NTNU
Norwegian University of
Science and Technology

Discrete-Event Simulation of a Multimodal Downstream Supply Chain for Future Norwegian Aquaculture

Ronja Eide Lilienthal
Ragni Rørtveit

Marine Technology
Submission date: June 2017
Supervisor: Bjørn Egil Asbjørnslett, IMT

Norwegian University of Science and Technology
Department of Marine Technology

Summary

The aim of this thesis is to model logistics for the downstream aquaculture supply chain interaction for increased production of salmon in the future. We employed a quantitative approach using discrete-event simulation, where we developed a model, replicating the real-world supply chain in MATLAB's add-on software Simulink. The output was basis for evaluations regarding system dynamics and flexibility, based on composition and throughput. Project limitations excluded human influence, as well as all commercial aspects, from the analyses.

The motivation for the thesis was the Norwegian governments goal to increase aquaculture production by five-fold within 2050. Following this increase comes the development of farm sizes and locations, as well as an adapted supply chain. Such an expansion may include the introduction of additional transportation modes, like processing vessels and seaborne transportation of products to market.

As a multimodal system, the challenge lies in coordinating the various modes, pursuing seamless integration between them. This is effectively assessed using discrete-event simulation, modelling the continuous movement of entities.

The model was run several times with different system compositions and scenarios. These served to verify the model, establish a benchmarking fleet and test system robustness against external influence. Two crash scenarios, simulating emergency slaughter and waiting cage ban, were imposed.

The results showed that delivery to market using the current day trucking system is efficient and reliable. However, a simulated expansion required a large fleet, which the introduction of cargo vessels could relieve. Processing vessels withheld transit-time to a minimum regardless of imposed scenarios, and were a contribution to continuity in operation.

In conclusion, we showed that any system upgrade must be supported by equivalent infrastructure to increase system performance. Collaboration and redundancy through parallel nodes were key factors in ensuring system up-time. Further work should include evaluations on commercial competitiveness.

Sammendrag

Målet med denne masteroppgaven er å modellere sammenhenger i logistikken til en nedstrøms verdikjede for norsk havbruk ved fremtidig økt produksjon. Vi brukte diskret hendelsessimulering i MATLABs programvare Simulink for å utvikle en modell som gjenspeilet virkeligheten best mulig. Resultatene fra denne dannet grunnlaget for å gjøre vurderinger om systemets samspill og fleksibilitet, basert på gjennomstrømning og oppbygning. Prosjektet begrenset seg til praktiske aspekter, og kommersielle og menneskelige hensyn var følgelig ekskludert.

Innen 2050 har den norske regjeringen satt et mål om å femdoble produksjonen fra norsk havbruk. Denne utviklingen vil kreve en utvikling av lokasjoner og merder, i tillegg til en tilpasset verdikjede. Det kan også bli aktuelt å introdusere nye transportmoder, som slakteskip og frakt av produkter på kjøl til Europa.

Som et multimodalt system ligger utfordringen i å få overgangene mellom forskjellige transportetapper og prosesser til å gå så strømlinjeformet som mulig. Det var dette som ble vurdert gjennom simuleringen, hvor bevegelsen av entiteter gjennom verdikjeden ble vurdert kontinuerlig. Modellen ble kjørt iterativt med forskjellige oppbygninger og scenarioer, derav to krasj-scenarioer; nødslakt og ventemerdforbud. Hensikten var å verifisere modellen, etablere en sammenligningsflåte, og teste systemets motstandsdyktighet mot ytre innflytelser.

Resultatene viste at transport til marked med dagens system er effektivt og pålitelig. Ved femdobling av produksjonsvolum ble derimot en parallell sjøveis transport vist effektivt mhp. tidsbruk og belastning av veinettet. Prosesskipene gjorde at total tid til marked ble holdt minimal, og bidro til kontinuerlig drift under krasj-scenarioer.

Samlet har prosjektet vist at alle systemoppgraderinger må være støttet av tilsvarende bearbeiding av infrastruktur for å kunne utgjøre en forskjell. Samarbeid og redundans gjennom parallelle produktstrømmer var viktige faktorer i å opprettholde systemets oppetid. Videre arbeid bør sikte på å inkludere kommersielle aspekter.

Preface

This thesis is submitted to the Department of Marine Technology as the final fulfillment of the requirements for the degree of Master of Science (M.Sc.). The work has been carried out from January 2017 to June 2017 under the department of Marine Technology, Norwegian University of Science and Technology, Trondheim. The work serves as a contribution to the SFI EXPOSED program, and the thesis description is submitted as an attachment to this paper.

The motivation for this study is the increasing focus on Norwegian aquaculture development. The thesis is a continuation of the project thesis written the fall of 2016, and aims to serve as a feasibility study, using simulation as a tool to give intel into the downstream supply chain interactions, and subsequently unveil potential future challenges facing Norwegian aquaculture.

We are highly indebted to our main supervisor Professor Bjørn Egil Asbjørnslett at the Department of Marine Technology, Norwegian University of Science and Technology, who has meticulously guided us through the work. We would also like to thank Svein Knudtzon Waagbø at Møre Maritime AS for help with vessel specification data, and Guri Rørtveit who has functioned as an academic consultant.

Ragni Rørtveit — Place and Date

Ronja Eide Lilienthal — Place and Date

Table of Contents

List of Tables

List of Figures

List of Equations

Abbreviations

CC / CV = Cargo Carrier / Cargo Vessel
DWT = Dead Weight Tonne
FSA = Food Safety Authority
HSE = Health, Safety, Environment
IQR = Interquartile Range
KNH = Kristiansund and Nordmøre Port Authority
KPI = Key Performance Indicator
LFC = Live Fish Carrier
MAB = Maximum Allowed Biomass
NCA = Norwegian Coastal Authority
PV / Proc = Processing Vessel
RSW = Refrigerated Seawater
SD / STD = Standard Deviation
SDI = Simulation Data Inspector
SFI = Senter for Forskningsdrevet Innovasjon

CHAPTER 1

Introduction

1.1 Background

This master thesis is a continuation of a preliminary project thesis, written in the fall of 2016, which used simulation to gain an understanding of the complexity and interactions in the downstream aquaculture supply chain. The downstream system includes every part of the supply chain after the salmon is fully grown, but disregards everything leading up to it. Initially, the project presented how Norway has unique natural advantages for aquaculture production with its sheltered fjords and long coastline. According to SINTEF, Norwegian aquaculture farmers have the ability to produce 5 million tonnes of fish per year by 2050, compared to the 1 million tonnes in 2010, if today's challenges related to production and environment are solved (SFI EXPOSED and SINTEF Ocean, 2015; Olafsen et al., 2012). The SINTEF report "Value created from productive oceans in 2050", emphasizes that the industry, by managing its resources properly, focusing on education and research, and providing our industries with good and predictable framework conditions, could raise the potential value in the marine sector from approximately 80 billion NOK in 2010, to 550 billion NOK in 2050 (Olafsen et al., 2012).

Consequently, the Norwegian government has posed an ambitious goal to increase production by five-fold (Nærings- og Fiskeridepartementet, 2015). As presented in the aforementioned project thesis, will this bring new challenges; locate sufficient farm locations, ensure effective and gentle salmon transportation to shore, and development of on-shore infrastructure (Rørtveit and Lilienthal, 2016). This includes both factories and transportation to market, which today is performed by truck directly to the consumers. Salmon is fresh produce, and time is of essence. With a tight schedule and large production volumes come the need for constant high system efficiency and large production- and transportation capabilities.

The industry is currently at a point of no expansion due to strict government regulations imposed in order to control problems related to disease and eco-pollution (Olafsen et al., 2012). However, the willingness to experiment is at an all-time high, proportional to recent salmon prices, and it is thus assumed that the technological breakthrough is imminent. If/when this happens, the logistics infrastructure has to be ready for a rapid expansion. It is this new market this master thesis aims to comprehend.

Fish farmers who have gradually moved their production to more exposed sites in order to position themselves against industry development, report on significant difficulties in maintaining a reliable production, due to unreliable weather, wave and current conditions, and increased distance to the on-shore facilities (SFI EXPOSED and SINTEF Ocean, 2015). To meet these challenges, and possible future regulations regarding live fish transportation and fillet quality, the industry has been conducting trials where the fish is slaughtered directly from cage, prior to transit, and cooled on refrigerated seawater (RSW), before further transportation to shore (Mindling et al., 2011). Research results showed that harvesting- and processing vessels could represent an addition as support vessels for the fish farming industry in the future.

Today, most of Norwegian salmon is exported by truck across the mainland. When interviewed on 05 October 2016, Sigurd Bjørgo, special advisor for Sør-Trøndelag county council, informed that logistics at the factory infrastructure, and strain on narrow Norwegian roadways, will at some point be expected to reduce the efficiency of truck transport. Thus, significantly greater harvests can make seaborne or railroad transport solutions more relevant. The authors of the report, "Sustainable sea transport solutions for fresh salmon exports from Mid Norway to Continental Europe" claim that the production volume in Mid Norway already is large enough to justify seaborne export of salmon (Bjørshol et al., 2015). However, the current delivery system, where finished and custom-made products are delivered fast and directly to the end consumer, has become a part of the industry trademark. Sven Amund Fjeldvær, CEO for Lerøy Midt AS, stated during a private interview on 6 October 2016, that several farming companies have been skeptical regarding bringing cargo vessels into the supply chain. This is primarily due to fear of delay and slow loading/unloading processes.

There are many reasons speaking to moving cargo transport from land to sea, the most prominent being the large capacity per vessel. The challenge rests in the same feature; to fill a vessel, large volumes have to be ready at the same time. This is insignificant for preserved produce, as the cargo can safely wait in storage for an available vessel. Fresh seafood, however, imposes strict demands on the transportation, both regarding time, quality and safety. It is therefore important that a new shuttle service in addition to satisfying requirements for cost and time also satisfies the market's expectations regarding quality and delivery. Bjørshol et al. (2015) state that market requirements, data availability and a competitive relationship between the manufacturers additionally challenges cooperation opportunities.

It is also important that new supply chain solutions are flexible, so that minor changes in the schemes may be permitted and adapted without the basic intentions being sacrificed. Whereas the early industry could grow gradually, the modern world calls for custom-fit, complex integration. Thus, the desire to model or predetermine outcomes and interactions has grown. Discrete event simulation has gained popularity in supply chain modelling, as it allows for time-dependent tracking. In this report, simulation was used to understand flow throughout the entire supply chain. This knowledge is essential in a time of considerable industry growth; as upstream development (until the fish reaches its ideal weight), is cheaper than its downstream counterparts.

1.2 State of the Art

This thesis seeks to study the complexity and interaction between the various elements in an aquaculture supply chain. There are a series of approaches related to the challenge of analyzing a multimodal system, but the large-scale aquaculture problem has largely been left alone from scientific efforts. Recently, however, the industry has seen, and shown, an increased interest in using research to acquire a deeper system understanding. This is related to increased profitability, availability from the traditional offshore market, and ambitions to utilize marine knowledge (Fon, 2017).

According to Pawlewsk (2013), is a multimodal transport system defined as ”an internally integrated system of carrying goods along with accompanying services provided with use of at least two modes of transport on the basis of a multimodal transport contract”. As organizations want their supply chains to be efficient, fast, agile, custom-configured, and flexible, all at the same time, challenges related to this combined contract and need for actor collaboration has become more prominent. This has led the transportation of goods to become widely global (Srai and Gregory, 2008).

The globalization is driven by the hunt for high efficiency and speed. In order to adapt to this new demand for high efficiency systems, containerization has become a vital link in the development. Containers allow producers to sort and organize their own products, while transporters are distanced from the product itself by only handling standardized units. This is reflected in the exponential growth in container traffic, increasing from 15 million TEUs in 1995, to almost 60 million in 2008 (Rodrigue and Notteboom, 2013).

It is also, as described by Kant et al. (2016), with an aim to increase the freight efficiency when cooperation between carrier companies is initiated. The other effects of following a combined contract are largely negative, such as increased risk and possibilities for prolonged warehouse stops. Consequently is one of the more prominent challenges with multimodal systems the increased risk of system failure. By depending upon various companies, policies and equipment, a multimodal supply chain becomes exposed to breakdowns. This was emphasized by Vilko and Hallikas (2012) in their report on risk management in multimodal maritime supply chains, where it was unveiled how differently risk is managed. They particularly found differences between larger and smaller companies, a combination which is almost inevitable in a Norwegian aquaculture supply chain.

Kant et al. (2016) also emphasize the role of the government in shaping the development of future supply chains. This is evident through incentives and taxing, and will be influential in tipping the scale in choice of transportation means. The aim in choosing supply chain systems is to make it cheap and effective, and the government is able to make publicly favorable decisions economically favorable. In Norway has the government chosen an active approach, openly stating support for expansion of the aquaculture industry (Nærings- og Fiskeridepartementet, 2015).

Improving supply chain performance is a continuous process requiring an analytical performance measurement system. In the article “Advanced traceability system in aquaculture supply chain”, Vilko and Hallikas (2012) elaborates how product control, work organisation, time management and customer confidence can be improved by evaluating a set of key performance indicators (KPIs) relative to a set of predetermined objectives. The challenge, however, lies in analyzing and selecting the right KPI groups and strategies for improved supply chain performance. This is particularly prominent for companies who have continuously changing strategic objectives, and need to meet requirements of a dynamic decision-making environment (Caia et al., 2009).

1.3 Objective

This master's thesis aimed to serve as a feasibility study, using discrete - event simulation as a tool to give intel into the downstream supply chain interaction, and subsequently unveil potential challenges facing Norwegian aquaculture in the future. The thesis was a continuation of the authors' preliminary project thesis, which emphasized the flow to and from the slaughter- and processing facilities. This project continued the preceding work by expanding to the complete multimodal downstream supply chain composition, evaluating system dynamics and flexibility based on composition and throughput.

Modelling a continuous movement of entities made it possible to locate system weak links, as well as evaluate whether new transportation modes provided added value to the supply chain. The goal of this project was not to find an optimal solution, but rather to provide an understanding of the system as a whole, and as such be useful when the industry expands.

1.4 Project Structure

In order to structure the project, the workload was divided into three parts. Together these made up the process leading up to this report.

Part 1 - Obtaining Background Information

Initially, the focus was to achieve a better understanding of possible future developments of the downstream supply chain by accessing real-life data. The information was obtained from article- and report research within NTNU's database Oria, relevant web pages, as well as personal communication with experts and aquaculture stakeholders. Knowledge and experience from the project thesis was also brought into the continued work.

Part 2 - Simulation Model Construction

Once sufficient information was gathered, the discrete event simulation model was built, as a multimodal simulation of entities in MATLAB's Simulink. It was important to build a solid model which replicated the real world system as close as possible, and was able to vary with input. In order to be useful and intuitive, was it desired to develop a generic and user-friendly model. This included for instance weather data, vessel specifications, and segmentation and distribution of end products. Success factors for the simulation was subsequently a properly functioning model which carried the cargo through the system's many joints.

Part 3 - Assessment and Evaluation of Results

The final part comprised running the simulations, and assessing and validating results in compliance with the objectives. Simulink is able to produce a series of outputs, both tabular and graphic, so the key element was to find the output which provides valuable intel. The output should illustrate the movement of entities, both representing vessels, trucks and cargo, but also the utilization of infrastructure. System dynamics, vulnerability and flexibility is evaluated with respect to throughput and composition for the various imposed events.

CHAPTER 2

System Description

The aquaculture supply chain from farm to market makes out the system of interest for this thesis, and will be further described in the following chapter. This system will be considered both in regard to the way it is operated today, and which challenges and alterations can be expected in the future.

2.1 Supply Chain Boundaries

The overall supply chain purpose is to serve its customers. As defined by Chandra and Grabis (2016), “a supply chain is a network of supply chain units collaborating in transforming raw materials into finished products to serve common end-customers”. The supply chain is built up by various elements like storage, transport and servicing. These have different purposes in order to play their respective part in developing and moving products. In addition to the physical features of a supply chain, are there a series of abstract system properties, like interrelationships, system boundaries, purpose, environment, input, output and constraints.

Together, these make out the framework and driving forces of the supply chain, and the system can as such be perceived as a social-technical system, see Figure 2.1.

Elements which are considered to be a part of the system are defined within the system boundaries, while the environment will be the supply chain's competitive environment. This includes actors who influence the system, but are otherwise disregarded, and include suppliers, competitors, official authorities etc. The purpose of a supply chain analysis is thus to analyze the influence the environment has on the elements in the system, but not vice versa.

The supply chain constraints are divided into network-wide constraints and unit-wide constraints, where the first mainly defines global operating requirements such as regional differences, legal requirements etc., see Figure 2.1. The second defines local operational requirements, such as allocation of resources and capacities (Chandra and Grabis, 2016).

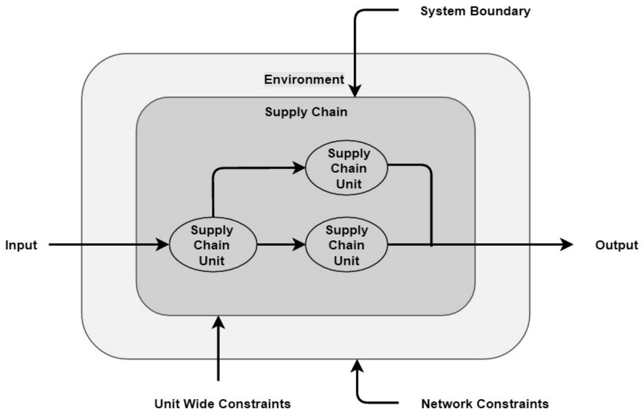


Figure 2.1: Schematic display of a social-technical supply chain system, and its interfaces.

In this paper, the production system will be limited to technical structures, including its interfaces and relations with the environment. All human factors are thus disregarded. The environment will consist of the geographical area, customers, weather and some authority influences like operation restrictions regarding fish welfare. The infrastructure part of the system will primarily comprise all aspects regarding the physical movement of products, but disregard all telecommunication and external information services and networks.

The main supply chain factor is customer demand. The customers are displayed as final nodes in the supply chain network, and thus within the system boundaries. However, this representation will only concern their physical location, while their logical behavior is external to the analysis. Inputs are materials and services provided by supply chain units outside the scope, like fish production and environmental impacts, while outputs are products delivered to customers and general system performance indicators.

2.2 The Downstream Aquaculture Supply Chain

The downstream fish farming supply chain comprises all elements related to the production of salmon, after it has reached full growth. As presented in the preliminary project thesis, Chapter 2, the supply chain is complex, comprising several nodal- and modal supply chain units, linked together in multiple ways forming a network structure (Rørtveit and Lilienthal, 2016). The nodal parts of the infrastructure are the farms, warehouses, ports and slaughtering- and processing facilities. The modal segments of the supply chain are the transportation modes, e.g. road and waterborne transport. The supply chain units are linked together in a way which imposes the movement of materials and products from their initial state to the final state in the form of end-products (Chandra and Grabis, 2016). A sketch of the most important elements in a typical aquaculture chain composition can be seen in Figure 2.2.

Figure 2.2: Today's supply chain for farmed salmon.

The following sections will briefly describe the traditional Norwegian downstream aquaculture supply chain, and is based on Chapter 2 in the aforementioned project thesis (Rørtveit and Lilienthal, 2016).

Commercial net pens today are allowed to hold up to 200 000 individual fish, weighing no more than 780 tonnes in total (945 in the two northernmost counties) (Marine Harvest, 2016). This is regulated by the government, and called Maximum Allowed Biomass (MAB). Upon acquisition, a fish carrier loads fish through a submerged pipe in the cage, while the net volume is simultaneously being reduced by lifting the sides up. This forces the salmon to swim through the large pipes, and into the fish carrier.

When interviewed on March 3^{rd} 2017, Svein Knudtzon Waagbø, general manager for Møre Maritime AS, informed that the newest generation of fish carriers have the capacity to carry approximately 1000 tonnes, i.e. more than one MAB. These vessels carry out the transportation from the fish farm to the slaughtering facility, all the while keeping the fish alive by providing oxygen and adequate living conditions. The movements of the vessel, in addition to the loading activities, are unusual to the livestock, and thus cause stress. This stress is transmitted into the muscles, reducing the product quality, and leading to a rapid entering of rigor mortis state after slaughter (Fiskeri- og Havbruksnæringens Landsforening, 2009). Any treatments processing done in this state will result in meat cleavage, which reduces the product quality, and as a direct result of this, the fish is pumped into waiting cages prior to slaughter (Mindling et al., 2011). These are cages located in immediate proximity to the slaughtering facility, in which the fish is allowed to calm down. Before departure from the cages, throughout transit, and during the stay in these cages, the fish is not fed. Sven Amund Fjeldvær informed that this is to avoid unnecessary expenses, as well as prevent the stressed fish from disgorging (pers.comm., 6 October 2016). These cages also serve as buffer for the slaughtering facility, which usually is incapable of processing a full vessel load upon arrival.

