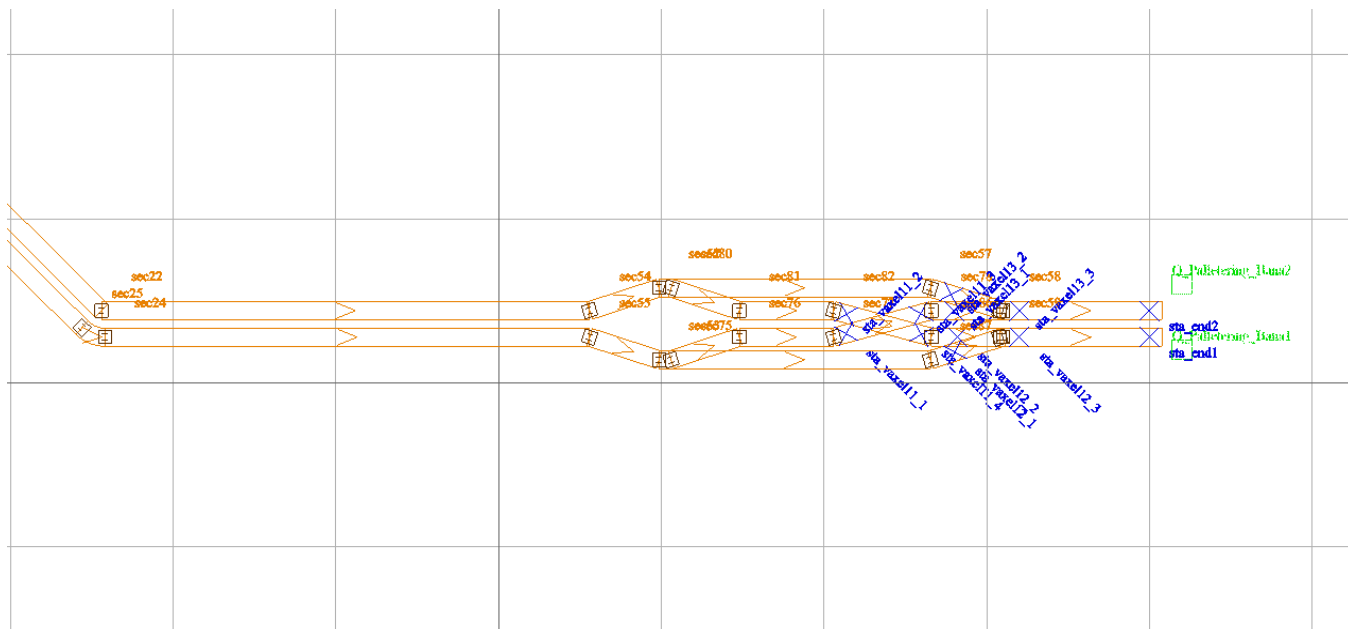
Figur 7: Automod modellen.

Varje In-station som representerar packmaskiner skickar ut lådor med olika fart beroende på indata, i figur 8 syns ett exempel på en sådan packmaskin där lådor går in i systemet.



Figur 8: Inzoomad bild på Automod modellen.

I slutet av systemet finns en växel vilken gör att lådorna kan byta slutdestination. Slutdestinationen avgör i dagsläget vilken palleteringskö som lådorna färdas till. I figur 9 kan slutdestinationen och växeln ses.

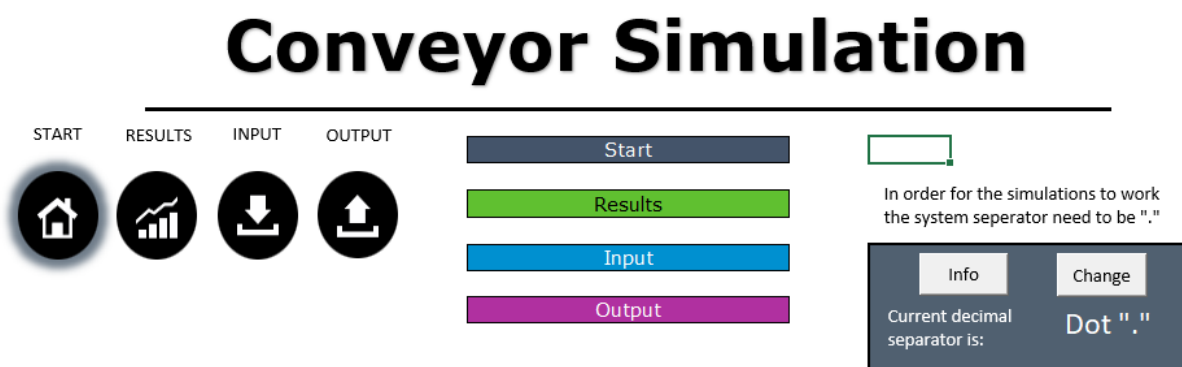


Figur 9: Slutet på Automod modellen, de gröna boxarna är palleteringsköerna.

Palleteringsköerna förenklar den nästkommande delen av produktionen, palleteringen. Transportbandet stannar om palleteringsköerna blir överfulla, och väntar sedan på att starta när kön har nått ett värde, definierat av indata.

4.4 Resultat av Interfacet

Här presenteras resultatet av interfacet som byggdes i Excel.



Figur 10: Startside i interfacet.

Interfacet består av tre viktiga delar, input, output och results som ses i figur 10. I inputen sattes företagets parametrar in, denna information överfördes sedan in i koden, se bilaga 1. De två transportbandens egenskaper justeras var för sig. Inputen omfattar totalt 5st Excel-sidor där företaget kan ändra följande faktorer:

Sida 1 packmaskinsinställningar:

- Packmaskinernas respektive slutstation
- ON/OFF för packmaskin
- Matningshastighet på packmaskinerna

Sida 2 haveri packmaskiner:

- Fördelningsfunktion för driftstoppen
- Tre fördelningsvariabler V1, V2, V3

ON	MTTF:	Type of distribution	V1	V2	V3		MTTR:	Type of distribution	V1	V2	V3
	Sta-1 Extruder	Triangular	40	49	58.8		Sta-1 Extruder	Triangular	1	12.6	16.38
	Sta-2 Extruder	Triangular	20	25	30		Sta-2 Extruder	Triangular	1	12.6	16.38
	Sta-3 Kettle	Triangular	18	22	26.4		Sta-3 Kettle	Triangular	1	5	6.5
	Sta-4 L11	Triangular	20	24	28.8		Sta-4 L11	Triangular	0.1	1	1.3
	Sta-5 L10	Triangular	16	20	24		Sta-5 L10	Triangular	1	1.2	1.56
	Sta-6 L08	Triangular	17	21	25.2		Sta-6 L08	Triangular	1	6	7.8
	Sta-7 L06	Lognormal	12	15	18		Sta-7 L06	Triangular	1	4.7	6.11
	Sta-8 L05	Triangular	17	21	25.2		Sta-8 L05	Triangular	1	9.6	12.48
	Sta-9 L03	Triangular	15	19	22.8		Sta-9 L03	Triangular	1	7.2	9.36
	Sta-10 L02	Triangular	8	10	12		Sta-10 L02	Triangular	1	4.8	15
	Sta-11 L01	Triangular	6	8	9.6		Sta-11 L01	Triangular	1	8	15
		Triangular Lognormal Normal Weibull Uniform Gamma Exponential Constant									

Figur 11: Sida två i interfacet, input för packmaskinshaverier.

Sida 3 passningshaveriet:

- Fördelningstyp för passningshaveriet
- Tre möjliga värden på fördelningen

Sida 4 omställningsschema för packmaskiner:

- Mängden lådor som ska göras av varje sort innan omställning, görs för varje packmaskin och kan max innehålla max 6 omställningar för en veckas produktion
- Vilken typ av omställning, lång eller kort

Sida 5 palletering-input:

- Antalet lådor palleteringskön klarar att ta in per minut från de två transportbanden
- Antalet lådor innan palleteringens kapacitet överbelastas
- Antalet lådor innan palleteringen återhämtar sig

All angiven information skrivs om automatiskt, genom interfacet, till textfiler som sedan läggs in i systemet. När simuleringen sedan körts klart skrivs informationen från Automod till resultatdelen av interfacet.

I outputdelen skrevs ett genomsnitt för hur mycket lådorna påverkades av respektive fel. Varje timme registreras hur mycket lådorna har påverkats av de faktorerna som ses i bilden nedan. Sedan tas ett genomsnitt fram, markerat med ljuslila, vilket även ses i bilden nedan.

	L8				
	Working	Station Breakdown	Changeover Type	Palletizing breakdown	Breakdown Hole
Min	28%	39%	0%	5%	0%
Average	30%	43%	3%	23%	1%
Max	34%	63%	10%	25%	1%

Figur 12: Bilden nedan är en av packmaskinernas utdata. *Station breakdown står för packmaskinernas haveri.

I outputen så får användaren även veta hur många lådor som producerats totalt och hur många lådor varje packmaskin producerat. I bilden nedan syns några av dessa som exempel.

Total Production		Extruder L20	Extruder Kettle	L11	L10	L08	L06	L05	L03	L02	L01	
Total	Time(h)	Total	Total	Total	Total	Total	Total	Total	Total	Total	Total	
77888	95	4824	6579	21456	8602	8430	5452	5612	5606	5661	5666	0

Figur 13: Total produktion samt den individuella produktionen för packmaskinerna.

I resultatdelen av interfacet sammanställs outputen till mer överskådlig information. Här sammanfattas den viktiga datan ifrån outputen.