After resting for at least twelve hours, the fish are once again forced into a suction tube, this time pumped directly onto the conveyor belt, to be electrocuted. Execution of the slaughtering process in the salmon industry is a complex procedure that comprises fish crowding, pumping and killing; all the while ensuring that the fish is not subjected to high activity- or stress. Recent developments, however, introduces super-cooling and fish filleting immediately after electrocution, which intensifies the importance of gentle handling of the fish to avoid stress (Fiskeri- og Havbruksnæringens Landsforening, 2009).

Upon slaughter and processing, the finished products are sent by plane to the intercontinental markets, and with truck for direct consumption and further processing in Europe. Transport to the Western European market is done primarily by Norwegian and Danish trailers, while transport to the Eastern European markets are mostly handled by Russian and Estonian trucks (Mathisen et al., 2009). Bjørnar Johansen, General manager for Blått Kompetansesenter AS, informed that this door-to-door delivery has been one of the trademarks of Norwegian salmon industry, making it reliable and efficient (pers.comm., 16 March 2017).

2.2.1 Salmon as a Product

The fish processing is either done locally in Norway, or Europe. Fish is easily perishable and needs to be preserved in order to be stored, and transported to the consumer. Cooling is the most common preservation method used today, and the products are transported to the customers either on ice in fish boxes, in a super cooled state, frozen or in a modified atmosphere. By lowering the temperature in the fish meat down to - or below - the freezing point, the shelf life of the fish significantly increases compared to room temperature preservation (Nordtveit, 2009). Both preservation trials and upscale in industrial production have shown that every stakeholder involved can benefit from keeping the lowest possible temperature from the fish is caught until it is exported and reaches the markets (Nordtveit, 2009).

The obvious advantage of using freezing as preservation method is the increased shelf life, and the possibility for a much more robust transport chain, with less waste and less need for swift movement (Ellingsen et al., 2009). However, the market does not perceive the frozen products as fresh, although the quality can be high. The export volumes of frozen salmon from Norway today are thus insignificant compared to the export of fresh or chilled salmon (Statistisk Sentralbyrå, 2017), all export products are consequently treated as a uniform product group.

As more of Norwegian farmed salmon is expected to be sold as fillets, knowledge regarding fish quality, cleavage, RSW-chilling etc. will become increasingly important for Norwegian aquaculture (Mindling et al., 2011). Every time a school of fish is subjected to sudden high activity or stress, e.g. crowding and pumping associated with live fish carrier loading, a percentage of the specimen does not survive, increasing the importance of limiting the amount of times the fish is handled while alive.

Sending volumes of salmon through the downstream aquaculture supply chain is a complex operation of congestion, pumping and killing combined with warehouse stops and long transport legs to market. It is thus vital that the fish is treated optimal according to welfare in order to preserve fish quality; avoiding stress, and followed by rapid cooling and swift transport to consumer.

2.3 Supply Chain Development

With an expansion in production volumes will the supply chain have to evolve. For smaller fluctuations would this usually entail acquiring an additional transport unit, or factory reconstruction. However, an increase in the five-fold scale may need an entirely different system composition.

2.3.1 Fish Welfare and Production Increase

A popular pronoun says that the only certain thing about the future is that it is uncertain. However, the EXPOSED aquaculture operations center (SFI EXPOSED) and Blått Kompetansesenter AS (pers.comm., 16 March, 2017) recognizes Norway's strong position in the aquaculture, and states that further expansion in fish farming is indeed possible; Norway could farm five million tonnes of fish per year by 2050, compared to the one million tonnes in 2010, if key production and environmental challenges are met (SFI EXPOSED and SINTEF Ocean, 2015).

Driven by the need for increased space and better production environment, the farming of salmon is moving gradually towards more exposed areas. These sites provide more stable growing conditions, greater distribution of wastes due to constant water movement and reduces environmental impact, like the abundance of lice. In these exposed areas, new technical solutions combined with operational concepts is vital to preserve safety and ensure reliability in production (SFI EXPOSED and SINTEF Ocean, 2015).

Emergency slaughter is a part of the nature of aquaculture, and is when a cage or farm needs to be killed off prior to planned harvest. This can either be a result of disease, lice or fear of contamination. In case of disease outbreaks, like infectious salmon anemia (ILA), it is important to highlight the fact the fish can be consumed by people without danger (Steinum and Budalen, 2013). Additionally, according to Lerøy, most emergency slaughters are conducted due to overwhelming outbursts of lice. This leads to batches of unattractive products, with major skin flings, but which are safe to eat. Fish which, on the other hand, needs to be discharged, for whatever reason, can per Lerøy Midt AS (S.A. Fjeldvær, pers.comm., 6 October, 2017) in some cases be used in other biological products like fish-oil, and pig feed.

According to Blått Kompetansesenter AS (B. Johansen, pers.comm., 16 March, 2017) and Møre Maritime AS (S.K. Waagbø, pers.comm., 3 March, 2017), is an altogether ban of the traditional open waiting cage solutions used today probable. Practice today is that infected fish only can be held in conventional open cages if it can be confirmed that it does not constitute a risk of infection. This is, however, hard to regulate and control, and the cages are thus a contagiousness challenge (Steinum and Budalen, 2013). In order to maintain fish welfare, and meet these new potential restrictions, new technologies are erupting, e.g. closed cage constructions which can be located at sea or ashore. Other approaches to the regulation changes would be to hold the salmon on board the live fish carriers, serving as waiting cages, or slaughter the fish during transit to shore.

2.3.2 Hitra/Frøya Region and Collaboration

This paper will be based on the aquaculture industry in the Hitra/Frøya region. The particular location was chosen because it is located in geographic proximity to NTNU, subsequently entailing possibilities for access to firsthand information on both current operations and future development. Limiting the scope to a distinct reference district also helps to confine the simulation models, keeping the analysis distinct and intelligible.

As a part of the project ”Sustainable Infrastructure Development in Trondheimsleia” a Central Norwegian Coast Harbor Alliance between Rørvik Harbor (from 1.1.2015, Nord-Trøndelag Harbor Rørvik) and Kristiansund and Nordmøre Port Authority (KNH) was established in 2014. In order to position themselves towards a future export increase from Hitra/ Frøya, local municipality and industry stakeholders have worked on developing Jøstenøya industrial park since 2009. According to Hanssen et al. (2014), the region cluster produced 149 500 tonnes of salmon and trout in 2014, which represents approximately 140 truck departures pr. week. The industry park include a port, which grants access to the area from the sea (Bjørshol et al., 2015). Several companies, such as Lerøy and Marine Harvest, have already secured building areas on the island, ensuring a good position for the possible future industry growth. The SalMar AS factory Innovamar, which currently holds the highest production freight in the region, is located on the north-west side of Frøya, at Nordskaget.

The preliminary project thesis included three representative factories in the region; two of which were located on Jøstenøya, and a third on Nordskaget, see Figure 2.3. This representation of the industry network was partly continued in this master thesis, where it was attempted to replicate a scene with room for collaboration between the different aquaculture operators. However, as the main focus area was to which degree the main factory needed assistance during scenario alterations, it was decided to only operate with two factories; one main located on Jøstenøya, which represents one third of the regional production, and one at Nordskaget. The latter was exclusively used as backup.

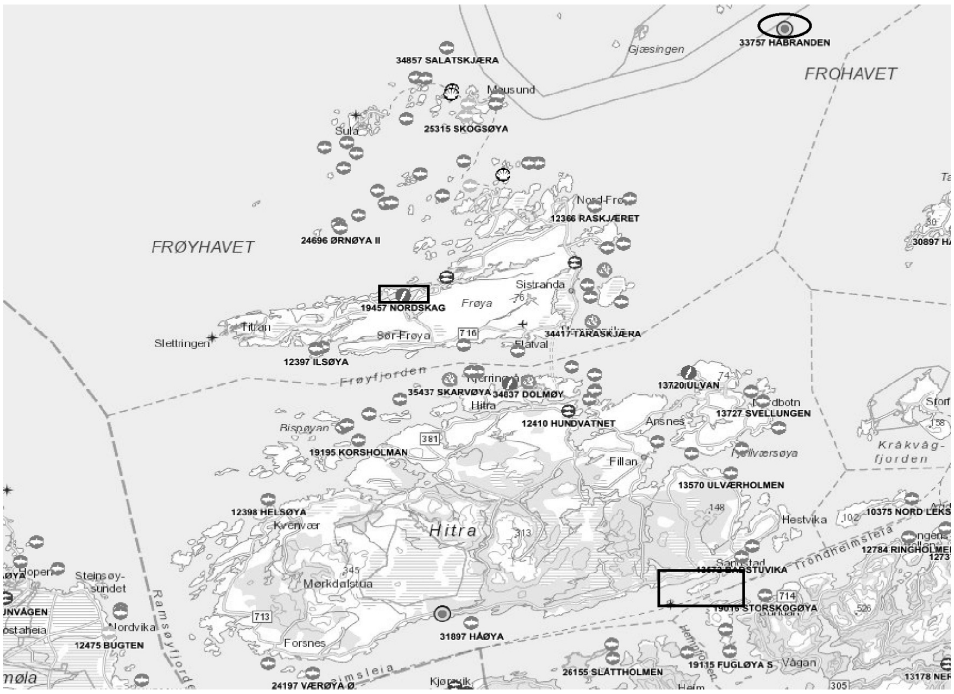


Figure 2.3: Map of Hitra/Frøya region. Jøstenøya and Nordskaget industry areas are highlighted with squares, industry fish farms are shown as encircled fish. Håbranden, location for SalMar's up-and-coming exposed fish farming in Frohavet, is encircled in the right upper corner (Kartverket, 2017).

Today, this cooperation is based on internal agreements, and restrictions implemented by the Food Safety Authority (FSA). When interviewed on the 6^{th} of October 2016, Sven Amund Fjeldvær explained that collaboration was primarily limited to crises which can compromise fish welfare, the producers are obligated to assist each-other in handling the emergency. There is, however, no commercial cooperation between the farmers, and no join supply chain infrastructure.

This lack of collaboration between the farmers is one of the most inhibitory factors regarding industry development according to Bjørnar Johansen at Blått Kompetansesenter AS (pers.comm., 16 March, 2017). For instance; an absolute precondition for achieving greater use of intermodal transport solutions is that it provides a financial gain for those involved. Thus, terminal owners and ship owners need sufficient volumes to fill a cargo ship. Scheduling liner traffic between the continent and central Norway requires that it is possible to enter binding freight contracts between the exporters and ship owners. Collaboration is a measure which increases the otherwise low frequency of a large vessel, all the while maintaining its cost effectiveness. However, it does also erase some of the competitiveness between the companies, and makes them dependent on the competitor.

2.3.3 Vessel Development

Where the live fish carriers of today focus on cage-side operation, the vessels of tomorrow must emphasize transit as the legs will grow more demanding. These new fleets will need greater capacities, due to increased production volumes. Not only will the individual vessels have to change, but the way they are operated, and fleet compositions will have to be reconfigured to fit a new market. Larger volumes, and in time, potentially longer transits imply possibilities to implement specialized vessels, where transit time can be effectively used to start the salmon slaughter process during mobility.

Following the assumptions that more of the export of Norwegian farmed salmon is going to be sold as fillets, and problems related to processing fish in pre-rigor states, studies on slaughtering the fish prior to reaching land have been carried out. The fish is loaded quickly and gently on board, then killed by a electrical impulse, bled, and moved into chilled saltwater storage tanks. Avoiding all restrictions regarding fish welfare during mobility, the vessel can carry as much as 2/3 fish and 1/3 water during transit. When interviewed on March 3^{rd} 2017, Svein Knudtzon Waagbø informed that processing vessels, which combine live fish carrier with harvesting vessel features, could in the future carry as much as 450-900 tonnes electrocuted fish. The combination of features also opens for more customized and flexible operation of the vessels. However, in order to acquire quality benefits from this method, the time spent between fish loading and processing needs to be short.

Introducing processing vessels to the supply chain means that approximately 1/3 of the factory operations can be moved from the on-shore facilities to the vessel, which naturally induces a high increase in ship complexity, and associated costs. These vessels represent the possibility to increase production efficiency, but also the need for important strategic logistics decisions in order to coordinate the different batches of slaughtered fish in need of processing. Møre Maritime AS informed that this can reduce the flexibility at the factory, as all fish arriving from the processing vessel ideally must be handled at once (S.A. Waagbø, pers.comm., during fall 2016). However, if the live fish carrier industry gets subjected to new restrictions regarding transportation of live fish in open tanks, or the water quality during closed transportation, harvesting directly from cage could get even more relevant. The fact that the FSA and customers consider this method better regarding fish welfare, compared to live fish carriers and waiting cages, is also of high importance (Mindling et al., 2011).

2.3.4 Seaborne Transportation to Europe

According to Amble and Ferraz (2012), approximately 1000 trucks associated with the export of fresh farmed salmon drive across the Norwegian mainland every week. Thus, in the light of possible future production increase, a series of projects have been initiated, looking into possible solutions for alternative transportation (Mathisen et al., 2009; Amble and Ferraz, 2012). Hitra municipality in cooperation with KNH has, for instance, taken an initiative to establish a sustainable maritime transportation solution for exporting fresh salmon from mid-Norway, to Esbjerg Port in Denmark. They argue that this reorganization will reduce greenhouse gas emissions and man-hours spent on shipment, improve road safety and contribute to increased mobility on the road network in Norway (Bjørshol et al., 2015).

During the research for the preliminary project thesis in 2016, it was found that the industry was reluctant to incorporate seaborne transportation. The argumentation was that the involvement of a vessel requires major production logistic changes in an already elaborate supply chain, and adding to the risk of shipment delay (Rørtveit and Lilienthal, 2016). Additionally, it is important to emphasize the commercial factor, as the driver costs today are carried by the importer. These typically utilize transport companies from low-cost countries in the EU, with significantly lower driver salaries than Norwegian trailer transport. Hence, all future transportation chains will be compared with this relatively inexpensive trailer transport line including the challenge regarding the ballast mode transit back again (S.Bjørgo, pers.comm., 5 October, 2016) (Amble and Ferraz, 2012). However, Bjørnar Johansen at Blått Kompetansesenter AS expects seaborne transportation to be reality within three years, and the Central Norwegian Coast Harbor Alliance has entered into a collaboration with Esbjerg Port (pers.comm., 16 March, 2017).

2.4 Supply Chain Intersections and Bottlenecks

Sudden disruptions to the supply chain could reduce the total product supply, leading to income deficit. A supply chain bottleneck would in this case be a process, activity or factor, whose limited capacity reduces the capacity of the whole chain, under the pretext that a system is only as strong as its weakest link. In the journal "Supply chain configuration: Concepts, solutions, and applications", Chandra and Grabis (2016) addresses some supply chain factors which are directly influenced by the supply chain configuration, and hence can serve as performance- and bottleneck indicators in a supply chain assessment. These include, for instance the flow and accumulation of materials, inventory, information and cash, total product throughput, capacity utilization and waiting times.

Analyses of these factors throughout the chain will involve approaches which need to consider interactions among nodal- and modal activities and capacity differences. As previously stated, the aquaculture supply chain consists of multiple interactions, and a flowchart describing the different transportation modes can be seen in Figure 2.4. This illustrates three phases; from cage to shore, through the land-based facilities with slaughter and processing, and transportation to market. These are the three phases of a supply chain; the procurement of materials, transformation of these materials into intermediate and finished products, and distribution of these finished products to customers (Campuzano and Mula, 2011).

In some supply chains, or structures, single items (components, subsystems) may

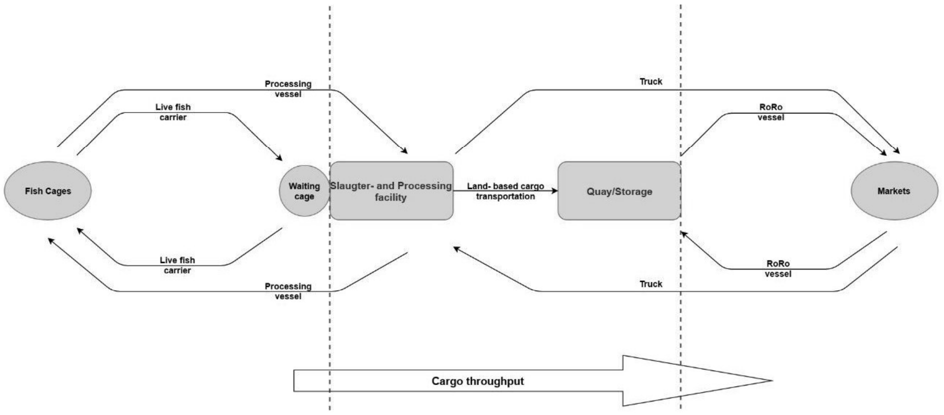


Figure 2.4: Downstream aquaculture flowchart (Rørtveit and Lilienthal, 2016).

be of much greater importance for the system's ability to function than others. For instance, if one single unit is operating i series with the rest of the system, failure or delay of this single item implies that the system fails. Two ways to ensure higher system reliability in this case would be to (1) use items with very high reliability/capacity in these critical places in the system, or (2) introduce redundancy (Rausand and Høysand, 2004). This would in this case mean two or more units or subsystem operating in parallel. The subsystem could then either share the load form the start until one of them fails, or one of them could be kept in standby until failure of the main system.

Figure 2.4 displays the different modes which need to interrelate. One important part of the figure is how there is a high degree of redundancy in the parts where transport modes, like road and waterborne transport, share load. Slaughtering- and processing facilities are, however, operating exclusively, and the dependency on cooperation between the regional factory owners in crisis situations, is thus vital.

Through the project thesis, it was found, unsurprisingly, that the bottlenecks tended to be at the interfaces – the areas of exchange of either information or goods (Rørtveit and Lilienthal, 2016). A bottleneck in the transportation systems sector can be defined as a shortage of infrastructure resources of the multimodal transportation systems network. Hence, delays or downtime in some parts of the supply chain could have a significant impact on the rest of the chain, and consequently, the total cargo throughput.

System Challenges

As presented throughout this chapter is Norwegian aquaculture built up of a series of modes and nodes which have to interact in order to serve their common purpose as a supply chain. With development of the fish farms, these all have to adapt to a new level of production. Introduction of additional seaborne transportation modes, like harvesting vessels and cargo vessel shipment, might thus be justified, further increasing supply chain complexity.

As an evolving industry, the supply chain has to be under constant observation for inefficiencies and capacity mismatch. This involves considering which nodes and modes serve their purpose satisfactorily. Subsequently, evaluations of system interaction and throughput becomes important knowledge upon developing the system, which will be presented in the next chapter.

CHAPTER 3

Problem Description

The downstream aquaculture supply chain is complex, and requires frequent and swift restructuring of cargo in order to prepare for the next transportation leg. The key to developing a sustainable supply chain with smooth transitions lies in the comprehension of component interactions. Understanding the effects an element has on preceding and proceeding elements allows for better management decision-making on development and re-structuring of the supply chain (Campuzano and Mula, 2011).

This thesis aims to give intel into supply chain interactions, and as such contribute to useful knowledge regarding future development of infrastructure for downstream salmon aquaculture. In order to unveil potential weak links in the future supply chain, a high-level schematic of the total flow was decided as a useful tool. Subsequently were supply chain performance evaluations performed in a discrete-event simulation modelling tool, into which initial research and input analyses formed the basis for model construction. Emphasis was put on the composition of transportation modes and nodes, and the interactions between them with respect to capacity- and dimensioning miss-matches, and system throughput. This formed a basis for further evaluations of system sensitivity, dynamics and flexibility, and thus provided a better system understanding.

3.1 Project Approach

In order to acquire a further understanding of the overall flow patterns and interactions, two overall approaches were applied. The first aimed to understand the system as a whole, and through iteration find a system composition which served the consumer in the best possible way. The second imposed different scenarios on the model, which aimed to replicate real-life supply chain alterations. Due to the implementation of these external factors, it was vital that the incorporated decision-making replicated the real-life conclusions.