4.5 Validering

När valideringen gjordes så var det svårt att identifiera exakt hur många lådor företaget tillverkade på en 95 timmars intervall, därmed gjordes en beräkning utifrån hur många pallar som packades i timmen.

- 12st lådor får plats på en pall
- 72st pallar/h
- 95h

Lådor som det verkliga systemet producerar i realtid: $12st \cdot 72st/h \cdot 95h = 82080st$ producerade lådor i det angivna tidsintervallet. Följande bild visar resultatet från simuleringen som gjordes med den insamlade datan.

Total Production	
Total	Time(h)
77888	95

Figur 14: Antalet producerade lådor i simuleringen.

$$82080 - 77888 = 4192$$

$$4192 \div 82080 = 0,05109$$

Detta ger en procentuell felmarginal på **5,1 %** enligt beräkningen ovan.

En visuell validering gjordes, men eftersom produktionen varierade betydligt mer än trott var det svårt att säkerställa om simuleringens beteende var den samma som verkligheten.

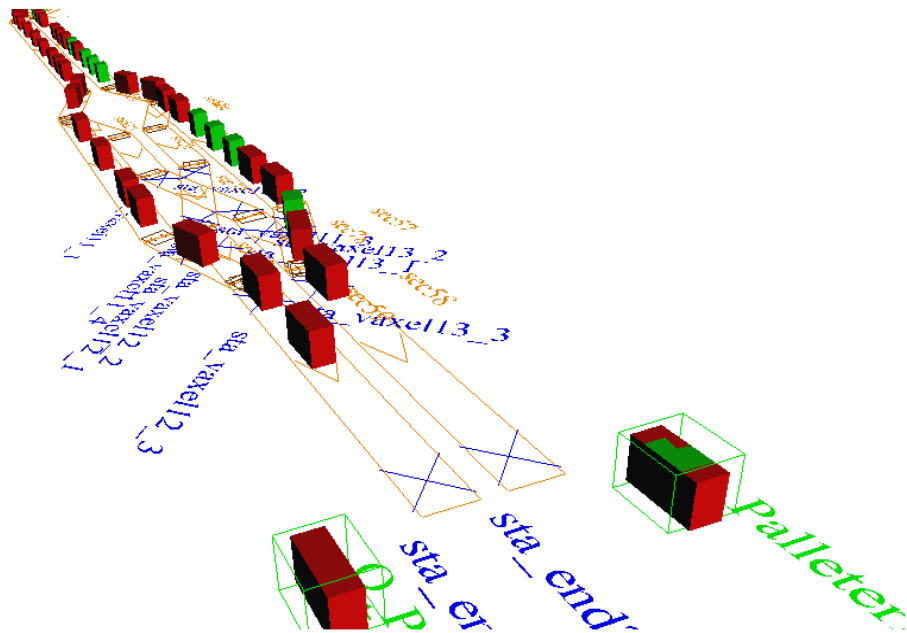
4.6 Kontroll av kapacitet

Vid test av simuleringen användes följande inställningar under körningen:

- Packmaskinshaveriet: OFF
- Palleteringshaveriet: ON, med framtida siffror
- Passningshaveriet: OFF
- Omställningar: OFF

Modellen kördes alltså med fyra faktorer: växlar, hastighet på bandet, kapaciteten för respektive packmaskin samt palleteringshaveriet med dess framtida värden.

Matningshastigheten för packmaskinerna var densamma som tidigare, vilken beskrivs i 4.1.



Figur 15: Test av kapaciteten.

Systemet kördes och visuellt kunde inte några märkbara stopp upptäckas, datan som sedan analyserades baserades på om packmaskinerna var blockerade när de skulle skicka ut lådor. Det visade sig att maskinerna inte var blockerade någon gång, vilket tyder på att transportbandssystemet inte var en begränsande faktor.

Status	Packmaskiner
Fungerar	100%
Blockerad	0%

Tabell 7: Resultatet av kapacitets testet.

Detta betyder att kapaciteten på transportbandet, i en optimal värld utan driftstopp på packmaskiner eller hindrande från palleteringen, överstiger packmaskinernas kapacitet. Detta hypotetiska scenario användes för att kontrollera om transportbandssystemet klarade av packmaskinernas maximala kapacitet, vilket det visade sig att den gjorde.

4.7 Återkoppling till frågeställningarna

- **Vilka driftstopp förekommer i transportbandssystemet och hur kan dessa simuleras på ett verklighetstroget sätt?**

Det förekommer flertalet orsaker till varför det sker driftstopp i transportbandssystemet. Orsakerna kunde spåras till följande fyra felkällor, passningsfel på lådor som orsakade att de fastnade i hålet i väggen, omställningar på packmaskinerna, kapacitetsbrist på palleteringsstationen samt haverier på packmaskinerna. Även fast felmarginalen blev låg **5,1%**, så kan systemet inte ses som helt verklighetstroget. Detta på grund av karaktären på

packmaskinernas haverier, datan gjordes till en triangulär fördelning men en lognormal skulle representerat verkligheten bättre. Därmed stämmer inte simuleringens beteende med det dynamiska beteendet som finns i verkligheten.

- **Hur kan en simuleringsmodell utformas så att den på ett användarvänligt sätt blir användbar för företaget vid kommande investeringsbeslut gällande test av förbättringar?**

När simuleringen fungerar på en acceptabel nivå gäller det att göra den användbar för beställaren, som inte kan arbeta direkt i koden när de ska använda simuleringen. I företags fall vill de kunna ändra bland annat kapaciteter, produktionsintervall och fördelningar. Detta löstes med ett interface där företaget enkelt kan välja och justera de egenskaper som önskats, därmed kan systemet testas i olika scenarion. Då kan framtida investeringars påverkan testas i simuleringen och på så sätt ses det om investeringen är värd att genomföra i verkligheten. Det framtagna interfacet beskrivs detaljerat i kapitel 4.4.

- **Kommer företags transportbandssystem ha tillräcklig kapacitet för att inte bli en flaskhals efter kommande investeringar gjorts i palleteringsområdet?**

I kapitel 4.6 beskrivs det att simuleringsmodellen kördes under sådana förutsättningar att samtliga packmaskiner, som producerade och skickade ut lådor på transportbanden, arbetade med full effektivitet och utan avbrott. Det efterkommande palleteringsområdet arbetade enligt den kapacitet som framtida investeringar kommer resultera i. Detta innebar att transportbandssystemet testades vid maxproduktion av packmaskinerna, utan avbrott. Datan som togs fram av simuleringen visade att samtliga packmaskiner inte varit blockerade, vilket betyder att transportbandssystemet inte varit en flaskhals och har därmed tillräcklig kapacitet även efter investeringarna implementerats.

5. DISKUSSION

Här diskuteras resultatet vilket bryts ner i fem delar, datainsamling, den visuella delen, design av interface, programmeringsdelen och valideringen. Även framtida förbättringar diskuteras. Slutligen diskuteras för och nackdelar med den metodik som använts samt vad som kunde gjorts bättre.

5.1 Resultatdiskussion

Simulering är ett bra verktyg för företag och dess användningsområden är vida spridda, dess fördelar är många och potentialen för implementering inom produktionssektorn är stor. Simulering används idag av flertalet företag runt om i världen, däremot finns det stor potential för vidareutveckling.

Projektets fem huvuddelar var datainsamlingen, den visuella delen, design av interfacet, programmeringsdelen samt valideringen var av varierande komplexitet.

Datainsamlingen visade sig vara mer komplicerad än förväntat. För att få systemet att fungera med god noggrannhet krävdes stora mängder med validerad data, data som var ytterst svår att samla in. Företaget saknade viktig information om transportbandssystemet, vilket gjorde att informationen till stor del samlades in genom fysiska mätningar eller uppskattningar från experter. Data som samlades in till stationshaverierna var byggd på en sammanställning av företaget. Insamling av samtliga individuella stopp hade gjort datan mer verklighetsanpassad samt att en fördelning med hjälp av en lognormal kurva hade beskrivit stoppen bättre, men brist på tid gjorde att detta uteslöts.

Den visuella delen av simuleringen var relativt enkel att konstruera och ritades upp i programmet med god noggrannhet.

För att göra den färdiga produkten användarvänlig var designen av interfacet av stor vikt. I detta fall var konstruktionen av interfacet inte nämnvärt problematiskt, mycket tack vare den färdiga grundmall som erhöles av konsultbolaget. Kodkopplingen till Automod visade sig vara relativt tidskrävande.

Programmeringsdelen var någorlunda komplext då Automod omfattar mängder med olika kommandon, vilka kan användas på många olika sätt för att uppnå de önskvärda resultaten. För att på ett smidigt sätt bygga upp en simulering med detta program krävs stor erfarenhet och en genuin grundläggande kunskap om programmet. Eftersom programmeringsspråket blev inlärt löpande så fanns stora skillnader i kvalitén i koden mellan slut och början, vilket behövde åtgärdas.

För att validera resultatet fick den färdiga simuleringsmodellens resultat jämföras med det verkliga systemets resultat, under förutsättning att de båda arbetar under samma förhållanden. Genom simuleringstest togs resultatet fram att modellen inte är fullständigt verklighetstrogen i de dynamiska rörelserna, utan att in och output snarare ger en viss representation av verkligheten. Värdet i simuleringen jämfördes med är en uppskattning av den totala produktionen på tidsintervallet, och kan variera. Därför bör resultatet av valideringen ses som

en uppskattning. Projektet resulterade i en simuleringsmodell som närmade sig verkligheten med en felmarginal på 5,1%. Tidsbrist resulterade att målet om att bygga en verklighetstrogen modell inte uppnåddes helt. Med fortsatt arbete med simuleringsmodellen skulle detta mål nås.