3.1.1 Steady State

The first part of the problem was evaluating a steady state operation. This was the system under continuous running, without interruptions. The model should be able to run continuously without accumulating queues for neither fish nor vessels, and also function as a baseline simulation validation.

The steady system first aimed to replicate the current supply chain, which utilizes trucks for the transportation to market, and standard live fish carriers from cage to shore. This was then up-scaled in an attempt to predict how much the current infrastructure can handle. This is a theoretical infinite, and the limitation is primarily infrastructural aspects and strain on the Norwegian roadways. Subsequently, the goal was to find at what production level the introduction of cargo vessels would be favorable.

The steady state analyses also provided information about the fleet composition. Larger capacity implies needing fewer vessels, however, classic ship-building theory states that ships are cheaper per deadweight tonnes (dwt) the larger they become. This was an aspect which the model did not consider, but which would be important in real-life managerial decisions. Moreover, from a modelling perspective, the use of larger vessels implies lower frequency pick-ups, and thus longer warehouse stays for the finished cargo.

Collaboration was described as a key element in developing the future of aquaculture by Bjørnar Johansen from Blått Kompetansesenter AS (pers.comm., 16 March, 2017). This is because the industry and associated infrastructure is expected to grow out of proportions for Hitra/Frøya. Thus, joining forces opens new possibilities by sharing the cost of establishing new systems. In this study this possibility has been primarily manifested through shared capacity in the cargo vessels, and occasionally standby-production help from other farmers during peak periods.

3.1.2 Imposed Crash Scenarios

After analyzing how the system handled normal conditions, they were tested with different scenarios replicating production flow deviations. These were included to force the system out of equilibrium, in order to test how long it would take to return to the original steady state. This provides useful information about the robustness of a system. Naturally any supply chain which only functions under ideal conditions is not fit for operation. A set of independent simulation runs evaluating scenario infliction separately was also included to provide a better evaluation basis.

The imposed scenarios were chosen to be emergency slaughter and waiting cage ban. As explained in Chapter 2.3.1, these are realistic scenarios, which impact the supply chain. When large amount of fish needs to be rushed to shore outside of schedule, the system needs to restructure quickly in order to adapt. All vessels need to prioritize the critical cage or farm, and bring it abruptly to shore. This implies a larger than anticipated strain on the slaughtering facilities.

Imposing a ban on waiting cages would most likely be a result of pressure from the market regarding fish welfare. As described in Section 2.3.1, the result of a ban in the long term is most likely going to be land-based cages, however, this is technology which is yet to be invented. In the meantime it is likely that the live fish carriers will have to serve as stopping ground, laying quayside for 24 hours in order to calm the fish after transit. This imposes issues both because of the restricted volume in the cargo hold, and because it occupies the vessel for an excessive amount of time. In this regard, it is relevant to evaluate whether the amendment can speak to introduction of processing vessels.

3.2 Problem Boundaries

Exposed aquaculture has various multidisciplinary problems, and the scope of this thesis needs to be confined. Based on this, the included topics emphasized the fields in which discrete-event simulation is particularly effective. This implied disregarding topics such as economy, pollution and HSE, and is restricted to the practical movement of entities. The analyses focused on the influence of the imposed scenarios, but disregard all other events, e.g. extreme weather, inflation, etc. These are events which are of interest, however, they have been left outside the scope.

In order to create a relevant market situation, the simulation was from the point of view of one arbitrary producer. The input was based in part on both Lerøy Midt AS and SalMar AS production numbers, and scaled to future quantities, but it is not the goal of the project to give producer-specific decision support. As the study aimed to give intel into the supply chain interaction, focus was not on finding accurate numbers, but rather uncover tendencies.

Fish Production

The focus for this report was the downstream supply chain, which included everything which happens after the fish is fully grown, but disregarded everything up to that point; from delousing to maintenance. Modelling a biological system was a challenge because it does not follow strict mathematical algorithms, but vary with both time, salinity, temperature, and other environmental factors. This implied that some simplifications had to be made in order to simulate the development.

Only studying the down-stream chain implies considering everything up-stream to be a black box. There are, as mentioned, several factors adding to the level of production output from the cages, however these were neglected. The primary reason to include each cage would be to get the sailing times correct, but this was solved otherwise, see Chapter 6.1.2. The steady production might seem artificial, however, production in real-life is planned in great detail, and scheduled to avoid overlaps, and it is subsequently considered a fair assumption. The seasonal variations are impossible to ignore in real-life, and were thus taken into account in the model. It is also important to emphasize that the production volumes used in this project represents one company alone, and all scaling was done relative to these numbers.

As mentioned in Chapter 2.2.1 is the export of salmon dominated by freshly gutted fish, and this proportion will increase. Due to this, all products were treated as fresh produce in this study. After entering the slaughter facility, all products went through the same lane of processing, including all emergency slaughtered fish. This is assumed because the fish is sent to various consumers; some process the fish further, while others use it for consumption directly.

Slaughter Facilities and Regional Cooperation

Chapter 2.4 states that one way to ensure system reliability would be to either ensure constant high capacity, or introduce redundancy in form of a parallel standby system. In this supply chain, there was capacity flexibility for all the transportation legs by chartering additional units, whereas the land-based facilities have a rigid potential. It was thus made sure that the factory capacity in this thesis was held high throughout most simulations, in order to spot additional system bottlenecks. Because all management adjustment possibilities regarding factory, e.g. shift work, was disregarded, would the utility measure be misleading regardless.

According the system description and assumptions made in Chapter 2.3.2, one factory consisted of waiting cages and slaughtering and processing facilities. In order to represent the cooperation agreement between the current aquaculture stakeholders in the region in the best possible way, it was chosen to include two additional factories, represented by one common production line. These took over parts of the production if there was excessive pressure on the main factory, and hence only represented the shared load. All additional costs induced by moving production between the various facilities were excluded, and it assumed to have sufficient capacity available.

As mentioned in Chapter 2.3.1, all human factors, such as the individuals' competence, downtime from human error, etc. were disregarded. This also includes management and organizational aspects. The processes happening inside the slaughtering facilities were disregarded, and considered a "black box" with no production deviation. The facilities were also assumed to be modern, ensuring quick processing after slaughter, and no delay related to fish in rigor mortis state.

The standby factory is only used in emergencies to unload the primary system. This feature was included as a on/off option, and included cooperation both in terms of shared slaughtering capacity, and cargo vessel transportation to Europe. This was done because the extreme values provide the most information about the effect, however, a partial collaboration could also be possible. The production up-scaling was assumed applicable to all actors who contribute to loading the cargo vessels, thus assuming a three-way equal partnership between farmers in the region.

Chapter 2.3.1 implies that a prohibition of open waiting cages could be a reality before long. This thesis, however, only included the transition period, when live fish carriers have to substitute in as temporary waiting cages.

Vessels

Vessels transporting the salmon from farms to the slaughtering facilities have different contracts of employment. This thesis only included a fleet of vessels performing round-trip operations. Several sources have, however, supplied vessel operational data, ensuring a realistic basis for modeling the ring of vessel operations, see Section 5.3.2. Any differences regarding seasonal operation patterns are not accounted for.

The three fleets of vessels were modeled as homogeneous groups, where each group comprised the same capacity, annual downtime and average speed. Incorporating individual vessel differences was evaluated to require a model complexity level which could not be justified in terms of scope limitations and return on results. The vessel speed is an average value, as the speed close to shore or port will be reduced, and higher speed will be allowed in open waters. Other vessels operating in the areas were not accounted for, and the weather influence on operation activities were limited to a uniform representation of specific wave heights, see Chapter 5.2.2.

The vessels which were incorporated for fish transfer will typically also be used in operations apart from transit, but were for the purpose of this study exclusively used for transportation, and (in the case of processing vessels) for slaughter. This is a modification, however, it is likely that vessels in the future will become more specialized, so the representation might grow more true with time. The vessels were given a maintenance schedule, represented by an annual downtime, in order to give a more realistic and dynamic round-trip. This schedule also includes time used for cleaning of the on-board tanks etc. after transportation of emergency slaughter fish batches.

Processing vessels used in this project functioned solely as “harvesting vessels”, slaughtering the fish directly after loading, although they can be designed to operate as live fish carriers as well. This was done to underline the operating differences between the two vessel types, because it represented a new system flexibility, as described in Section 2.3.3.

Following the disclaimer of all human- and organizational factors, it was assumed that all transporters can operate every day throughout the year, and at all times of the day. This will be a big modification from the real world, where, for instance, all loading- and unloading activities primarily would take place during the daytime.

Ideally, the initial vessel transportation should be coordinated to with land-based facilities and cargo vessel departure. However, considering the continuous nature of the model, would schedule departures according to shift hours not be possible.

Product volumes lost due to rough fish handling, and leftovers discharged after slaughter and processing, was discussed, but not included quantitatively in the simulations. Consequently, was all input be equal to the output. This is a rough assumption, but makes it easier to validate the model, and compare production flow between simulation runs.

As described in Chapter 2.3.4, one of the biggest challenges of introducing seaborne transport of salmon to the continent, is the ballast mode transit back again. However, as this study disregarded all monetary considerations, the only consideration made was trying to keep the number of vessel round-trips within bounds.

The only included parameter input from the weather is significant wave height, however, other factors would in reality also come into play, such as wind, currents, rain, etc. All weather generation in the models were applied as uniform layers for the areas they represented. There were no local deviations, which is most noticeable for the cargo vessel sailing legs, as these span the largest geographical areas. Additionally, the data used from the Norwegian sea to simulate these wave conditions does not represent the exact location of the cargo vessel sailing legs, but merely aim to serve as representations of the general conditions. The future weather is going to be unpredictable regardless, and should thus only be considered here as an indication of how significant the influence is.

Market Distribution and Truck Transport

Domestic consumption of the products were not considered in this thesis, as the transport lines regarding export were weighted more interesting for the scope of the thesis. Over 95% of all seafood caught or produced in Norway is exported, and this percentage is expected to grow (Kvistad, 2014). Additionally, all export volumes bound for markets in the far east and west were assumed to leave by plane, and are thus transported to Gardermoen Airport for further distribution. It should be noted that these figures are retrieved for Norway's total salmon export, and are assumed to give an approximate distribution ratio for the region Hitra/Frøya as well.

The procurement of trucks is perhaps the most inexpensive and flexible part of the system. In reality these are chartered when needed, and will thus never be in excess. The model aimed to replicate this, and by generating a sufficiently large fleet one acquires constant supply. Hence, the supply of trucks will not represent a potential bottleneck for the system, but rather be a measure of necessary transportation capacity.

After consulting with Lerøy Midt AS for the project thesis (pers.comm., 6 October, 2016), fragile roads, traffic jams and mechanical unreliability was said to occasionally lead to severe transport delays, sometimes resulting in necessary disposal of the cargo. However, this aspect easily can be included in final results consultations without including delay-factors in the simulation model, and was thus not evaluated qualitatively.

Problem Approach

In light of the thesis objective, the problem description emphasizes how one can obtain information which is useful in understanding system interactions. Two approaches, one for steady state and one for imposed scenarios, were selected to illustrate the challenges the industry might face. These were included in a discrete event simulation, in order to provide information about entity movement through the system, which will be described in detail in the next chapter.

CHAPTER 4

Methodology

Focus has been on gaining a qualitative understanding on a high system level, based on a quantitative approach. For this purpose, simulation is a powerful tool. Simulation has become widely available with the evolution of high performance computers, and problems which are theoretically possible to solve analytically, but excessively complex in practice (Ross, 2013) are the typical examples. By simulation, the calculation are simplified, although runs over a longer stretch of time still require significant processing capacity, due to the large amount of processed data. This chapter will present how simulation has been used for this thesis, and what performance indicators have been applied.

4.1 State of the Art

In a modern supply chain, there is a near constant exchange of goods, information and responsibilities. Industrialization has motivated a gradual move towards optimization, and a continuous margin hunt, which in turn has lead to sophisticated analytic tools. These all aim to serve as decision-making tools, optimizing the interaction between supply and demand. The primary objective is to minimize time and money spent on stop-overs awaiting the next link of the chain, storage, transit, etc.

With the development of computer technology, ever more powerful tools are becoming available to the common man. With it a series of IT-related fields and technologies have emerged, making way for a brand new series of data analyses. Waller and Fawcett concluded in their 2013 paper that data science, predictive analytics and big data were expected to revolutionize supply chain management (Waller and Fawcett, 2013). These also mention a set of skills which are needed in order to best manage a supply chain. In addition to the obvious marketing and finance elements, were optimization, statistics and discrete event simulation mentioned as important knowledge fields.

Performing supply chain simulation is considered where full scale testing is expensive, impractical or impossible to do. The two main approaches to supply chain analyses are analytical or simulation-based. The former provides optimal solutions based on calculation, but is highly dependent on assumptions and available formulae. Simulation-based analyses are able to describe complex systems, however they do not provide an absolute optimal solution, and every change in input must be related to a series of outputs. In order to provide value, the results have to be understandable and provide insight. In simulation, this value lies in assessing system performance, which can be understood from queue length, waiting time, total throughput and utilization (Campuzano and Mula, 2011).

In 2013, Cigolini and Pero researched the effect of supply chain length from start to finish. It was uncovered that the expected order size from each distributor increases with lower lead time. A longer supply chain also implies that the time from registered demand until development in supply is elongated, making it harder to correctly estimate production. The research also found that the average stock per actor is affected by chain length (Cigolini et al., 2014).

A common term in supply chain simulation is the bullwhip effect, which refers to fluctuations imposed in the early phase of the chain, without direct effect on the latter phases. The bullwhip effect then trickles down the system, causing a wave of distortion through the system, from where it is imposed, indirectly affecting the entire chain (Campuzano and Mula, 2011).

Omogbai and Salonitis tried to uncover the value of discrete event simulation during a transfer to lean practice. These showed, among other things, that processes and procedures do not provide any value without another supply chain element counterparts. They also experimented with maintenance frequency and duration, finding that rarer and longer stops have a significantly lower effect on total lead time than frequent, shorter stops.

4.2 Simulink

The software which was applied is MATLAB's SimEvents, which is a model based simulation tool. SimEvents presents a graphical drag-and-drop interface for assembling and using models in Simulink. Being a model based simulation tool, it is fitting for the scope of this thesis, and the interactive interface allows for a fairly intuitive model structuring. Simulink provides a discrete-event simulation engine and component library of blocks, allowing the developer to build the model block by block, using customized block libraries and solvers for modeling and simulating dynamic systems. This makes the program apt for simulation od a large spectrum of systems, ranging from fluids, to advanced rocket science.

As an integrated MATLAB program, Simulink can incorporate MATLAB coding and export scope results to MATLAB workspace (MathWorks, 2017). This provides flexibility in post-analysis treatment of data, by finding statistical figures for a run. By adding scopes one is able to track entity movement in a block, and thus understand the flow, or evaluate errors. The very flexible interface also poses a challenge as it puts the responsibility for the correctness of the model entirely on the programmer. Finding errors in a model with close to no set framework can be difficult, and requires constant validation throughout construction.

The backbone of the model is the chronological, physical system. This can then be elaborated by simulation-technical blocks, serving to the outputs of the model. Together these represent the system, as well as the ability to collect data from it. Simulink is able to provide output is a series of formats, from scopes to workspace matrices, and it is thus up to the programmer to chose the appropriate method considering the scope and limitations.

4.2.1 Model Terminology

Entities are the units which are being transported through the modeled system. These are the ones being handled in blocks, and will move according to instructions. An entity can also be assigned various attributes.

Attributes are characteristics written to entities. These can be altered as the entities move from block to block, and can thus model loading cargo, choosing a distinct route, track time, etc.

Global variables are variables which can be accessed from any part of the model through the use of “MATLAB Function”- blocks, “Data Store Read” and “Data Store Write”. This means that their current value can be queried and changed, allowing different model parts to communicate with each other. Subsequently can generation of waves, fish cages, storages and other intersections be accessed quickly.

Blocks make up the entities path from generation to termination. The various blocks have different functions, some aim to imitate the real-life system, whereas others function as sensors, recording and transmitting results. Below is a list of the most frequently used blocks in this project.

> **Servers** are blocks where an entity is kept for a predetermined amount of time, simulating a time-demanding event, such as sailing. A server block is illustrated by a circle, as shown in Figure 4.1.

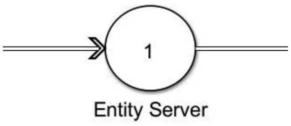


Figure 4.1: Simulink server block.

Queues are blocks where entities are kept until the next block is available, just like the queue for an available cashier in a store. All blocks in this project are FIFO (first in-first out), indicating that the first entities to arrive are the first to leave when there is availability. A queue block is illustrated by a rectangle with the queue type inscribed, as shown in Figure 4.2.

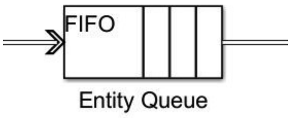


Figure 4.2: Simulink queue block.

Entity batch creators and splitters join a predetermined amount of entities from the same path into one batched entity. This is useful for gathering cargo entities into one entity, the size of each batch corresponding with the capacity of a transporter entity, allowing them to merge later on. The accessibility to entity data is greatly reduced while batched, and altering attribute values of specific entities in a batch is not possible. An entity batch creator and splitter can be seen in Figure 4.3.

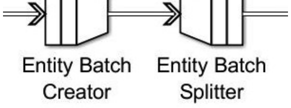


Figure 4.3: Simulink batching- and unbatching blocks.

Composite entity splitters and creators join two entities from different paths into one. This is useful for merging cargo entities with transportation entities during their common transit. The incoming entities do not need to have the same structure, but the accessibility to entity data is greatly reduced while merged. A composite entity creator and splitter can be seen in Figure 4.4.

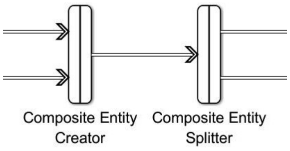


Figure 4.4: Simulink composite entity creator- and splitter blocks.

Entity input and output switches join several paths into one. The output switch chooses the next path based on attribute values, or a predetermined switch pattern. The incoming entities need to have identical structure, i.e. attribute set-up. This is useful for joining identical entities which have been on different paths, such as new and returning vessels. Swithing does not impact the entity structure or accessibility at all. An input and output switch can be seen in Figure 4.5.

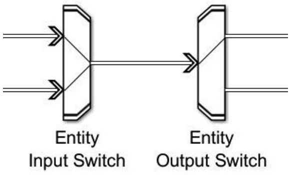


Figure 4.5: Simulink entity input and output switch blocks.

Entity gates control a path. A corresponding function block sends a signal about whether the gate should be open or closed. This is useful where entities could otherwise proceed to an unavailable block and disappear, or because the entity should be held awaiting a new message. An entity gate with function block is shown in Figure 4.6.

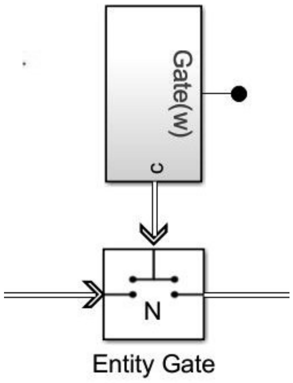


Figure 4.6: Simulink gate block.

Scopes display the output from another block. The different blocks produce different statistics, e.g. entities in block, entities departed, etc. These scope plots provide valuable information about how a block interprets and treats the incoming entities. An entity server with a scope block is shown in Figure 4.7.

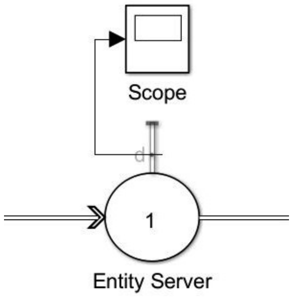


Figure 4.7: Simulink scope block.

4.3 Simulation Limitations

A perfect model will be impossible to achieve, thus it is important to take note of the most prominent deviances from the real-life system. Many of these challenges were bypassed by adding blocks, data stores or other work-arounds, while the ones which could not be sidestepped within the scope of this project. In addition to the system assumptions and scope limitations presented in Chapter 2.1 and 3.2, will there thus be a set of limitations from the simulation tool.

The model is adjusted for around-the-clock running, with average hourly production. In a real-life system would production shut down during the night, and the process would be delayed until the next morning. This caused slight shifts in deliveries. Due to this will the utility measure be misguiding, and cannot be included as a direct performance indicator.

Steady State

Unlike the real-life system, the simulation runs with no manual intervention. During peak periods, a producer would hire additional capacity from its competitors, or charter additional vessels. The resulting extremes in the rest of the supply chain would thus be taken care of outside of the simulated system, whereas the simulation includes no spot options or other flexibilities.