Mycket av den eftersökta informationen fick tas fram på egen hand, med hjälp av mätningar och uppskattningar. Det fanns därför en viss felmarginal på dessa värden. Denna felmarginal påverkar simuleringens exakthet och verklighetstrogenhet, vilket betyder att det finns en del utrymme för förbättringar. Om mer tid funnits hade noggrannare studier av varje enskilt haveri kunnat fånga systemets verkliga beteende vid haverier mer exakt.

I mån av tid hade simuleringsmodellen kunnat byggas vidare och inkluderat även de angränsande systemen, vilket hade lett till att helheten lättare kunnat beskådas. Vidare hade det blivit lättare att tyda systemets påverkan på de omkringliggande systemen. Detta var dock orimligt att hinna med under den tid som detta projektarbetet omfattade, däremot hade det varit givande för företaget om en vidareutbyggnad på simuleringsmodellen gjorts. Det hade hjälpt med databearbetning på palleteringshaveriet, packmaskinshaverier och vidare arbete med implementeringen av detta resultat i simuleringsmodellen.

5.2 Metoddiskussion

Projektet inleddes genom att göra en planeringsrapport. I planeringsrapporten definierades projektets milstolpar och aktiviteter. Dessa milstolpar, som krävdes för att uppnå målen med projektet, sattes upp enligt uppskattningsvis rimliga tidsramar då projektmedlemmarna inte var fullt informerade om projektets omfattning. Projektarbetets syfte och mål var inte heller helt definierat, så planeringsrapporten var till stor del uppskattad och därmed avvek det verkliga händelseförloppet något från planeringsrapporten.

Under projektets gång gjordes flertalet fabriksbesök på företaget, vilket var mycket givande på flera sätt. Besöken gav en grundläggande förståelse om hur systemet fungerar i verkligheten.

Fördelarna med att följa Banks metod var att det gav projektet en realistisk plan där återkoppling, omarbete och datainsamling skedde. Banks metod lägger fokus på att hela tiden se om simuleringen liknar verkligheten genom återkoppling. Verkligheten var inte ideal och majoriteten av datan som samlades in behövde förenklas.

Metoderna och dess ordning hade vissa nackdelar under arbetes gång. Användbarheten av interfacet utnyttjades inte till fullo under utvecklingsfasen av koden. Bristande kommunikation med företaget gjorde även så att viktig information kom in för sent för att analyseras.

Vid framtida projekt vore en mer utökad granskning av liknande projekts metodik varit gynnande. Detta för att få en bättre förståelse hur man kunde använt interfacet för felsökning. Gruppen hade även lagt mindre tid på Automodupplärning i början som i slutändan ej användes. Istället kunde denna tid använts för bättre planering.

6. MILJÖ

Miljön har alltid varit en av huvudanledningarna till simuleringens uppkomst. Då simulering påverkar användandet av resurser, kan företag både dra ner på omkostnader och mängden utsläpp. Inom simulering av produktion så kan livscykeln av en produkt analyseras samt även dess utsläpp. Tack vare simulering kan det ses hur en ändring inom produktionen kan påverka systemets totala utsläpp. Oftast kan en maskin i systemet ses som en station där exempelvis material kommer in och förbrukar en viss mängd energi samt resulterar i en viss mängd utsläpp. Genom justering av dessa parametrar kan miljöpåverkan variera och med prövning kan en så låg utsläppsnivå som möjligt nås (Thiede, Seow, Andersson, & Johansson, 2013 s. 2 & 9).

Simuleringen som byggts visar transportbandssystemets förmåga att hantera varierande produktionsvolymen samt hur dessa påverkar systemets effektivitet. Ett effektivt system använder mindre resurser och på så sätt belastas miljön i mindre utsträckning. Genom att ge företaget en inblick över hur deras investering i palleteringen kommer påverka systemet, kan ett miljösmart beslut väljas från företagets sida. Detta miljösmarta alternativ kan innebära att de behöver köra produktionen färre timmar samtidigt som samma produktionsvolym uppnås, vilket kräver mindre energi och därmed mindre utsläpp.

Stödet för miljö-integrerat tänkande i simuleringmjukvaror är begränsat, vilket är problematiskt när det kommer till miljöfokuserad simulering. Framtiden för dessa system är lovande om mjukvarutillverkarna öppnar upp för ett mer integrerat miljötänk. Med hjälp av de tekniska förutsättningarna som programmen utger kan en simulering bli mer heltäckande och inkluderande av miljöaspekter i produktionsföretag redan vid planeringsfasen (Thiede et al., 2013, s. 9).

7. SLUTSATS

Projektets resultat blev en interaktiv simuleringsmodell där företagets driftstopp kartlades, kapaciteten testades och framtida investeringsbeslut analyserades.