Artificial Queues

In a simulation model are queues a vital part. If an entity is sent to a server which is unavailable, it is dropped from the system. It is thus paramount to include queues wherever there is a slight chance of occupied blocks. This implies adding queues where there is no physical equivalent, which only serve to maintain a realistic system. These queues pose a problem when they start to accumulate entities in places where these would never be stored. In real-life would other measures be taken, such as rescheduling, or slowing down to time arrival.

Lingering

At every production site there is going to be delay due to lingering. It is inevitable that even short processes and transits will be postponed and downgraded occasionally. This is a factor which was disregarded in the simulation, where it was assumed that the factory works with 100% efficiency. There would most likely be added time due to unnecessary and necessary lingering, however, this would affect random entities, and as such not have a general effect on supply chain efficiency. Based on this, all lingering is disregarded.

Continuous Adjustments

The simulation model is self-regulated, meaning that once the developer hits play, no further adjustments are subject to manual intervention. This is a simplification from the real world, where adjustments would be incorporated continuously. The simulation thus has to be coded so that it is able to make decisions and prioritize similarly to that of an operator. This proved to be a trying process, for instance, is vessel routing always based on the availability of cargo and vessels, replicating servicing the most critical ports first.

Attribute Accessibility

The use of composite entity creators in order to model merging of transporters and cargo, has some drawbacks. The first is that although the two entities merged do not need to have the same structure, they need to be of a consistent size thought the simulation. This means that one unable to wary the degree to which the transporters should be loaded between round-trips, and has made it necessary to assume 100% full load every run. Subsequently taking unnecessary time spent waiting to fill a whole ship, or truck, during times with lower production volumes. Additional, the accessibility to entity data is greatly reduced while in a merged state.

4.4 Measuring System Performance

Improving supply chain performance has become one of the critical issues for gaining competitive advantages for companies. Caia et al. (2009) writes in the report “Improving supply chain performance management: A systematic approach to analyzing iterative KPI accomplishment” how evaluation of supply chain activities according to a set of KPI categories can be a systematic approach to supply chain performance assessment.

Individual measures of supply chain performance have usually been classified into four categories: quality, time, cost and flexibility (Caia et al., 2009). For the multimodal supply chain at hand, is a comparison of KPIs related to time consumption of particular interest, due to its quantifiable measure possibilities and the customer demand for tight delivery schedules.

The quantification of other indicators, such as system responsiveness and customer satisfaction, can, however, prove to be challenging. Figuring out the intricate relationships among these KPIs, and the order of priorities for accomplishment is difficult due to system complexity and the frequent reconstruction of cargo. Subsequently, in order to evaluate how nodes and modes in the downstream aquaculture supply chain interact, one has to remain vigilant with regard to queue development, and entity accumulation continuously, rather than retrieving quantitative KPIs.

4.4.1 Time as Key Performance Indicator

For a biodegradable product, such as salmon, is time to market an important factor due to the compromising effect it has on the product. Most producers find themselves working against the clock one way or another, consumers become increasingly spoiled and the range of suppliers increase (Marr, 2012). Supplier are under pressure to able to deliver quickly. This is especially applicable to consumer goods such as food.

Tracking time is not a built in feature in Simulink, and it is thus the responsibility of the developer to find methods which measure the desired time, and produce output. The challenge is to find start and end blocks which include relevant phases. A KPI should represent information which in turn can be used for qualitative judgment (Baker, 2006), which entails that the time measured needs to be from a mirrored reality, and not influenced by simulation technical infrastructure.

Simulation Data Inspector

In order to assess results, output needs to be accessed. This is done continuously through scope blocks, but the most powerful evaluation tool in Simulink is the Simulation Data Inspector (SDI). Where scopes and “To Workspace” blocks will replace the stored data every time the simulation is run, the SDI saves results from all previous simulations. This allows for detailed comparison of the same data from different runs, or different data from the same run, and provides an easy way to export large amounts of data for various runs.

The most useful feature in the SDI in conjunction, with respect for this project, was the ability to access attribute data. These are altered in specific blocks, and include information about time, loading, route selection, etc., and comparison between runs proved useful both during model validation and result evaluation.

4.4.2 Supply Chain Vulnerability

Understanding which key functions and capabilities that are prerequisite for the ability to move goods, is essential for detecting chain throughput vulnerability. The next sections will present how a set of factors, and the system input and output, make up the basis for the system vulnerability analysis.

According to Asbjørnslett and Rausand (1999) in the article “Assess the vulnerability of your production system”, system vulnerability is related to its ability to endure threats and survive different types of accidental events. Or, in other words, its inability to resists impacts of hazardous events, and restore itself to its original function following the event, see Figure 4.8. This new, and stable situation for a resilient system may be lower or higher than the former stable situation.

Section 2.4 addressed the importance of system redundancy, highlighting which subsystems that are most exposed to external and internal threats. Examples of

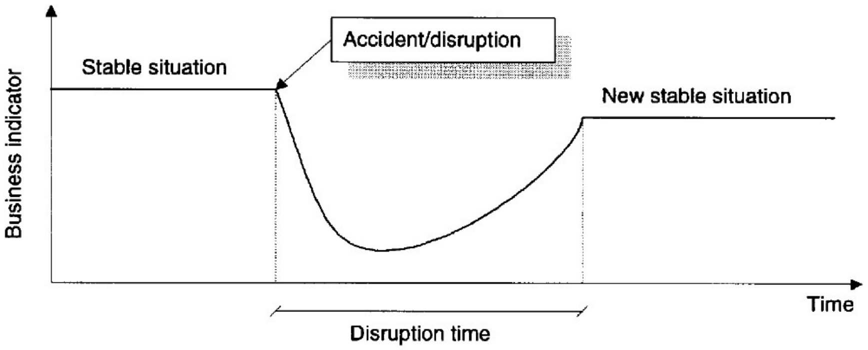


Figure 4.8: Regaining system stability after disruption, (Asbjørnslett and Rausand, 1999).

these are all carefully described in Chapter 3 in the report "Assess the vulnerability of your production system". Based on the system boundaries presented in Chapter 2.1, would a vulnerability assessment in this case be most useful for systems exposed to external impulses, like emergency slaughter actions and environmental impacts. This is in contrast to internal, reliability-oriented failure accidents which are disregarded in this study (Asbjørnslett and Rausand, 1999).

After evaluating which factors and subsystems contribute to system vulnerability, or (in cases of reduced capacities) function as barriers, the next step would be to evaluate the wanted and unwanted flows in the system. The first include fish production, delivery of products to the end consumer, vessel performance, etc. The unwanted outputs, however, comprise direct- and latent failures. The most prominent in this case would be waste production, low utilization of supplies etc., which reduces the financial margin of the system (Asbjørnslett and Rausand, 1999). In this study, would the flows of particular interest be related to throughput and accumulation of entities, and how these affect system performance.

By detecting and evaluating how the deviations caused by the external impulses affect the system, one can assess the system vulnerability. Mitigation form normal operation would then become evident, and system vulnerability possible to evaluate.

4.5 Continuous-time Markov Process

Generation of weather will in this thesis be done by a markov chain approach, and will be further described in Section 5.2.2 and 6.1.3. A Markov chain is called a Markov process if it fulfills the Markov property. This means that the system behavior at any time is instant and independent of history (except for the last state), and is restricted to constant rates.

Markov processes have many applications as statistical models of real-world processes. In continuous time, the Markov process undergoes transitions from one state to another state, with a exponential probability distribution. The time spent in each state is also exponential distributed, and is non-negative. Hence, future behaviour of the model (both remaining time in current state and next state) depends only on the current state of the model, and not on historical behaviour. In order to simulate a system where the transition probabilities are influenced by long-term trends and/or seasonal variations, it has thus been assumed that the environmental and operational conditions for the system are relatively stable as a function of time.

Simulation as a Tool

By modeling time-dependent movement of entities analysis through a system where several transportation modes with varying capacity are bound to a common cargo, can simulation be an effective tool for generating supply chain knowledge. In this thesis, simulation was used to understand the supply chain; its processes and key problems. The software which was employed, Simulink, is a discrete-event simulation add-on to the MATLAB package.

Simulink has a graphic user-interface, there are a series of different settings and codes implemented with each block. This allows for personalizing of the model, but also requires cohesion. Model deviances from the real world is primarily related to adjustments which would be implemented continuously, such as the decision to charter a vessel short term, reduce staff during slow periods, or be prepared for to sudden production changes. These are considerations which have to be taken into account when evaluating the simulation results.

Simulation can help understand causality in a supply chain, by investigating what inputs affect what outputs. This is entailed by collecting information along the entire food supply chain, tracking the products from farm to the consumer. An ideal system should see very slight accumulation of queues, with no excess or lack of capacity. Based on the evaluations regarding entity flow through system intersections, can system vulnerability and dynamics be discussed in relation to system composition and dependencies, relative to simulation input.

CHAPTER 5

Simulation Input

The credibility of simulation results are dependent on the system input. The following sections present gathered information, whereas the validity of these will be discussed further in Chapter 8.1.

The input data was divided into three primary categories: absolute, variable and crash scenario. Absolute input data are based on real life absolute values or system limitations, and form the framework for the model through remaining unaltered throughout the runs. These are all possible to adjust through the Excel input sheet, but will be withheld in this thesis. The variables feature input which is altered by the user in order to produce a specific setting. These are changed throughout the simulation runs in order to provide information about how the input affects the output. The crash scenarios are exterior influences which can be imposed on the system to evaluate the magnitude of the influence. These are only introduced when it is of interest to check system response, and will be excluded otherwise.

5.1 Units

Simulink is a unit-less program, and operates with undefined entities and time units. These thus had to be decided upon, and maintained throughout the model in order to get time, distance and capacity relations correct. Because the amount of fish makes out the basis of the supply chain, this was a natural entity unit. However, defining one entity as one fish provided excessive results, and became very computationally extensive. It was thus decided that one entity should represent one tonne of fish. This implies that a capacity of 400 is equivalent to 400 tonnes fish. It was also assumed that all fish were harvested at the average slaughtering weight of 5kg. An exemption from this is emergency slaughter, which was only stated in weight, and where the weight per fish can be lower, allowing more individuals to be included in a vessel.

The simulation time-unit was set to hours, implying that a run over 13140 hours is equivalent to a year. The final model utilized traditional units and capacities for the seaborne modes; knots, nautical miles and tonnes. 1 knot and 1 nautical mile correspond to 1.852 km/h and kilometers, respectively. All trucks operated with kilometers and kilometers/hour. This simplified the interaction between the speed-units, because they correlate directly with the time-unit.

5.2 Absolute Input

5.2.1 Fish Generation

The generation of fish was one of the primary inputs, and the basis remains absolute according to the following section. The only added variation was the scaling, which can be altered in the Excel spreadsheet, see Section 5.3.1.

The development of slaughter ready salmon was considered to be an absolute number, and disregards how long the smolt has been in the cage. The fish was harvested steadily, and not in manners of complete cage or farm entities. After conferring with Kristian Kvam at Lerøy Midt, the production of fish was set to vary from 1000 tonnes to 9000 tonnes per month, peaking at approximately mid-August when the water temperatures are the most favorable for growth. This implied an hourly production of 12.5 tonnes at peak period, and 1.4 tonnes at the lowest level, six months later. The variation was made completely regular in order to simplify long-term simulation. With ultima at 12.5 and 1.4, the mean was 6.95 and the amplitude 5.55. In order to acquire a yearly cycle (8760 hours), the time value was multiplied by $\frac{2\pi}{8760}$. High production is during summer, so to shift the peak to August, a phase shift of 5532 hours was added. The resulting function for production, with x being the time in hours, was thus as follows:

$$f(x) = 6.95 + 5.55sin\big(\frac{2\pi x}{1394} - 5332\big) \tag{5.1}$$

In Simulink, all entity generators are reading input as a time differential, dt, time until production of next unit, which in this case was $dt = \frac{1}{f(x)}$. This represents today's production levels. Assuming these remain as is, i.e. no new inshore farms developed or closed down, the additional volumes in a five-fold production increase came from exposed locations. These are likely to be bigger and sturdier constructions, and combined with being surrounded by more stable water temperatures, the production is going to be less affected by seasonal variation. As a rough estimate, the exposed fish generation was thus set to the following:

$$dt = \frac{1}{27.95 + 2.55sin\big(\frac{2\pi x}{1394} - 5332\big)} \tag{5.2}$$

These can be scaled in the Excel file, which adds the factor to the mean, increasing the integrated total correspondingly. Both production rates are graphically shown in in Figure 5.1. The upper curve represents the hourly production rate throughout the year for the exposed fish farms, the lower for conventional farms. The corresponding integrated areas under the curves are the total yearly production in tonnes of fish (e) for exposed, and (c) for conventional. This means that the conventional and exposed fish farms produce approximately 60 000 and 243 000tonnes of fish respectively throughout a year.

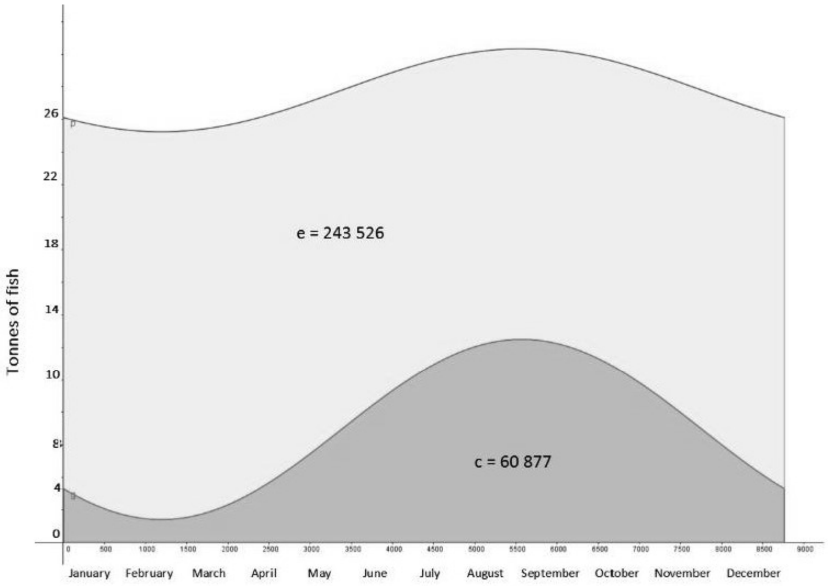


Figure 5.1: Hourly production throughout the year. Exposed farm represented with (e), and (c) for conventional.

5.2.2 Weather Data

Weather conditions in the simulations were represented by significant wave height (Hs), which is the mean value of the upper third of the individual wave heights in a given period. This corresponds to the height estimated by a ”trained observer”, and is commonly used as a measure of the height of ocean waves (Meteorologisk Institutt, 2010). Conditions are in real life also strongly dependent on current- and wind conditions. These factors were not included in this study, but were assumed to occur at approximately the same periods as high waves.

In order to predict weather conditions under vessel operations, two sets of met-ocean data were retrieved, and used as a basis for a Markov chain assessment of possible sea states, and transition probabilities between them. The first set was twelve years of historical data downloaded from *waveclimate.com*, for an area in the Norwegian Sea at 60° 00'N, 20° 00'W. The second set was measured by two oceanographic buoys developed by SFI EXPOSED. This data is classified, and will not be presented in the report, but served as input to affect operation windows by the farms and the time spent sailing; one for the exposed locations, and one for the traditional farm sites.

Frohavet was for this report used as an example for exposed farming area. This is because Håbranden, which is located in Frohavet, has been approved for installing a new cage design (Nygaard, 2014). Håbranden can be found in Figure 2.3 in Section 2.3.2. According to SINTEF, is the maximum significant wave height at this location at approximately 5m (Nygaard, 2014). This corresponds to a maximum wave heights of almost 10 m, and real life wave data from the are can be monitored through the home web page for SFI EXPOSED (SFI EXPOSED, 2017). As the met-ocean data used as simulation input for this part represents calmer waters, Hs values for the exposed farm was scaled up to fit a realistic weather picture presented by SFI EXPOSED, with a mean Hs of approximately 1.5 m.

5.2.3 Farm Locations

Regional farm locations were chosen through maps administrated by the Norwegian Coastal Administration (NCA), as shown in Figure 2.3. Eighteen conventional farm locations spread around the two islands Hitra and Frøya, and one exposed location at Håbranden in Frohavet, were used as basis for generating the sailing legs. All the sites were listed in an Excel spreadsheet, including their position relative to both processing facilities, and can be found in Appendix B. These then formed the input for four MATLAB vectors; two [18,1]-vectors representing the conventional sites and their distance to the factories, and two [1,1]-vectors for the exposed. Thus, next sailing distance could be set by keeping track of current vessel position, and next destination. The locations and distances can be altered if need be, but remained constant throughout this study.

5.2.4 Market Distribution

In order to model approximate transportation distances, a salmon export market distribution had to be assessed. Since domestic consumption was disregarded in this project (see Section 3.2) was the export volumes bound for four sub- categories; Western Europe and Denmark, East Europe (mainly Poland), distant markets (overseas markets, mainly USA and Asia) and the Nordic Countries (except Denmark and Norway).

Statistical figures retrieved from market insights from Statistisk Sentralbyrå (2017), were used to find an approximate salmon export distribution. The numbers were listed by export weight, and divided by country, and can be found in the attached Excel-file B. By categorizing the importing countries into sub-groups, and calculating their respective share, the market distribution was defined (see Table 5.1). Due to simulation restrictions, it was necessary to calculate internal shares between some of the market groups, see Section 4.3. These are presented in Table 9.3 inn Appendix B.

Table 5.1: Total market distribution of salmon export, by geographical area.

	Weight [tonnes]	**Percent of Total [%]**
Denmark	74 638	17.1
Belgium/ Netherlands	62 704	14.4
Germany	42 412	9.7
France	114 221	26.2
Spain	57 840	13.3
UK	72 087	16.5
Portugal	11 902	2.7
Total West Europe	435 804	47.3
Total Distant Markets	204 883	22.3
Total East Europe (incl. Russia)	218 166	23.7
Total Nordic Countries (SWE, FIN)	61 862	6.7
Grand total	**920 715**	**100**

Distances between Jøstenøya industrial park and the final export destinations can be found in Table 5.2. In order to simplify the simulation input, an example city in each country was chosen, and provided an associated driving distance for each location. Esbjerg harbor in Denmark represents a start-up opportunity for onward transportation to West - and East Europe after unloading the cargo vessel, and these associated driving distances are presented in Table 5.2.

Table 5.2: Transportation distances to markets.

	Destination representative	Distance from Jøstenøya [km]	Distance from Esbjerg [km]
Denmark	Padborg	1 203	107
Belgium/ Netherlands	Amsterdam	1 818	723
Germany	Hamburg	1 365	270
France	Paris/Rugis	2 257	1 153
Spain	Barcelona	3 128	2 083
UK	London	2 277	1 188
Portugal	Lisbon	3 985	2 879
Distant Markets	Gardermoen	541	-
East Europe (incl. Russia)	Poznan/PL	1 844	807
Nordics (SWE, FIN)	Stockholm	878	-

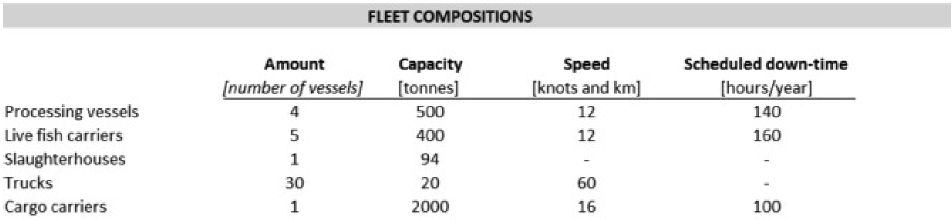
It follows from Table 5.1 that almost 50% of salmon exports go to customers in the western part of Europe, and around 20 percent to East Europe. Another 20 percent goes to distant markets, while the Nordic countries, except Denmark, gets about 7 percent. Looking exclusively at the total load being sent to Eastern and Western Europe, see Table 9.3 in Appendix B, as much as 67 percent are consumed by the latter. The distribution between volumes exported to distant markets and the Nordic countries, are respectively 77 and 23 percent.

5.3 Variable Input

5.3.1 Excel and MATLAB Workspace

In order to increase user-friendliness, and easily change impact scenarios and fleet compositions, the decision was made to use Excel as baseline input. This allowed for easy and intuitive structuring, rather than tweaking large MATLAB matrices. The input Excel spreadsheet can be found in Appendix B.

Having Simulink read directly from Excel is, however, an extensive and restrict-



FLEET COMPOSITIONS

	Amount [number of vessels]	Capacity [tonnes]	Speed [knots and km]	Scheduled down-time [hours/year]
Processing vessels	4	500	12	140
Live fish carriers	5	400	12	160
Slaughterhouses	1	94	-	-
Trucks	30	20	60	-
Cargo carriers	1	2000	16	100

Figure 5.2: Example of fleet composition as displayed in the input excel spreadsheet.

ing operation. We therefore decided to write a script which assigned the Excel values to MATLAB variables, which in turn was treated as values in the Simulink model. This proved to be an efficient system construction, although it did require remembering to save in Excel, and subsequently run the MATLAB script. Another challenge was that the MATLAB code collects data in specific cells, and any alteration in the Excel lay-out thus required a thorough update of the MATLAB code. The final MATLAB script can be found in Appendix A.1.

The Excel spreadsheets represented all the variables going into the model. These focused on the vessel and scenarios characteristics, and include production volume, robustness against weather, cooperation between companies, waiting cage ban, and emergency slaughter due to disease, etc. The spreadsheets also included sailing distances between cages and slaughtering facilities, berthing capacity, and market distribution.

5.3.2 Fleet Composition

The fleet composition was the key variable in the system. This provided information about how many units of each vessel, as well as their respective capacity, speed and downtime (annual maintenance). It can easily be altered in the spreadsheet, an example of which is shown in Figure 5.2, which was later imported into the Simulink model.

The three sets of vessel fleets, and the trucks, were homogeneous, and vessel specification numbers, like speed and capacity, used in the different simulation runs, were based on characteristics provided by Møre Maritime AS (pers.comm., 03 March, 2017), and development prognostics presented in Section 2.3.3. See references Møre Maritime AS (2016) and Mindling et al. (2011). Vessel speed reduction, during operations in harsh weather, was easily regulated in Excel. The same applied for the loading- and maneuvering activities for the cargo vessels at Esbjerg and Jøstenøya port.

Based on numbers from Lerøy Midt AS, the operation period at the fish farms for both vessel types was set to 5 hours (pers.comm., 14 March, 2017). This can be altered in Excel, but was kept steady for this project.

The sailing distance, and thus sailing time, was calculated based on the farm locations from Chapter 5.2.3. These can also be altered in the Excel input, along with specific wave height limit for cage operations and cargo vessel transit Hs limit. These were set to 3 m for both exposed, and traditional farm sites, and 9 m Hs for the cargo vessel, following recommendations from Lerøy Midt AS (pers.comm., October, 2016) and Blått Kompetansesenter AS (pers.comm., 16 March, 2017) respectively.

Truck Transport

The trucks were used for transportation between factory and end consumers, based on market distribution in Section 5.2.4, and according to schedules presented in Chapter 6.1.6. Real life conventional trucks used for fish transport over long distances are for example FH and Scania R500. These have a cargo capacity of approximately 25 tonnes, but restrictions on European roadways prevent loading more than 23 tonnes (Ellingsen et al., 2009). As the number of trucks used in the simulations only served as tentative numbers, a truck capacity of 20 tonnes was used in this project. Driving speed was set to an average of 60 km/hour.

5.3.3 Slaughter and Processing

The foundation for the slaughtering facilities was the same as in the project thesis, and was based on scaling a Hitra/Frøya production average, according to Chapter

5.4 in the preceding report. Hence, the capacity was set to 94 tonnes/ hour, which was the estimated average capacity of the three Hitra/Frøya facilities today, multiplied by five. The same applies for the processing facility. Hence, this value was used for all simulations carried out for a five-fold production, but is easily changed in the Excel spreadsheet for lower production volumes. Assuming production of 94 tonnes per hour, the time spent on each entity of 1 tonne, was calculated as follows:

$$\frac{1}{\left(\frac{94.79[t/h]}{1[t/entity]}\right)} = 0.01[h/entity] \tag{5.3}$$

This was also scaled with production volumes from the Excel input, as the facilities would necessarily have to grow with increased production volumes. The number of berths at each factory, can be altered from the Excel sheet. The waiting cage at Jøstenøya was modeled to have a capacity of approximately two commercial net pens; 1500 tonnes. This is the same capacity utilized at Innovamar (SalMar, 2017).

5.4 Crash Scenario Input

In order to expose the system to sudden and realistic disruptions, we decided to introduce emergency slaughter episodes and ban on open waiting cages according to the assumptions made in Chapter 3.2.

When emergency-slaughter was imposed, a specified number of salmon needed to be retrieved from the cages as soon as possible, once per year. The idea was, that it should be easy to adjust both the volume in question, and when the action needed to take place every year. This was done by changing the Excel sheet. Based on this, a set of vectors were created in MATLAB before simulation start; one for each farm. See Appendix A.1 under the headline *Emergency Slaughter*. The vector *SlaughterVec* initially contained a number of zero-columns equivalent the number of tonnes fish that was supposed to be emergency slaughtered each round, plus one extra to include an infinite time step at the end. The next step was to set the first column in the vector equal to the simulation time when emergency slaughter is imposed. This was a number between 4380 and 13140 hours, ensuring simulation has had time to stabilize up front, and was generated by a rand-function in the MATLAB script.

A binary variable in the Excel sheet determines whether the simulation should incorporate a ban on open waiting cages, or not. This was then picked up by the MATLAB script, and integrated in designated blocks in the model, see Section 6.2.2.

Input Influence

The previous sections presented the various inputs which were used in the simulations. As a model cannot be better than the information entering it, it is important to ascertain that the input was correct at all times. Each change made in the input variables requires a separate solution or a series of runs. Some input was thus not changed throughout the study, whereas others were altered between simulation runs.

Throughout the project, emphasis has been on getting as realistic data as possible, and contact was therefore made with a series of industry actors. The input for current data is thus a close approximation, whereas future estimates had to rely on some assumptions. The input and model construction needs to correlate to one-another, as they form the basis of all generated output.

CHAPTER 6

Model Construction

Model construction not an absolute right or wrong, but a process aiming to generate a true enough model. Campuzano and Mula (2011) have provided a list of guidelines for simulation-based modeling, which has been essential throughout the whole modeling process:

- Aim not to construct a complicated model, but a simple one that works.
- Understand problem modeling in order to find a suitable technique.
- Models must be validated before applying them.
- A model must never be put under pressure to do, or be criticized for not doing, that which it has never been devised for.
- A model cannot be better than the information entering it.
- A model must never be considered literally.
- Models cannot replace decision makers.

The simulation model presented in this chapter was built from scratch, based on model construction knowledge acquired during the project in the fall of 2016. The models which were developed in relation to this, were based on a preliminary model built in the course TMR4565 the same fall.

A model is only as strong as it was built, and must be validated before being relied upon. Bearing these in mind, the model was constructed step-wise, with the initial skeleton as the logical, chronological representation of the system shown in Figure 6.1.

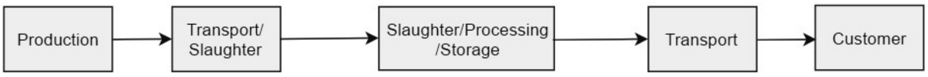


Figure 6.1: Initial simulation model skeleton.

With time, the model was supplemented with more and more features to replicate the real-life system, as presented in Chapter 2.2. To verify model performance, all runs were checked for scope outputs which deviated from expected values. This was done continuously during model construction, where simulation output needed to be in line with the reference data.

6.1 Simulink Supply Chain Modelling

The final model was a product of several reconstructions. The Simulink model was built of blocks with predetermined functions in the graphical drag-and-drop interface. These are generic queues and servers until the are given a purpose in the system. The challenge of the developer was thus to interpret and mold these functions to represent reality, and at the same time provide sufficient information on system performance and throughput. This was initially a struggle, and error messages were frequent and confusing. However, it also allowed for creative developments, and upon familiarization, new possibilities arose.

Figure 6.2 shows the conceptual framework for the finished model, and represents one of many possible ways to approach the problem. Other developers could

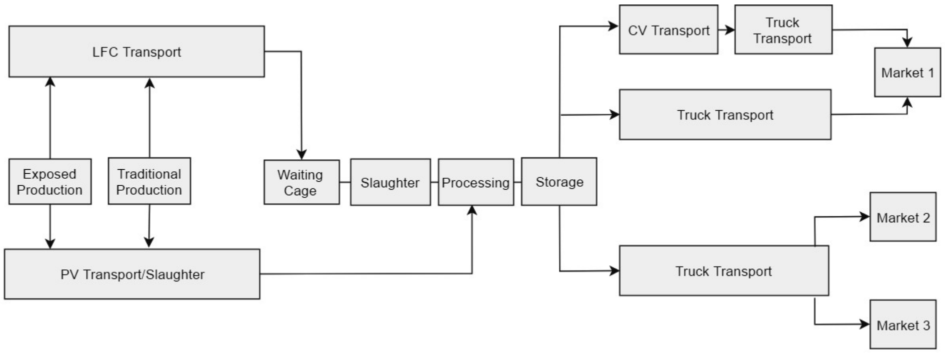


Figure 6.2: Conceptual framework of final simulation model.

have created an entirely different model. It thus becomes especially important to validate and evaluate results. This was done in Chapter 8. The complete model is shown in Appendix C.

The rest of this chapter is devoted to explaining in detail how the different system parts were molded. Some of these elements made out blocks which did not represent a specific physical supply chain feature, but rather served to the exactness of the model, and are called null-blocks. This included changing attribute values, evaluating storage capabilities, deciding paths, etc., and are explained with their respective systems.

6.1.1 Global Data Stores

In order to keep track of generated fish, queue lengths, weather conditions and inventory levels, global data stores was utilized throughout the model construction. Attribute values, assigned to different entities, was written to global variables with the use of “Data Store Write” blocks, upon which they could be accessed from any part of the model with the equivalent “Data Store Read”. This means that the data store values could be queried and changed, allowed different model parts to communicate with each other, and gave an opportunity to monitor the attribute values at respective sections. Figure 6.3 shows the set of *Simulink functions* and global variables used in order to model the exposed fish farm.

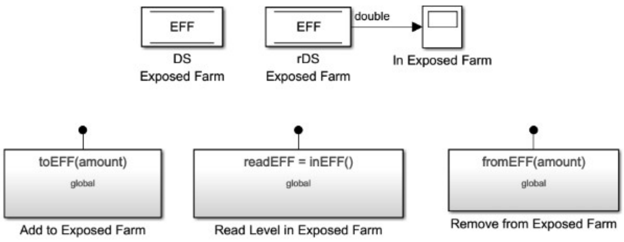


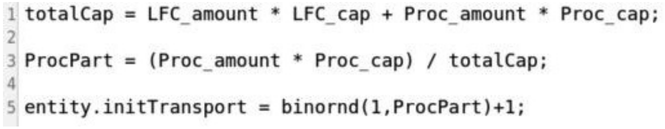
Figure 6.3: Data store function blocks for exposed fish farm.

6.1.2 Fish Production

Following the production according to Chapter 5.2.1, the produced fish were sent through a null-server. This had a service time of 0, and only served to record the passing fish to a global memory storage "Traditional Farms".

The next block combined the lane with any predetermined volumes of emergency slaughtered fish. These were generated according to Section 6.2.1, and sent to a null-server immediately after in order to write the volumes to two global stores; one for emergency slaughter fish volumes, and one for the traditional farms.

In the proceeding queue, the attribute value *initTransport* was altered. This determined whether the entity was sent to the live fish carrier or processing vessel. This was done by binomial distribution, with probability of each vessel type equivalent to their accumulated capacity relative the other, as seen in Figure 6.4.



```
totalCap = LFC_amount * LFC_cap + Proc_amount * Proc_cap;

ProcPart = (Proc_amount * Proc_cap) / totalCap;

entity.initTransport = binornd(1,ProcPart)+1;
```

Figure 6.4: Action code for binomial distribution between live fish carriers and processing vessels.

The fish entities were then divided according to their respective attribute *initTransport*-values by an entity output switch. The whole generation sequence is shown in Figure 6.5, and was identical for both traditional and exposed farms, save for the generation frequency.

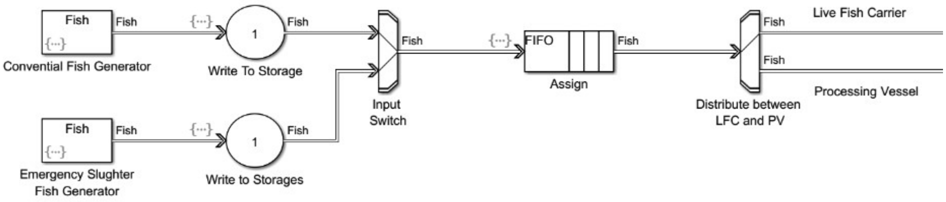


Figure 6.5: Block sequence of fish generation in Simulink.

Next, the fish was batched by an ”entity batch creator”-block. This batched the fish equivalent of one live fish carrier or processing vessel together to one entity, allowing it to later be combined with a vessel in a 1-to-1 relationship. These then proceed to an entity queue, where they waited for an available vessel, which were divided into one cycle with processing vessels, and one with traditional live fish carriers. See Figure 9.7 in Appendix C for the whole sequence.

6.1.3 Weather Conditions

In order to implement weather restricted operations by the fish farm and vessel transit delays, weather data, represented by significant wave height, were generated. Three almost identical MATLAB codes *mcwave.m, mcwave_trad* and *mcwave_exp* were reading specific wave heights at three different locations every three hours, as presented in Chapter 5.2.2, and writing them to entity attribute values generated at the same time.

The codes were originally created by engineer Knut Støvler in conjunction with course TMR4565, and was modified to fit input files, and output purposes. *mcwave.m* can be found in Appendix A.2. Assuming that the environmental and operational conditions for the system were relatively stable, the codes use Markov chains to model the transition rates between a set of sea states, according to theory presented in Chapter 4.5. The states were based on historical data, and makes it possible to predict future weather developments based on the current weather.

Weather conditions generated for the traditional cage locations will be used as example for the rest of this section. The method will be similar for the exposed locations, and for the cargo vessel sailing legs.

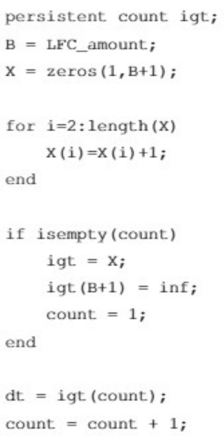
The code *mcwave.m* retrieves a vector consisting of two years of wave data from an Excel sheet. The wave heights were divided into ten even sections, representing different states. Transition rates between the states were found, and set into a normalized matrix. After checking for absorbing states in the matrix, it was decided how many state transitions to perform, and where to start the generation.

In Simulink, a digital clock was updated for every time unit, setting the time series to a string of integers, increasing proportionally to the simulation time. This was later accessed through the function $y = currentTime()$. The function *tradwaveState(currentTime())* used the pre-processed sea state time series to generate wave variables for Simulink, which has the same dimension as the aforementioned $states$-vector.

Using the same procedure for the exposed locations and the Norwegian sea, a data series of the associated significant wave height were written to global variables, using the MATLAB functions *plot_trad_seaStates(x)*, *plot_exp_seaStates(x)* and *plot_CV_seaStates(x)*.

6.1.4 Vessel Generation

The generation of vessels has to be done somewhat unorthodox in order to utilize the Excel and MATLAB input. Intergeneration time was thus coded to first develop an array of zeros with length the same as the desired fleet, plus one. The array was then looped to comprise all 1s instead of 0s. This allowed the fleet to be produced equispaced with one hour. The last column then has to be set to infinite in order to stop further production of entities. The vector generation only happens the first time the code was run, and the array was named ”igt”, upon which the the time until production of the next unit was read by the ”count” index. ”Count” and ”igt” were held persistent, meaning that they maintain their values within the block and between entities. By adding 1 to the current ”count” variable, the simulation will thus initiate the next generation after a given time according to the next value of ”igt”. The code can be seen in Figure 6.6, and was the same for live fish carriers, processing vessels and cargo vessels, save for the variable B which was gathered from the respective MATLAB variables LFC_amount, $Proc\_amount$ and CC_amount.



```
persistent count igt;
B = LFC_amount;
X = zeros(1,B+1);

for i=2:length(X)
    X(i)=X(i)+1;
end

if isempty(count)
    igt = X;
    igt(B+1) = inf;
    count = 1;
end

dt = igt(count);
count = count + 1;
```

Figure 6.6: Action code for generating live fish carriers.

Live Fish Carrier and Processing Vessel Round-Trip

A schematic of the vessel round-trip is shown in Figure 6.7. The figure does not include queues, null-servers or code connected to the blocks, and is for illustrative purposes only, however it represents the backbone of the vessel cycle.

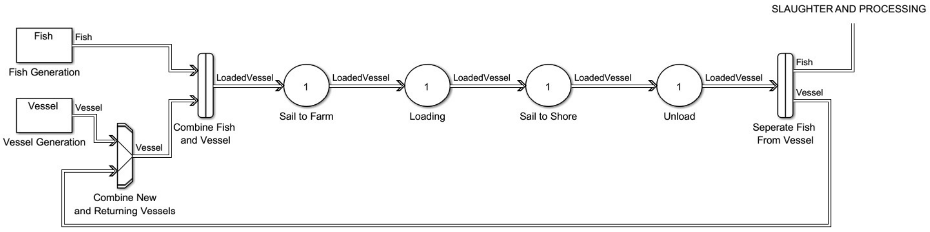


Figure 6.7: Simulink schematic of vessel cycle.

In the full model, the sailing to farms was divided into two parallel circuits, one servicing exposed farms, and the other to the traditional sheltered locations as seen in Figure 6.8. These paths also incorporate two additional null-servers, which add downtime as per the Excel sheet, holding the vessels back for a set amount of time, spread over the trips throughout the year.

For every new round-trip, the vessels first decide whether to serve the conventional farms or the exposed site. The decision was made based on the amount of vessels waiting to load at each respective location. This was done by writing each vessel in the queue to load to a global data store, and removing it when it has departed. If there were the same amount of vessels in both queues will the exposed location be prioritized due to its higher likelihood of first cargo. Location and associated sailing legs were chosen from the vectors presented in Section 5.2.3. A set of attributes keeps track of where the vessels were located at all times (farm site or factory) and calculate the next sailing distance based on this position. As explained in Chapter 3.2, the production volumes for all the conventional farms were combined into one generator. Hence, the choice of conventional farm site was randomly chosen from the set presented in Chapter 5.2.3.

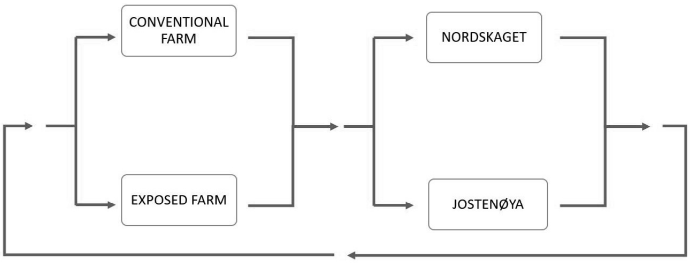


Figure 6.8: Live fish carrier and processing vessel routing.

The operation window by the farms was regulated by entity gate blocks. The release gates advances one vessel for each message that arrives on the count port. This signal was sent from an independent subsystem, which through the entity server “Trigger-Generator” checks if the weather window was wide enough for the operation to take place. This was done by using the function call blocks *tradwaveState(current time ())* and *expwaveState(current time ())*, which extracted the current wave height at the traditional and exposed farms. By adding the loading and maneuvering times *LFC_moor* and *Proc_moor* to these functions, e.g *tradwaveState(currentTime () + LFC_moor)*, one also retrieve a ”weather forecast”.

It was assumed that there would not be any significant changes in the weather conditions throughout this window.