Frågeställningarna och dess lösningar redovisas nedan:

1. Vilka driftstopp förekommer i transportbandssystemet och hur kan dessa simuleras på ett verklighetstroget sätt?
2. Hur kan en simuleringsmodell utformas så att den på ett användarvänligt sätt blir användbar för företaget vid kommande investeringsbeslut gällande test av förbättringar?
3. Kommer företagets transportbandssystem ha tillräcklig kapacitet för att inte bli en flaskhals efter kommande investeringar gjorts i palleteringsområdet?

Svar på frågorna:

1. Fyra felkällor kunde identifieras, passningsfel på lådor som orsakade att de fastnade i hålet i väggen, omställningar på packmaskinerna, kapacitetsbrist på palleteringsstationen samt haverier på packmaskinerna. Felmarginalen blev **5,1%**, dock följdes ej det dynamiska beteendet som finns i verkligheten.
2. Detta löstes med ett interface där företaget enkelt kan välja och justera de egenskaper som önskats, därmed kan systemet testas i olika scenarion.
3. Transportbandssystemet testades vid maxproduktion av packmaskinerna, utan avbrott. Simuleringen visade att samtliga packmaskiner inte varit blockerade, vilket betyder att transportbandssystemet inte varit en flaskhals och har därmed tillräcklig kapacitet även efter investeringarna implementerats.

Företaget kan ha stor användning av att simuleringar för att testa förbättringar då kraven för flexibilitet ökar allt mer. Simulering är ett bra verktyg för att testa förbättringar, se ett helhetsperspektiv samt att förbättringar kan analyseras utan att avbryta produktionen.

Sammanfattningsvis konstaterades att datainsamling var det svåraste momentet men att även kunskap inom Automod-programmering hade underlättat och minskat uppbyggnadstiden. Vilket hade givit tid för ett mer avancerat resultat.

Examensarbetet har resulterat i en helhetsbild av hur arbetsgången med simulering ser ut samt dess projektuppbyggnad, kodning och felsökning av kod. Vid fortsatt arbete hade data kunnat omstruktureras och förbättras. Vidare hade interfacet gjorts mer användarvänlig och systemet byggts ut. Simuleringen har stor potential för vidareutveckling med ett brett användningsområde.

KÄLLFÖRTECKNING

Banks, J. (2004). *Getting Started with AutoMod, Second Edition*. Utah: Brooks Automation, Inc.

Chacon, S., & Straub, B. (2019). *Pro Git. Second Edition*. Retrieved from: <https://git-scm.com/book/en/v2>

Hågeryd, L., Björklund, S., & Lenner, M. (2002). *Modern produktionsteknik. D. 1*. Stockholm: Liber.

Heilala, J. (1999). *Use of simulation in manufacturing and logistics systems planning*. Retrieved from: https://www.researchgate.net/publication/228359022_Use_of_simulation_in_manufacturing_and_logistics_systems_planning

Sargent, R. G. (2011). *VERIFICATION AND VALIDATION OF SIMULATION MODELS*. Retrieved from: <https://www.informs-sim.org/wsc11papers/016.pdf>

Skoogh, A., & Johansson, B. (2008). *A METHODOLOGY FOR INPUT DATA MANAGEMENT IN DISCRETE EVENT SIMULATION PROJECTS*. Retrieved from: http://publications.lib.chalmers.se/records/fulltext/79631/local_79631.pdf

Thiede, S., Seow, Y., Andersson, J., & Johansson, B. (2013). *Environmental aspects in manufacturing system modelling and simulation-State of the art and research perspectives*. Retrieved from: <https://doi.org/10.1016/j.cirpj.2012.10.004>

Bilagor

Bilaga 1

Läsa in filer från interfacet, exempel med omställning.

```
344
345 | open "input/Changeover_type.txt" for reading save result as UI_VFilePtr // Öppnar filen som skall läsas in.
346 |
347 | set VI_Temp(1) = 1
348 | while VI_Temp(1) < 12 do begin // första while loopen lägger in 11 värden till det första värdet i VI_ProLim.
349 | | set VI_Temp(2) = 1
350 | | while VI_Temp(2) < 7 do begin // Den andra while loopen lägger in 6 värden till det andra värdet i VI_ProLim.
351 | |
352 | // VI_ProLim håller vilken packmaskin och de nästkommande 6 omställningsvärdena för den packmaskinen. Även vilken slags o
353 | | read VI_ProLim(VI_Temp(1), VI_Temp(2), ORDERVOL), VS_Temp(2) from UI_VFilePtr with delimiter "\t"
354 | |
355 | | // kollar om omställningen är kort eller lång.
356 | | set VI_LongReconfig(1) = 60 // långt stopp definieras
357 | | set VI_ShortReconfig(1) = 15 // kort stopp definieras
358 | | // kollar om omställningen är kort, lång eller inget värde.
359 | | if VS_Temp(2) = "Long" then
360 | | | set VI_ProLim(VI_Temp(1), VI_Temp(2), SETUPTYPE) = VI_LongReconfig(1)
361 | | | else if VS_Temp(2) = "Short" then
362 | | | | set VI_ProLim(VI_Temp(1), VI_Temp(2), SETUPTYPE) = VI_ShortReconfig(1)
363 | | | else
364 | | | | set VI_ProLim(VI_Temp(1), VI_Temp(2), SETUPTYPE) = 0
365 | | | inc VI_Temp(2) by 1
366 | | end
367 | | inc VI_Temp(1) by 1
368 | end
369 |
```