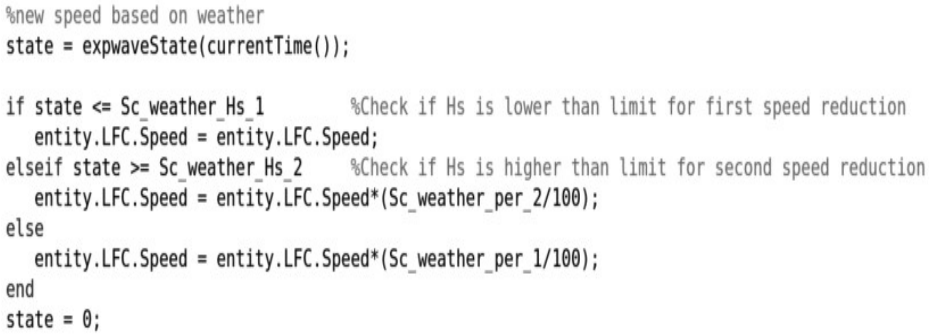
After loading, the two paths then merge, passing through a server which sets a sailing destination. The choice was made based on the amount of fish already waiting for slaughter at the Jøstenøya facilities. If this exceeds 1500 tonnes, which represents the volumes of approximately two waiting cages, and the regional companies were collaborating on the production, the vessel was sent to Nordskaget for unloading and processing. At their respective harbours the vessels were unloaded; simulated by a server, and separated the composite vessel and fish cargo in a composite entity splitter. No cooperation on the other hand, meant that the Jøstenøya factory needed to handle the production alone. This can be seen in Figure 9.9 in Appendix C.

In order to manage to number of berths available for unloading at the two factories, two resource pools were defined. A set of predetermined berths were assigned to each factory, which entities upon arrival can acquire, use, and then release. If no berth was available at the time, the vessels will queue up and wait.

Before every sailing leg a set of sever- blocks generates the sailing time. Vessel vulnerability regarding weather conditions was easily regulated from the Excel input. The time spent in the servers was a product of the generated distance to location, or to factory, the speed attribute, and the current sea state, where higher waves required reduced speed. Wave height was measured with *tradwaveState(currentTime())* and *expwaveState(currentTime())*, and bad conditions was implemented as an inhibitory factor to the equation, as shown in Figure 6.9. Note that an entity gate block only releases vessels to enter this sailing-server if the weather forecast was not to bad.

Cargo Vessel

The cargo vessels were generated in the same way as the live fish carriers and processing vessels, see Figure 6.6. Speed-, capacity- and distance attributes were set according to MATLAB values, as presented in Chapter 5.3.2. The capacity of



```
%new speed based on weather
state = expwaveState(currentTime());

if state <= Sc_weather_Hs_1        %Check if Hs is lower than limit for first speed reduction
   entity.LFC.Speed = entity.LFC.Speed;
elseif state >= Sc_weather_Hs_2    %Check if Hs is higher than limit for second speed reduction
   entity.LFC.Speed = entity.LFC.Speed*(Sc_weather_per_2/100);
else
   entity.LFC.Speed = entity.LFC.Speed*(Sc_weather_per_1/100);
end
state = 0;
```

Figure 6.9: Action code for speed reduction during live fish carrier transit due to weather influence.

the ship depended on whether the companies cooperate on filling the vessel, see Section 3.2. This was done by multiplying 0.33 to both vessel capacity, batch size, and quay storage limit during cooperation mode; hence ensuring a equal three-way partnership between farmers.

After merging with ready batched fish from the storage, the vessels enter a server which represents loading and maneuvering activities at Jøstenøya harbour. The next step was sailing to Esbjerg Port, where vessel and cargo split. The vessel will after unloading sail back towards Jøstenøya, whilst the fish will enter a storage waiting for further transportation with trucks.

In order to regulate the liner traffic to Denmark, an entity server “CV Trigger-Generator” checks if the weather window was wide enough for the sailing to take place. This was done using the function call blocks *CVwaveState(current time ())*, see Chapter 6.1.3. A corresponding entity gate block will advance one vessel for each message that arrives on the count port. The wave height limit for cargo vessel operation was regulated from the Excel spreadsheet. The time spent in the ”sailing”-server was a factor of the distance- and speed attributes, and the current sea state, where higher waves require reduced speed. Specific wave height for the area was measured with *CVwaveState(currentTime())*, and bad weather was implemented as an inhibitory factor to the system, equal to the process shown for the live fish carriers and processing vessels in Figure 6.9.

6.1.5 Slaughter and Processing

As described in Chapter 2.3.2, two processing facilities were modeled; Jøstenøya which represents the number one priority factory, and Nordskaget, which was a representation of the additional capacity available during cooperation periods. As both Nordskaget and Jøstenøya were identically simulated, but with different priority and capacity, this chapter only describes one of the two parallels.

The fish carried alive has to first be taken to a waiting cage and allowed time to calm down. This was simulated by a server with service time of 24 hours, the minimum actual requirement. The fish then proceeds to a queue, which in reality would still be the waiting cage, however, the simulated queue allows the fish to advance once the proceeding block was clear.

The next step was to move into the actual facilities, simulated by a server, where the fish was slaughtered. After the fish has been slaughtered it continues to processing. This was also where the load from the processing vessels rejoin the system. Slaughter- and processing capacity was set according to Section 5.3.3.

6.1.6 Transportation to Market

The next part of the chain was to transport the finished processed fish to the end user. Today, this is mainly done by trucks to the final destination, or to an airport for further distribution, but will also comprise regular freight with cargo vessels in a parallel loop, allowing for the transportation of greater volumes at a time. To model this cooperation between transport modes, we developed the following flow chart based on the distribution found in Section 5.2.4 and discussions regarding liner scheduling with Blått kompetansesenter AS (pers.comm., 3 March, 2017):

The transit between nodes in the left column of Figure 6.10 were modeled as entity servers rather than complete truck round-trips. Market distribution ratios were written from Excel to MATLAB as a set of variables: *M_TotWest, M_TotEast, M_TotNordic* etc. After the fish is processed, the simulation assigned the product entity a random attribute value between 0 and 100. Entities with an value be-

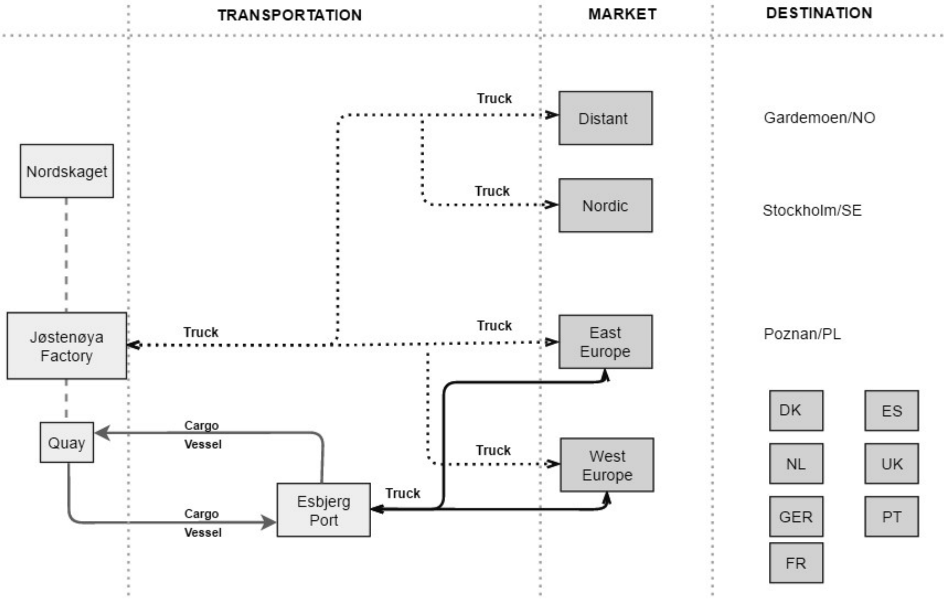


Figure 6.10: Flow chart of product distribution to market with associated transportation modes.

tween 0 and *M_TotWest* were sent to Western Europe, *M_TotWest - (M_TotWest+ M_TotEast)* to East Europe, *M_TotWest + M_TotEast - (M_TotWest+ M_TotEast + M_TotNordic)* to the Nordic countries, and the rest to distant markets. An illustrative description of the distribution can be seen in Figure 6.11.

As described in Section 4.3, were destination attributes were impossible to access after the fish was batched to fit the capacity of its transporter. Thus, in order to maintain the right distribution of goods to the markets, as described in Section 5.2.4, the fleet of trucks had to be split into multiple subsets prior to transporter loading, to ensure a correct market distribution. Export shares sent to destinations in Sweden and Finland, and via airports to distant markets, were not applicable for seaborne transportation to Esbjerg. Hence, these volumes were exclusively handled by trucks from Jøstenøya. The rest was divided between two passages; directly with truck to East- and West Europe, or to a quay storage to wait for cargo vessel pick-up. The associated Simulink sequence of blocks can be seen in Figure 6.10, where entity multicasts sent copies of the input entity to entity queue blocks configured with the same tag as the former block. The yellow servers were used

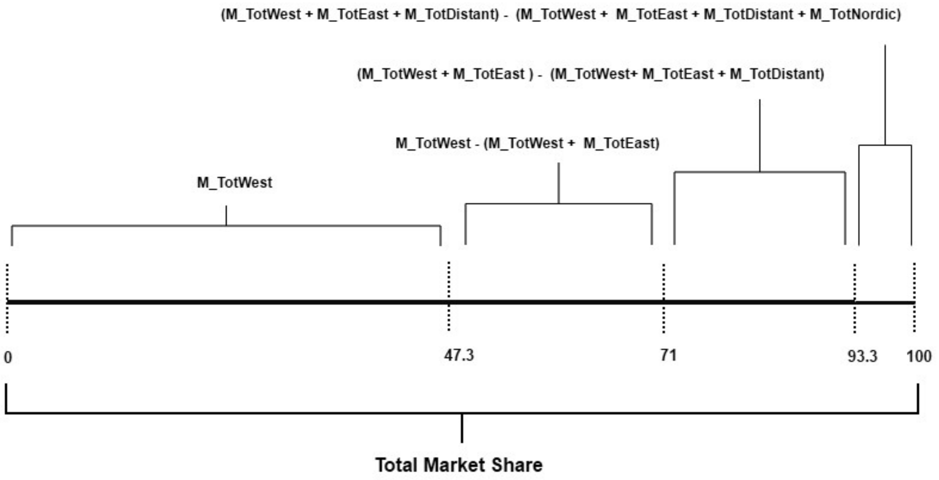


Figure 6.11: Illustrative description of how market distribution is implemented to attributes in Simulink according to MATLAB parameters.

for tracking attribute values in the Simulation Data Inspector, see Section 6.3.1.

An entity release gate controlled the flow of products according to the sub-system shown in Figure 6.13. A signal was sent to open the gate every time the quay storage was full, and when a cargo ship was returning towards Jøstenøya. Thus, batches of fish were sent to the quay storage to wait for vessel pickup solely when a ship was available for transport, avoiding accumulation of goods in the quay storage for longer time periods. Every fish batch was assigned an attribute called ”outputMode”, which was used for selecting an entity output port for departure; either by cargo vessel or by truck, in block ”decide Transport mode” in Figure 6.13. The quay storage limit was set to the equivalent of one cargo vessel shipload, which was changeable depending on whether the companies collaborated to fill the ship or not. The variable *Sc_coop* was used to scale the vessel capacity, cargo batch size, and the quay storage limit according to cooperation.

The physical movement of the products by fork-lift from factory to storage was modeled as an entity server with one hour service time. After the movement, the fish entities were written to a global data storage ”QuayStorage”, before entering a FIFO queue to wait for pick-up. After cargo vessel loading, the associated vol-

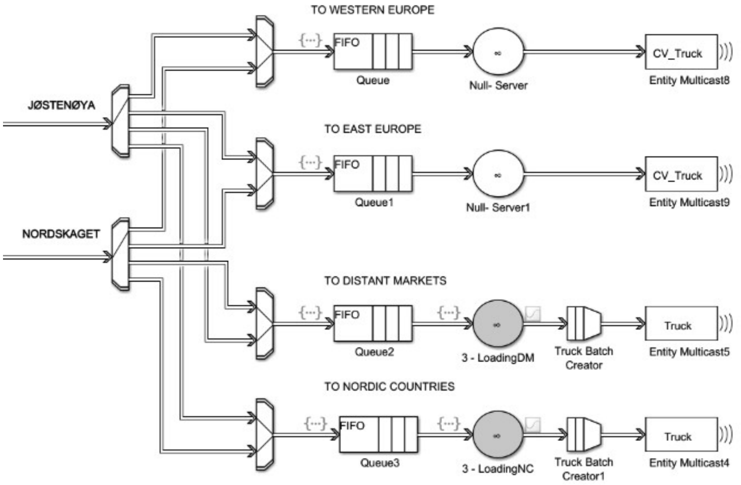


Figure 6.12: Simulink sequence displaying the distribution of processed fish from Jøstenøya and Norskaget, to various transporters with the use of multicast blocks.

umes were removed from the same data store, making it possible to monitor stock levels.

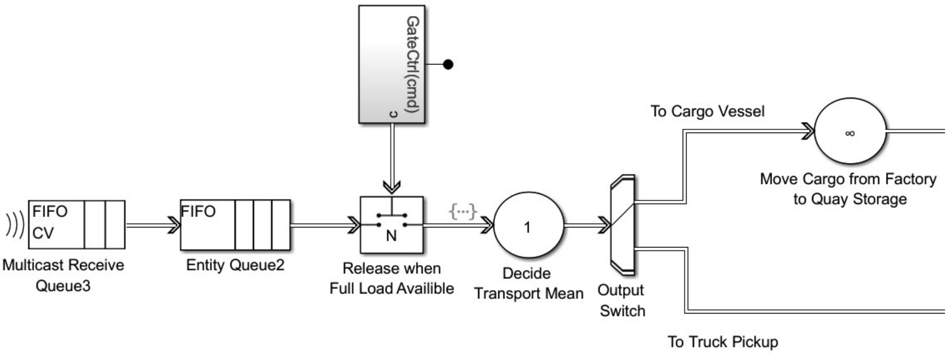


Figure 6.13: Simulink sequence regulating the flow of production volumes to quay storage and truck pick-up.

Truck Generation

Trucks were modeled in three round trips as defined in Section 5.2.4, 5.3.2 and according to the schedule described in Figure 6.10.

The trucks were generated with four attribute values; *speed, capacity, distance* and *destination*, where the first two were retrieved from MATLAB. If there was a sufficient level of fish available at the processing facilities at Hitra, or at Esbjerg Port, a composite entity creator block allows one truck entity to pass and merge with a newly created batch of cargo at the respective locations. The next server simulates loading times at the factory, but also provides a new cargo distribution according to Section 5.2.4, as the fish destination attribute was unavailable in a batched state. For the trucks which were going to West- and East Europe, this distribution also comprise a set of European countries which can be seen in Section 5.2.4. Truck entities were assigned attribute values accordingly, and sent to unload at their destination before returning. Figure 6.14 shows one of the three truck route simulations.

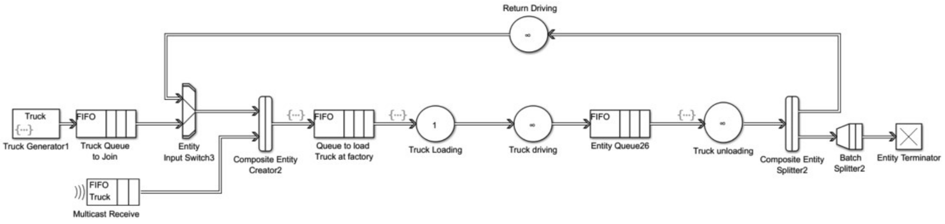


Figure 6.14: Simulink sequence of truck round-trip.

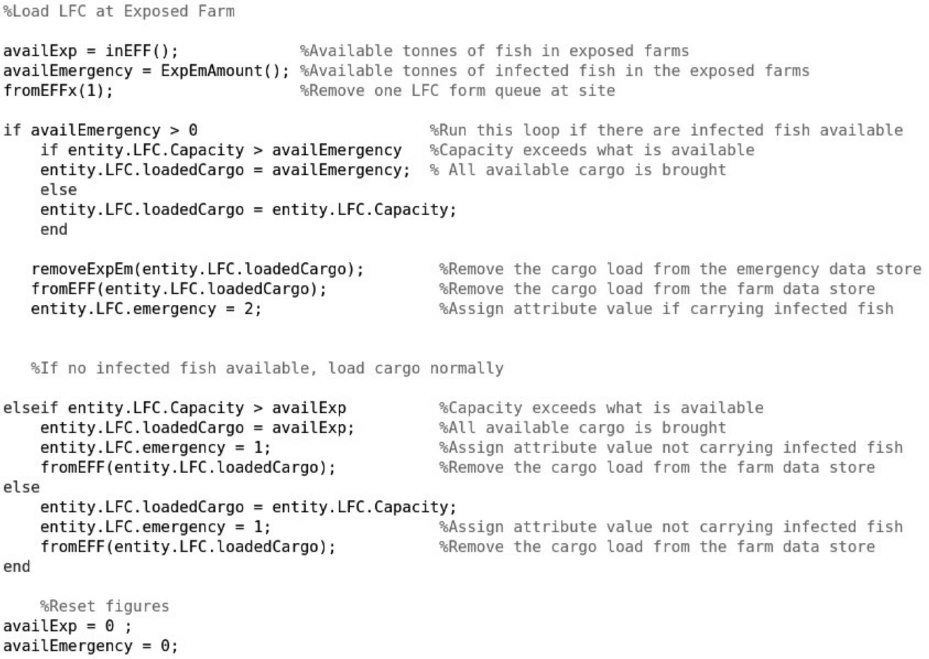
6.2 Crash Scenario Construction

Into the described system two shock scenarios were imposed. These were supposed to destabilize the system temporarily, and the effect had to be easily measured. The modelling of these should, however, not influence the model when the scenarios are not included, and the model thus has to check for implemented shocks.

6.2.1 Emergency Slaughter

The emergency scenario was implemented in the system using several blocks. A separate entity creator generated infected fish entities, using intergeneration times from the vectors created in Chapter 5.4. Upon creation, the fish entities were included in the cargo loop, and simultaneously sent signals to two data stores: ”Traditional Farm Emergency” and ”Traditional Farm” or ”Exposed Farm Emergency” and ”Exposed Farm”, see Figure 6.5. The volumes of emergency fish were thus included in the stocks of ready grown fish, and simultaneously monitor the tonnes of fish which needed to be processed as soon as possible.

Subsequently, for every live fish carrier and processing vessel round-trip, the vessels checked with the use of ”Data Store Read” blocks whether there were volumes of fish at the traditional- or exposed farms that needed to be retrieved. If this was the case, the vessels were assigned attribute values which led them to the respective farm through the entity switch block. The vessels were modeled to prioritize whichever site that has the highest numbers of infected fish. Arrived at the farm site, the model once more checks whether there are any tonnes of infected fish left in the data sores. If yes, these are loaded, and the cargo load is then removed from the two data stores as seen in action code in Figure 6.15.



```
%Load LFC at Exposed Farm

availExp = inEFF();             %Available tonnes of fish in exposed farms
availEmergency = ExpEmAmount(); %Available tonnes of infected fish in the exposed farms
fromEFFx(1);                    %Remove one LFC form queue at site

if availEmergency > 0                         %Run this loop if there are infected fish available
    if entity.LFC.Capacity > availEmergency   %Capacity exceeds what is available
    entity.LFC.loadedCargo = availEmergency;  % All available cargo is brought
    else
    entity.LFC.loadedCargo = entity.LFC.Capacity;
    end

   removeExpEm(entity.LFC.loadedCargo);        %Remove the cargo load from the emergency data store
   fromEFF(entity.LFC.loadedCargo);            %Remove the cargo load from the farm data store
   entity.LFC.emergency = 2;                   %Assign attribute value if carrying infected fish


   %If no infected fish available, load cargo normally

elseif entity.LFC.Capacity > availExp          %Capacity exceeds what is available
    entity.LFC.loadedCargo = availExp;         %All available cargo is brought
    entity.LFC.emergency = 1;                  %Assign attribute value not carrying infected fish
    fromEFF(entity.LFC.loadedCargo);           %Remove the cargo load from the farm data store
else
    entity.LFC.loadedCargo = entity.LFC.Capacity;
    entity.LFC.emergency = 1;                  %Assign attribute value not carrying infected fish
    fromEFF(entity.LFC.loadedCargo);           %Remove the cargo load from the farm data store
end

    %Reset figures
availExp = 0 ;
availEmergency = 0;
```

Figure 6.15: Action code for loading live fish carriers at exposed farm.

As the infected fish could not be released into the waiting cages, every live fish carrier used for the transportation needed to be assigned an attribute value, which decided which production line their cargo should use after unloading. Figure 9.9 in Appendix C shows this division of cargo after vessel deposit at the factories. Additional time used for cleaning of the on-board tanks etc. are assumed included in the ”annual-downtime” variable set before simulation.

6.2.2 Waiting Cage Prohibition

A simple signal from MATLAB determined whether the use of waiting cages should be impossible. As the processing vessels skipped this part of the chain regardless, model alterations were only necessary to include in the live fish carriers loop. These were, in the case of a ban, instructed to stay for at least 24 hours at the factory, allowing the fish to calm down, and ensure a controlled pumping of cargo ashore. This was thus the only section of the model which was affected by the restriction, and the associated code implemented in the unloading server at the factories can be seen in Figure 6.16.

```
%Decide unloading time for LFC at Jøstenøya

if Sc_waitcage == 1     % Sc_waitcage is retrieved from MATLAB
    dt = 24;
else
    dt = LFC_moor;      % LFC_moor is retrieved from MATLAB
end
```

Figure 6.16: Action code for unloading live fish carriers at Jøstenøya, where a waiting cage ban implies longer operation time.

6.3 Assessment of Simulation Results

Keeping in mind the list of guidelines for simulation-based modeling provided by Campuzano and Mula (2011), the model was constructed and ready for running the simulations. Simulink provides statistical output for the queue and server blocks, such as average waiting time, number of entities in block and utilization.

These signals were sent to a signal scope, which will plot the values against the simulation time. The statistics can also be sent to the MATLAB workspace by connecting a “To Workspace”-block to the scope output, as seen in Figure 6.17. These write a timeseries to MATLAB, which corresponds the amount measured in Simulink. The statistics were only available for the simulation entities, and not corresponding attributes.

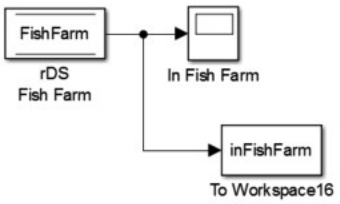


Figure 6.17: To Workspace-block writing scope results.

These scope blocks and their associated output plots were the initial validation. More than 300 individual scope blocks were added to the model, all used to provide an understanding of entity movement. The first step upon running a model was to check whether paths which should be empty were indeed, and paths which were supposed to be utilized were used. In the opposite case, scope blocks down the path were tracked, conclusively finding the source of error, typically a block which withholds entities, or an incorrect reading from a data store. These could then be altered, and the simulation run again. This intuitive approach provided a significant part of the model validation, particularly during development when the communication between blocks needed to be established.

6.3.1 Time Tracking

As mentioned in the introduction, is the time prospect particularly important in this thesis. Tracking entities throughout the system would provide valuable information about the movements in the system, however, this is not a feature which has been integrated in Simulink, and finding a way posed a challenge. Because the MATLAB-Simulink variable-communication is a one way street, it was difficult to write and store information continuously during simulation. However, by providing the fish entities with a series of time attributes, and assigning these values according to the current time at the point of passing a block, the information could be, although somewhat extensively, obtained through the data simulator inspector presented in Section 4.4.1. Each entity was thus provided attribute values named *interTime0, interTime1,* etc., six attributes in total. These all had an initial value of zero.

The measuring point had to be assigned carefully, as not all block types were compatible with the handling of attributes. Entity batch splitters, for instance, provided new data for each batch, and made the results impossible to compare. A queue cannot write new attribute values upon departure, and batched entities cannot alter attributes for single entities. Thus most of the measure points were set to servers with infinite capacity and zero serving time.

The selected point of interest were:

1. Produced fish waiting for vessel pick-up.
2. Reaching land (before the waiting cage from live fish carrier, before processing for processing vessels).
3. Upon completed processing.
4. Loaded onto respective means of transport.
5. After shipment (only the seaborne units).
6. Reaching destination.

The specific blocks are shown as the dark colored blocks in the model display in Appendix C. The attribute values in each block can then be plotted in the Simulation Data Inspector, displaying the time each entity has spent from slaughter to the various measure points.

Upon reaching the predetermined blocks, the attribute value was altered. The new value stayed with the entity throughout the model. This allowed for reading the values in simulation data inspector. The values were given by subtracting the time of arrival from the time of departure from the waiting cage in the beginning of the model. For measure point 4, between the cargo carrier and proceeding trucks, this entailed the command *entity.interTime4 = currentTime() - entity.interTime0*.

Model Relevance

In light of the limitations and restrictions mentioned in earlier chapters was the thesis model developed. This strove to reflect a complex multimodal supply chain with numerous intersections, and was built from scratch in Simulink. The output is directly linked to the model construction, and it was important to develop a model which could make the best possible logistics decisions, and thus handle multiple types of input. The model can be found in its entirety in Appendix C.

After running the model, the program calculates a wide range of data. By tracking a set of predetermined system performance indicators for the various system runs presented in Chapter 3.1, one was able to acquire an understanding of the system. This had to be done through an output analysis.

CHAPTER 7

Results

In the following section the simulation results are presented for a series of different runs. Simulation results were only considered valid after they had time to stabilize by running continuously for 4830 hours, thus avoiding possible errors from the start-up phase. For this thesis, simulation results were run over a 1.5 year period, equivalent to 13140 hours, but only considered for the last year (8760 hours). All graphs will be shown for the complete run to show the complete development, but tabulated results were adjusted to only include the last year in order to avoid averages with two winter periods.

As explained in Chapter 3.1, will the first section of this chapter present results retrieved from steady operation, when the system is under continuous running, without interruptions. The second will present results regarding how the system handled production flow deviations.

7.1 Steady State

7.1.1 Normal Operation and Model Validation

In order to have validate the simulation, a model was run with input features which simulated the current market situation. This implied no cargo vessels or processing vessels, as well as no production from exposed sites or collaboration between actors. To analyze current operation the factory capacity was equivalent to today's current slaughtering capacity, 18.8 tonnes/hour. Based on the current industry standard, the live fish carrier capacity was set to 400 tonnes and speed at 12 knots (Møre Maritime AS, 2016). The number of berths at Jøstenøya which could be used for unloading was one. All the input data is summarized in Table 7.1. This model was used to validate the model structure, as it was the only run which can be compared to real-life data. The results are also interesting as comparison for later runs, which simulate the future.

Table 7.1: Input for normal operation simulation.

	Included	Not included	Comment
Live fish carriers	X		3 x 400 tonnes
Processing vessels		X	
Slaughter facilities	X		18.8 tonnes/h
Cargo vessels		X	
Trucks	X		200 x 20 tonnes
Collaboration		X	
Exposed production		X	
Berths	X		1 at Jøstenøya

Flow

To ensure continuous flow throughout the system, the first element which has to function is the collection of fish. Although there is no specific rush to collect the fish from the cages at a particular time, it is in the farmers interest to avoid accumulation of fully grown fish, as these are expensive to feed, and take up unnecessary space and man hours. This is ensured by checking that all generated fish is removed from the cages by a vessel, which can be seen in the data store which tracks the current level of fish in the cages at all times. The data store scope output from traditional farms is seen in Figure 7.1, and shows that there was never more than 400 tonnes - equivalent to the capacity of one vessel - available. This implies that there were sufficiently many live fish carriers readily available to collect the fish once they reached slaughtering size.

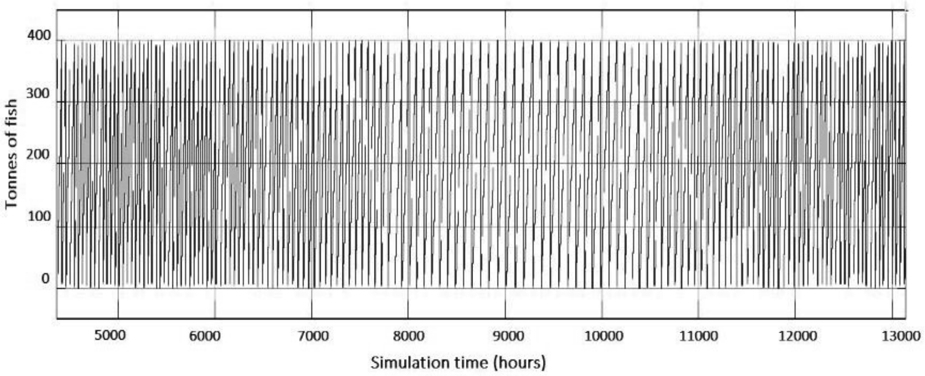


Figure 7.1: Tonnes of fish pending in traditional cages during current day production. The vessel fleet is ensuring steady pickup throughout the year, avoiding accumulation of fully grown salmon in the cages.

The ”vessels in queue” block output confirms this, and there is never more than one of the three vessels in use, see Figure 9.18 in Appendix D.1. An excess of 2 vessels ensured no shortage, but were unnecessary, and could safely have been reduced to a fleet of only one live fish carrier. This agrees with today’s operation, when one live fish carrier is typically used in one area. With only one vessel in circulation, there was also no queue to unload, and the transit to waiting cages is thus as efficient as possible. The wait until slaughter after waiting cage is on average almost 11 hours, as seen in Figure 9.19 in Appendix D.1, in practice this time would still be spent in the waiting cage, and is thus not critical. This time was induced because the slaughtering facilities needed to work through the entire incoming vessel load. At 18.8 tonnes/h it takes 21 hours to process all the fish, and the average time spent waiting was thus approximately half this time. It was therefore important that the fish carriers did not arrive with a higher frequency than 21 hours, as this would have started accumulation in the waiting cages.

After slaughter and processing, the entities were moved to their respective mode of transport, which should include no delivery to the ship quay, but the entity gate, described in Chapter 6.1.6, and shown in Figure 6.13, allowing entities to pass, still needed to let through all entities going to Europe by truck. This was controlled by checking the queue prior to the gate. As where there was never more than one entity at a time, the gate was concluded to function as intended, see Figure 9.21 in Appendix D.1. The amount of departed entities in the path to the cargo carrier circulation is also zero, ensuring that all cargo is directed towards the truck circulation.

Validation

Time to various measuring points is recorded as described in Section 6.3.1. The attribute value interTime0 records the time at which a particular tonne of fish reaches the waiting cage, and is the reference value to which all other time measurements are compared. This will thus be one straight line when processing vessels are not included, and all of the fish was proceed through the model chronologically. This can be seen in Figure 7.2, where the consistent slope of the line implies that the value assigned to the batch, and the time at which it arrives has perfect correlation, which proves that the model read the attribute values correctly. The frequency of the points represent the frequency with which an entity arrives. This is visibly lower between hours 8000 and 10 000, corresponding to the low production during winter months.

The time was measured from arrival at shore and to various intercepts, tracking the progress throughout the supply chain. The time it has taken the various batches to get from measure point 0 to complete processing varied between 25 and 48 hours, most of which came from spending at least 24 hours in a waiting cage, see Figure 9.22 in Appendix D.1.

The next measuring point was upon loading the fish onto trucks bound for their respective destination. This took approximately the same amount of time regardless of truck circuit, which implies that the different routes are simulated correctly and identically. The time from first arriving in the waiting cages has a mean of 42

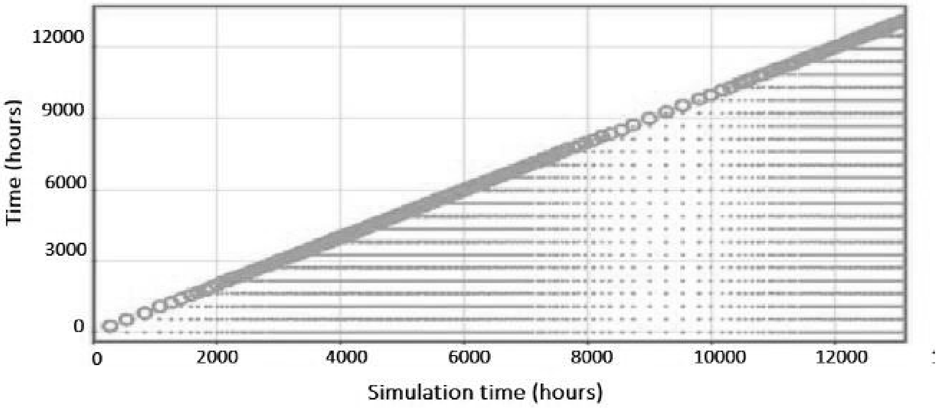


Figure 7.2: Time used by fish batches from pending in live fish carrier queue, until arrival at shore. The arrival frequency reduces during the winter months. The straight line indicates a correct value interpretation by the model.

hours, which correlates well with the time from completed processing, the exact values are shown in in Table 7.2.

Table 7.2: Time used by batches of fish until loading and unloading of trucks.

Market Segment	**Time to truck loading**				**Time to market**			
	Mean	Min	Max	IQR	Mean	Min	Max	IQR
East- and West Europe	42.5	25	47	8	80	49	117	18
Nordic Countries	42.2	25	48	8	55	38	66	6

The final time KPI was measured upon reaching market, which was recorded when trucks were unloaded. This is also shown in Table 7.2, and shows that the longer distance travelled to central Europe takes longer than to the Nordic countries and Gardermoen, which is reasonable, and indicate that the final transit is coded correctly. A mean time of 80 hours, roughly 3.5 days, to Continental Europe is safely within the 15 day limit for delivery. The time figures fit estimations from the industry, and the model was consequently assumed to track time correctly.

Additionally, it is important to check whether the market distribution is correct. This is one of few blocks which includes a degree of randomness, and is thus subject to slight variations. As per Table 5.1, the west, east, distant and Nordic markets should receive 47.3%, 23.7%, 22.3%, and 6.7% respectively. According to Table 7.3, these numbers are well replicated during simulation. This will require running the simulation for some time, allowing probability theory to spread the deliveries correctly.

Table 7.3: Simulated distribution of products to markets.

		West Europe	**East Europe**	**Distant Markets**	**Nordic Countries**
Arrived	[tonnes]	40 100	20 320	19 410	5 220
Share	**[%]**	47.1	23.9	22.8	6.1

7.1.2 Regional Cooperation

In order to evaluate how cooperation between farmers can influence the total production flow, a set of simulation runs were completed, with input and result according to tables presented in Appendix D.2. The cooperation is implemented by assuming three farmers collectively fill cargo each vessel. This in practice implies that the simulated farmer only has to fill a third of the total cargo volume. This will take less time, and a summary of the most important results can be seen in Table 7.4.

Table 7.4: Simulation input and mean output for tests conducted with respect to regional cooperation.

Run number.		**1**	**2**	**3**	**4**
Production factor		1	1	5	5
Cooperation		No	Yes	No	Yes
Quay Storage					
Amount	[tonnes]	850	822	621	249
Cargo Vessel					
Wait to load	[h]	514	185	48	4
Departures	[annually]	17	43	68	94
Mean time to					
Quay Storage	[h]	29	28	28	28
Truck loading	[h]	27	27	27	27
Esbjerg by CV	[h]	284	180	121	114
market by CV and truck	[h]	323	209	159	141

With current production levels was the accumulation of fish in quay storage substantial, as seen from run 1. Even collaboration introduced in run 2 barely reduced this number, although slightly better. For five-fold production was collaboration, however, a considerable contribution, reducing the average waiting stock by more than a third from simulation 3 to 4.

With time as the key factor, the storage waiting times become of particular interest. One of the concerns of the farmers is how long the fish has to be held, waiting to fill a vessel. The results show a substantial reduction in standby time by introducing cooperation. This becomes particularly important during the winter months, when accumulation is slower. As seen in Figure 7.3 was there always a vessel available, and the critical aspect is how long the first arriving cargo entities have to wait for the last. This waiting period for the cargo vessels is not critical to the simulation but will be costly off-hire to the ship owner.

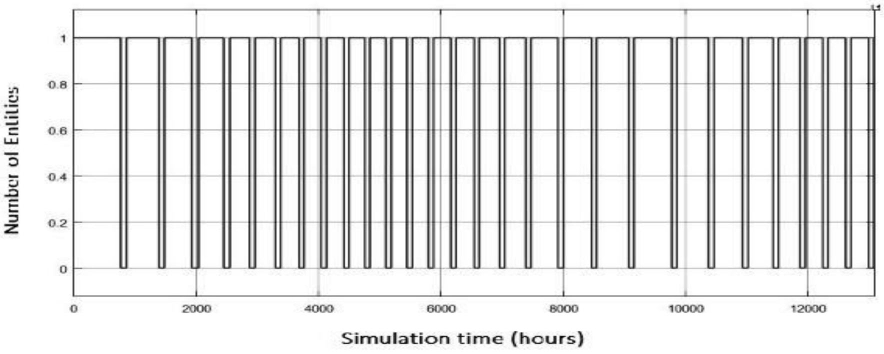


Figure 7.3: Number of cargo vessels waiting to load during simulation run number four. Time spent waiting increases during the winter.

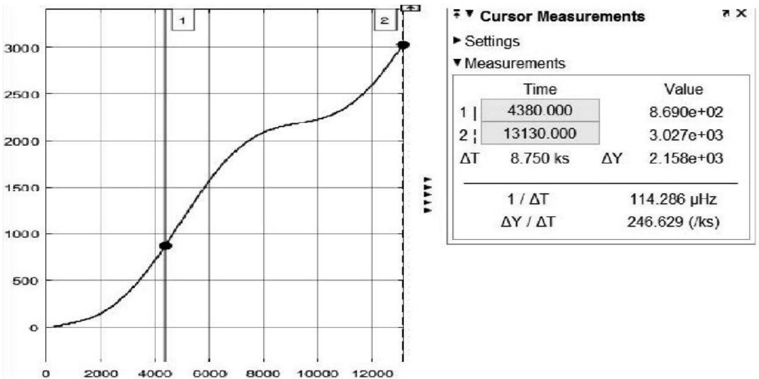
Table 7.4 shows that the mean time to quay storage and truck loading is unaffected by both cooperation and production volume. This indicates that the effect of cooperation does not work up-stream, and alterations in constellations will not influence the factory. For smaller production volumes will cooperation have a large impact on time to Esbjerg by cargo vessel, reducing the time by more than a factor of two. The larger volumes in runs 3 and 4 see a less significant difference. The same pattern propagates all the way to market delivery, because there are sufficiently many trucks, where cooperation has saved almost a day relative to a one farmer - one vessel solution.

7.1.3 Introduction of Cargo Vessel

Gradual development in every mode is efficient up to a certain point, after which a significant restructuring must be imposed. For this system, the natural next step is to implement cargo vessels in order to relieve the trucks and roadways. It is of interest to establish at which point this becomes favorable, in order to prepare for the future. Cooperation between three equal actors was assumed, based on the aforementioned results and clear signals from Blått Kompetansesenter AS (pers.comm., 16 March, 2017).

Current Production Level

Using one live fish carrier, according to findings in Section 7.1.1, at current production levels, the model was run without cargo carriers. This implied that a total of 2158 truckloads were transported to continental Europe over the span of a year, shown as ΔY in Figure 7.4. The simulation was run with 100 trucks in every circuit, however, at least 38 were in queue to load at all times. This implied that 62 trucks working on delivery to Europe were required to run the particular system without a shortage of trucks, see Figure 9.23 in Appendix D.1.



	Time		Value
1	4380.000		8.690e+02
2	13130.000		3.027e+03
ΔT	8.750 ks	ΔY	2.158e+03
	1 / ΔT		114.286 μHz
	ΔY / ΔT		246.629 (/ks)

Figure 7.4: The number of trucks departed from one factory at Hitra/Frøya, bound for continental Europe during one year, is 2158 for the current production levels.

The current truck system brought the produced cargo to market within 100 hours of arrival in the waiting cages, see Figure 7.5 and associated Table 7.5. These indicate a mean time of 74 hours, approximately 3 days. This is an acceptable figure considering the 15 day limit from slaughter to unsaleable. The standard deviation also implied that all values were fairly close to the mean, which the minimum and maximum support. This proved that the current system is suitable for handling by truck alone.

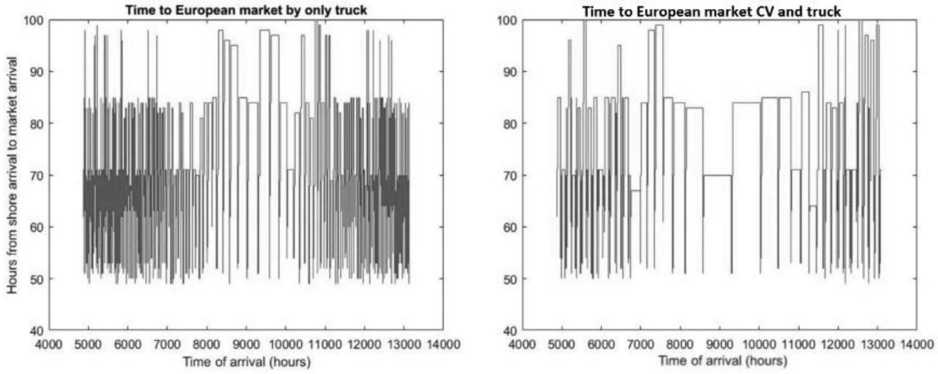


Figure 7.5: Time used from arrival at shore until reaching the European market by truck. Only trucks (left), in parallel with cargo vessels (right).