Fördelningsfunktionen

```
844 //Argumet4 Arg_Parameter3: Parameter 3
845 //Function that returns a random time for the given Distribution and parameters
846 begin F_Distribution
847 | if Arg_Distrubution = "Triangular" or Arg_Distrubution = "triangular" then begin
848 | | return triangular Arg_Parameter1, Arg_Parameter2, Arg_Parameter3
849 | end
850 | else if Arg_Distrubution = "Lognormal" or Arg_Distrubution = "lognormal" then begin
851 | | return Arg_Parameter3 + lognormal Arg_Parameter1, Arg_Parameter2
852 | end
853 | else if Arg_Distrubution = "Normal" or Arg_Distrubution = "normal" then begin
854 | | return Arg_Parameter3 + normal Arg_Parameter1, Arg_Parameter2
855 | end
856 | else if Arg_Distrubution = "Weibull" or Arg_Distrubution = "weibull" then begin
857 | | return Arg_Parameter3 + weibull Arg_Parameter1, Arg_Parameter2
858 | end
859 | else if Arg_Distrubution = "Uniform" or Arg_Distrubution = "uniform" then begin
860 | | return uniform Arg_Parameter1, Arg_Parameter2
861 | end
862 | else if Arg_Distrubution = "Gamma" or Arg_Distrubution = "gamma" then begin
863 | | return Arg_Parameter3 + gamma Arg_Parameter1, Arg_Parameter2
864 | end
865 | else if Arg_Distrubution = "Exponential" or Arg_Distrubution = "exponential" then begin
866 | | return Arg_Parameter2 + Arg_Parameter3 + exponential Arg_Parameter1
867 | end
868 | else if Arg_Distrubution = "Constant" or Arg_Distrubution = "constant" then begin
869 | | return Arg_Parameter1
870 | end
871 | else begin
872 | | print "Function " Arg_Distrubution " not found!" to message
873 | | return 0
874 | end
875 | end
```

Packmaskinshaveri

```
766 | begin P_Breakdown_Station arriving
767 |   while 1=1 do begin
768 |     // väntar mellan haverier med en tid definierad i interfacen som MTTF.
769 |     use R_BreakdownStation(AI_LineNumber) for F_Distribution(VS_StationDistsType(AI_StationIndex(1) , 1), VI_BreakDown)
770 |
771 |     // Om palleterings problem är aktivt ska haveriet registreras när detta är klart.
772 |     if VI_PalleteringsFlagga(AI_Line1) = 1 then
773 |       wait until VI_PalleteringsFlagga(AI_Line1) = 0
774 |     // Tar ner huvudresursen och stannar maskinen och därmed registreras problemet i datan.
775 |     take down R_Line(AI_LineNumber)
776 |
777 |     // Väntar till reparation med en tid definierad i interfacen som MTTR.
778 |     wait for F_Distribution(VS_StationDistsType(AI_StationIndex(4) , 2), VI_BreakDownMTTR(AI_StationIndex(4), 1), VI_BreakDown)
779 |     // Tar upp maskinen och kollar då om nått annat fel är aktivt. Om detta ej är fallet körs maskinen som vanligt.
780 |     bring up R_Line(AI_LineNumber)
781 |     if OL_LineOut(AI_LineNumber) current > 0 then begin
782 |       if VI_PalleteringsFlagga(AI_Line1) = 1 then
783 |         set R_Line(AI_LineNumber) active state = Palleteringsfel
784 |       else if VI_BreakdownHole(AI_Line1) = 1 then
785 |         set R_Line(AI_LineNumber) active state = Breakdown_Hole
786 |       else if VI_CleaningFlagga(AI_LineNumber) = 1 then
787 |         set R_Line(AI_LineNumber) active state = Cleaning
788 |       else
789 |         set R_Line(AI_LineNumber) active state = Blocked
790 |     end
791 |     else
792 |       set R_Line(AI_LineNumber) active state = Idle
793 |     end
794 |   end
795 | end
```

Palleteringshaveri

```
begin P_PalleteringOrder1 arriving
  while 1=1 do begin
    wait for 60/ VI_PalleteringsLimit(1, 1) sec
    order 1 load from OL_Palletering1 to die
  end
end

begin P_PalleteringOrder2 arriving
  while 1=1 do begin
    wait for 60/ VI_PalleteringsLimit(1, 2) sec
    order 1 load from OL_Palletering2 to die
  end
end

begin P_Palletering arriving
  if Q_Palletering_Bana(AI_Line1) current = VI_QueueLimit(AI_Line1) then begin
    if VI_PalleteringsFlagga(AI_Line1) = 0 then begin
      inc VI_PalleteringsFlagga(AI_Line1) by 1
      print "Palleterings fel bana " AI_Line1 to message
      clone 1 load to P_Breakdown_Palletering
    end
  end
  // Annars så går lådorna in i respektive kö.
  else begin
    if AI_Line1 = 1 then begin
      move into Q_Palletering_Bana(AI_Line1)
      wait to be ordered on OL_Palletering1
    end
    if AI_Line1 = 2 then begin
      move into Q_Palletering_Bana(AI_Line1)
      wait to be ordered on OL_Palletering2
    end
  end
end
```

```

742 | begin P_Breakdown_Palletering arriving
743 |   if AI_Line1 = 1 then begin
744 |     clone 1 load to P_Breakdown_Bana1
745 |     while Q_Palletering_Bana(AI_Line1) current >= VI_QueueRecovery(AI_Line1) do begin
746 |       wait for 5 sec
747 |     end
748 |     wait for VI_PalleteringsWait(1) sec
749 |     clone 1 load to P_Bringup_Bana1
750 |   end
751 |   if AI_Line1 = 2 then begin
752 |     clone 1 load to P_Breakdown_Bana2
753 |
754 |     while Q_Palletering_Bana(AI_Line1) current >= VI_QueueRecovery(AI_Line1) do begin
755 |       wait for 5 sec
756 |     end
757 |     wait for VI_PalleteringsWait(2) sec
758 |     clone 1 load to P_Bringup_Bana2
759 |   end
760 |   send to die
761 | end
762 |

```

Passningshaveri

```

598 | begin P_Breakdown_Hole arriving
599 |   while 1=1 do begin
600 |     if AI_Line1 = 1 then begin
601 |       wait for F_Distribution(VS_StationDistsType(AI_StationIndex(5), 3), VI_BreakDownMTBF(AI_StationIndex(5), 4), VI
602 |       if VI_PalleteringsFlagga(1) = 1 then
603 |         wait to be ordered on OL_BreakdownHole(1)
604 |       clone 1 load to P_Breakdown_Bana1
605 |       set VI_BreakdownHole(1) = 1
606 |       wait for F_Distribution(VS_StationDistsType(AI_StationIndex(2), 4), VI_BreakDownMTTR(AI_StationIndex(2), 4), VI
607 |       clone 1 load to P_Bringup_Bana1
608 |     end
609 |     if AI_Line1 = 2 then begin
610 |       wait for F_Distribution(VS_StationDistsType(AI_StationIndex(5), 3), VI_BreakDownMTBF(AI_StationIndex(5), 4), VI
611 |       if VI_PalleteringsFlagga(2) = 1 then
612 |         wait to be ordered on OL_BreakdownHole(2)
613 |       clone 1 load to P_Breakdown_Bana2
614 |       set VI_BreakdownHole(2) = 1
615 |       wait for F_Distribution(VS_StationDistsType(AI_StationIndex(2), 4), VI_BreakDownMTTR(AI_StationIndex(2), 4), VI
616 |       clone 1 load to P_Bringup_Bana2
617 |     end
618 |   end
619 | end
620 |

```

Omställning

```

498 | clone V_BoxQ(AI_LineNumber) loads to P_Start
499 | inc VI_LoadSent(AI_LineNumber) by V_BoxQ(AI_LineNumber)
500 | if VI_ProLim(AI_LineNumber, VI_OrderNr(AI_LineNumber), ORDERVOL) <= VI_LoadSent(AI_LineNumber) then begin
501 |   set VI_LoadSent(AI_LineNumber) = 0
502 |   while VLocPtr_Start(AI_LineNumber) current > 0 do begin
503 |     wait to be ordered on OL_Cleaning(AI_LineNumber)
504 |   end
505 |   take down R_BreakdownStation(AI_LineNumber) //Den hindrar packmaskinernas haveri från att inträffa under omstäl
506 |   set R_Line(AI_LineNumber) active state = Cleaning // sätter status till omställning.
507 |   set VI_CleaningFlagga(AI_LineNumber) = 1 // flaggar om att omställningen är nere.
508 |   wait for VI_ProLim(AI_LineNumber, VI_OrderNr(AI_LineNumber), SETUPTYPE) min // väntar tills omställning är klar
509 |   bring up R_BreakdownStation(AI_LineNumber) // Hindrar inte längre packmaskinernas haveri.
510 |   set VI_CleaningFlagga(AI_LineNumber) = 0
511 |   inc VI_OrderNr(AI_LineNumber) by 1 // läser in nästa värde från omställning.
512 | end

```