Upon imposing a 2000 dwt cargo vessel working in parallel with the trucks transporting cargo to central Europe, the image changed. Although only one third of the vessel needed to be loaded before departure due to the collaboration, the trucks became less frequent after being supported by a cargo vessel, see Figure 7.5. The mean time spent before being unloaded at destination was also slightly higher, although maximum and minimum values were the same, see Table 7.5. The required number of trucks also dropped, and only 392 departures annually were aimed for continental Europe, saving the roads for 1766 trucks annually, 33 per week, see Figure 9.26 in Appendix D.3,.

The cargo carriers, however, had trouble bringing the fish to its destination quickly enough, averaging at 208 hours, approximately 8.7 days. This is within the 15 day limit, and included 24 hours spent waiting in cages before slaughter, but still reduced the shelf-life of the product compared to letting the trucks drive the full

Table 7.5: Total time used until reaching final destination with use of various mode compositions.

Additional Info	**Transportation mode**	**Mean**	**Max**	**Min**	**SD**
No cargo vessel	Truck	74	100	49	14
One 2000dwt vessel	Truck	78	100	49	12
	Cargo vessel	209	855	89	105
One 1000dwt vessel	Cargo vessel	129	599	71	59

distance. The high maximum and standard deviation also imply that some of the cargo did not get delivered in time, and would have to be discarded.

Figure 7.6 shows the time before the cargo is loaded onto the cargo carriers (left) and after unloading (right). The plots show that the fish had not spent more than 31 hours from slaughter to reach the quay storage, but after unloading, the plot made a drastic jump, including batches which have taken more than 800 hours, 33 days, to reach Esbjerg harbor. The reason can be seen in the vertical points on each line, representing the simultaneous arrival of entities with significantly different arrivals to shore. Each vessel had to wait for approximately four separate deliveries (depending on size) from processing in order to depart, so the top points on a line will have waited for the lowest points. When production was low during winter months, it took longer for these to arrive, and the first to reach the quay had to wait a long time for the last. This is why the lowest points remain stable throughout, whereas the top vary with production. This forced the maximum time to grow during the winter months, making seaborne transportation virtually impossible, even with cooperation between producers.

By reducing the size of the cargo carrier to 1000dwt, and thus the amount which has to be available in order for it to depart, the waiting time was reduced, as seen in Table 7.5. The maximum wait decreased from 855 hours, to 599 hours. This was still above the acceptable limit, but with an inter-quartile range of 55, the bulk of the cargo was delivered within the scope.

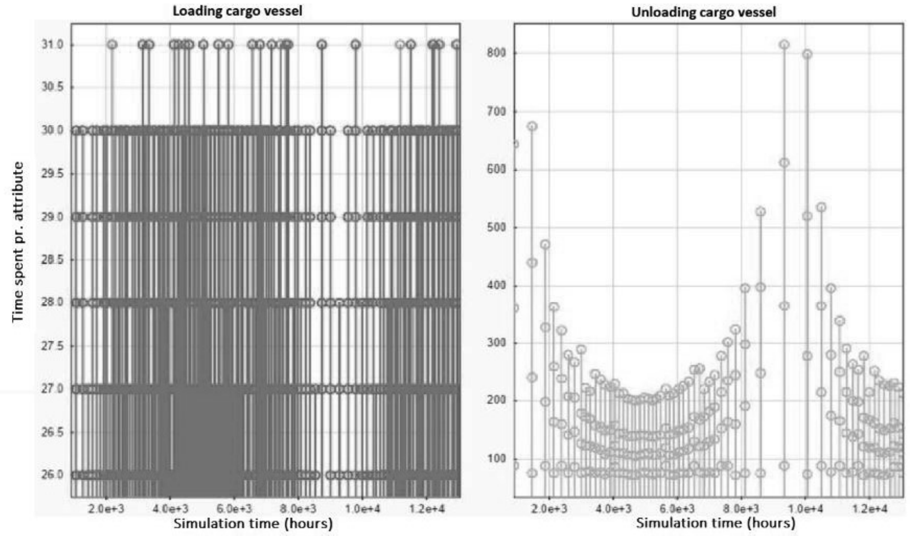


Figure 7.6: Time used until arrival at cargo vessel quay (left), and to arrival at Esbjerg Harbor (right), for current production levels. Low production during winter months influences the mean time spent to the Continent.

Increasing Production Volume

In order to establish how much the production needs to increase by before seaborne transportation becomes a realistic alternative, the simulation was run six times, at today's production level, and three and five times today's production. These were all run for both 1000 and 2000 dwt cargo vessels. The key figures from these runs can be seen in Table 7.6.

Table 7.6: Key time variables in six simulation runs increasing production volumes.

Production level	**CV capacity**	**Mean**	**Max**	**Min**	**Std**
	[dwt]	[h]	[h]	[h]	[h]
1x	1000	129	599	71	59
	2000	209	855	89	105
2x	1000	144	374	82	40
	2000	165	383	89	44
3x	1000	143	328	83	36
	2000	141	344	89	33
5x	1000	149	284	83	34
	2000	141	264	89	27

Based on the information gathered from the three runs, it seemed like 3x would support a transition to seaborne transportation with low accumulation and a predictable range. This narrowed down the interval of searching, and a fourth simulation was run at 2x production. This included 1000 and 2000 dwt vessels like the previous, and revealed the KPIs as according to Table 7.6. These show lower levels than 1x with the 2000dwt vessel. The 1000dwt have a higher mean, but lower standard deviation. This is because 1x has a higher variation between summer and winter periods, which causes seasonal fluctuation. However, it causes the winter months to pass quickly, which will be favorable to the mean.

At 2x will the cargo vessel almost never wait, and the cargo average a 18 hour wait, see Figure 9.27 in Appendix D.3. With a maximum at more than 15 days, the limits for delivery are being pushed, however, the low average and standard deviation implies that most of the fish will be safely within the margin. In other words; seaborne transportation is probably somewhat premature, but the pivot-point is imminent.

Table 9.1 in Appendix D.3 shows that with a production level five times as big as today's level, and one cargo vessel running in shuttle between the continent and Jøstenøya, the number of trucks which need to be serviced at the same time at Jøstenøya are 1.4 on average, and six at maximum. Without the vessel, the average number of trucks increases to 1.7, while eight trucks occasionally needs service simultaneously. Consequently, reducing the number for loading-slots for the trucks beneath three units, will most likely lead to insufficient flow of cargo out of the factory, regardless of the cargo vessel.

7.1.4 Weather Influence

During the course TMR4565 in the fall of 2016, the authors of this thesis developed several MATLAB scripts for identifying the average time during different seasons the farms were unavailable for vessel operation. This was based on a set Hs operating criterion, and the codes are attached in Appendix A.3. As the data used for the farm regions in this thesis are confidential, and the simulation of waves primarily helped to provide a dynamic subtone for the chain, corresponding availability-plots were not included in these results. The cargo vessel transit legs was, however, influenced by the generated weather, and time spent in reduced transit speed was thus evaluated using the same codes, and scopes from Simulink.

Figure 7.7 shows how the cargo vessel up-time varies according to season, and significant wave height limit. The plot is retrieved by using met-ocean data retrieved for the Norwegian sea, as described in Section 5.2.2, as input for the codes presented in Appendix A.3. The plot shows that by setting an Hs-limit of 9m, transport can be done constantly during summer season, and approximately 99% of the time during spring and fall. The time window for transport during winter is reduced by an additional 3%. These numbers are listed in Table 9.2 in Appendix D.4.

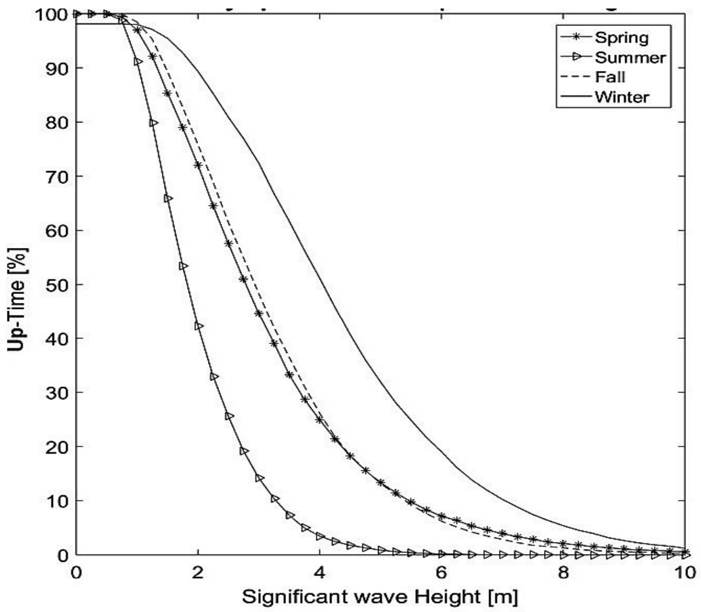


Figure 7.7: Cargo vessel up-time during transit, according to season and significant wave height.

Reading the figure, a 4 m Hs limit implies that 25% of every trip during the six spring- and fall moths needs to travel with reduced speed. The summer months, on the other hand, were only affected 3% of the time, and the winter 52 % of the time every month. Hence, for more strict operational criteria and smaller acceptable significant wave heights, the average transit time increases.

The probability that cargo ships should be delayed to a point where cargo needs to be discharged is however very small. Figure 9.28 in Appendix D.4, displays the simulated average time spent sailing to Esbjerg for one cargo vessel. It is very stable at approximately 41 hours throughout the year, and one can hardly differentiate between the seasons.

7.2 Crash Scenario Influence

The main objective of a production system is to have a flow which is as effective and robust as possible, relative to the effects that inputs have on the system (Asbjørnslett and Rausand, 1999). The following chapters will present the most prominent results acquired throughout scenario influenced runs, according to Chapter 3.1 and the vulnerability assessment presented in Chapter 4.4.2.

7.2.1 Emergency Slaughter

Emergency slaughter is a plausible situation for a modern fish farmer to be exposed to, and was introduced into the system by a sudden wave of additional fish generated, according to input values from Excel. These added fish had priority, and was collected first, but otherwise behaved like normal fish entities in the simulation. Six simulations with input as presented in Figure 9.29 in Appendix D.5 were conducted. The most prominent being as follows:

Table 7.7: The most prominent input for simulations testing emergency slaughter impact.

Run Number	**Unit**	**1**	**2**	**3**	**4**	**5**	**6**
Production volume	**[-]**	2x	2x	2x	4x	4x	4x
Factory capacity	**[tonnes/h]**	38	38	38	38	38	38
Emergency slaughter	**[tonnes]**	-	1000	1000	-	1000	1000

Simulations number 1 and 4 represented a basis for results comparison, at different level of production volumes and no emergency slaughter. Simulations 2, 3, 5 and 6 implemented emergency slaughter after 4830 hours, and number 3 and 6 also introduced processing vessels to the fleet. In order to highlight the juxtaposition between the supply chains, the number of cargo vessels, factory capacity and vessel specifications was fixed throughout the simulations. All simulations cooperated on production, and the level of emergency slaughter was the same in order to provide equal ground for comparison.

The initial transportation was maintained high in order to evaluate the effect on the subsequent operations. When the fish carrier fleet was large enough, the imposed shock propagated through it unaltered. In the first three simulations was the slaughter capacity proportional to production, whereas the last three operated on lower capacity relative production, and thus incorporated a bottleneck in the middle of the chain.

Time was used as a primary performance indicator, however the key to deliver on time was to avoid accumulation of entities in places where these would never be stored. Additional aspects of interest would be the need for borrowed capacity, and the influence of processing vessels on the production flow. Hence, a set of KPIs were selected, and Appendix D.5 presents the associated results gained through the simulations. The most interesting regarding production flow are shown in Table 7.8, with additional numbers listed in Figure 9.30 and 9.31 in Appendix D.5.

All numbers in Table 7.8 are retrieved from the time period 4380 to 13140 hours, except for the factory utilities. MATLAB was unable to re-sample the small numbers, and the utility presented was hence the mean number from the whole simulation time. They therefore only functioned as performance measurements relative the other simulation runs, but represented 1.5 years, January through June.

Both simulation 1 and 4 had low mean volumes in the farms, and approximately half the live fish carrier capacity of 400, indicating that they were usually serviced immediately after producing sufficient entities. This correlates to the large number of vessels. When emergency slaughter was introduced did, however, the large fleet not suffice, and there was accumulation of fish in both farm sites, and a higher mean level yet to be collected. The introduction of processing vessels in

Table 7.8: Selected outputs for simulations testing emergency slaughter impact.

Run number	Unit	1	2	3	4	5	6
Traditional Farm							
Mean Volume	[tonnes]	232	400	977	200	391	735
Factory							
Delivered to Jøstenøya	[1000 tonnes]	174	174	173	356	348	355
Delivered to Nordskaget	[tonnes]	0	0	0	0	8 000	0
Utility slaughter	[-]	0.3	0.3	0.1	0.7	0.7	0.3
Utility processing	[-]	0.3	0.3	0.3	0.7	0.7	0.7
Quay Store							
Mean Volume	[tonnes]	202	189	189	208	207	198
Time from processing							
to Quay Store	[hours]	33	32	21	31	34	24
to Truck Loading	[hours]	34	33	28	32	34	24
To Europe w/ truck	[hours]	78	77	71	73	75	64
To Europe w/ ship	[hours]	161	154	157	151	152	139

simulation 3 and 6 increased the mean level further. This was probably related to the somewhat arbitrary distribution between the initial transit fleets, risking longer waiting times for the first available vessel.

Figure 7.8 shows the development in the traditional cages over 13141 hours, emergency slaughter being imposed at 4380 hours. The graph shows that the system was quick to re-establish a new normal, taking approximately 150 hours to find a steady state. This was slightly elevated, because the shock had thrown it off balance of the correctly sized batches, and by picking up the cargo immediately, the cargo was not allowed to reach past 1.5 vessel loads. Corresponding plots for the other simulations can be found in Figure 9.32 in Appendix D.5.

The delivered volumes to Jøstenøya were steady across the respective production volumes; simulation 1 relative number 4, simulation 2 relative number 5 etc. The volumes were however slightly more than double for the last three simulations relative to the three first, indicating that the large volume flow was steadier with a higher production.

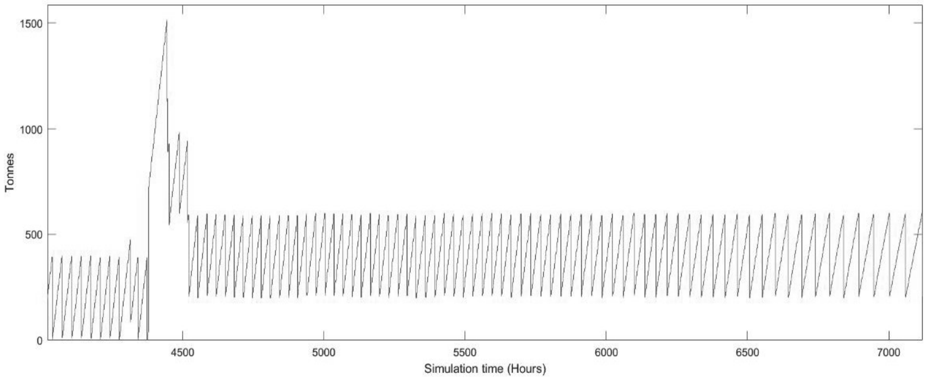


Figure 7.8: Tonnes of fish pending in the traditional cages during simulation number two. After a sudden volume increase, the system needs 150 hours to reach a new stable state, which, due to simulation restriction regarding vessel loading, is 200 tonnes higher.

The 5th simulation was the only one where the waiting cage at Jøstenøya was filled to the point where capacity had to be borrowed from Nordskaget. This makes sense given the high efficiency in delivery, and reduced capacity on land, forcing accumulation in the waiting cages. Figure 7.9 shows the level of fish pending in Nordskaget waiting cage after the 24 hours calm period had been completed. The figure shows how the waiting cage was unnecessary before the emergency slaughter, but after the ban needed to borrow capacity at uneven intervals. This is supported by Figure 7.10, which shows the amount of cargo in the waiting cage by Jøstenøya after the imposed emergency slaughter. The figure shows a controlled level, which steadily increased during the summer months, but was altered at 4380 hours. The high peaks were the periods when the waiting cage was full, and incoming cargo had to be redirected to Nordskaget. Upon introducing processing vessels in the 6th simulation, the waiting cages were bypassed, and the need for borrowed capacity was eliminated.

The use of the processing vessels ensures that Jøstenøya manages to keep production flow at bay, even at large volumes and emergency slaughter scenarios. Figure 7.11, shows the waiting cage volumes at Jøstenøya for the last simulation, which includes processing vessels. Compared to Figure 7.10, the figure illustrate the unloading effect the implementation of processing had on a system in strain, as the emergency slaughter was hardly visible in the plot. Output included standard de-

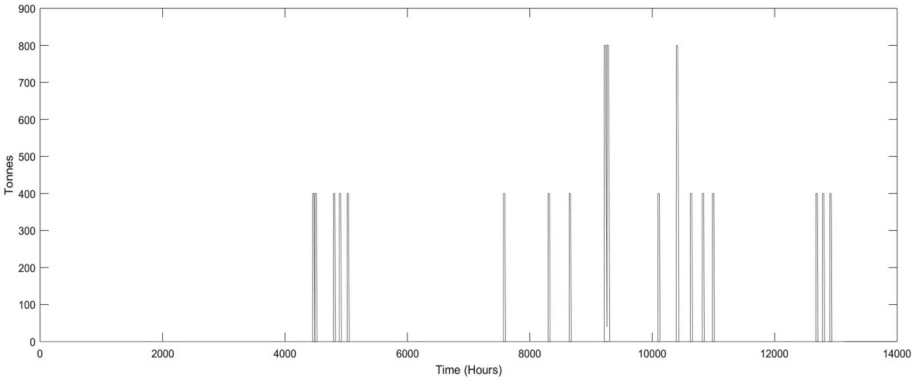


Figure 7.9: Tonnes of fish pending in Nordskaget waiting cages, after the completed service time of 24 hours, for simulation number five. The waiting cage is only used after the emergency slaughter at 4360 hours is implemented in the simulation.

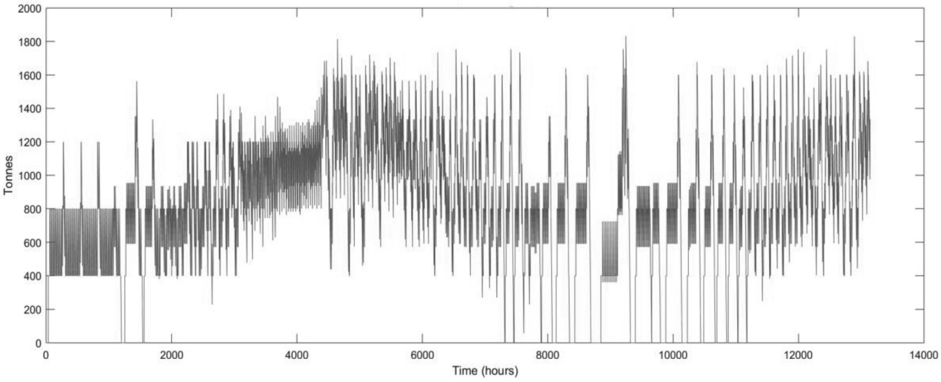


Figure 7.10: Tonnes of fish pending in Jøstenøya waiting cages for simulation number five, where high accumulation of fish is induced by emergency slaughter and an inefficient vessel fleet composition.

viation, and corresponding plots for the other simulations can be found in Figure 9.33 in Appendix D.5.

Table 7.8 shows that the utility of the land-based facilities increased in the last three simulations during four-fold production level, indicating less factory downtime, but also less flexibility to withstand a sudden shock, as can be seen from simulation five, where the system needed to borrow capacity from Nordskaget during emergency slaughter. The processing vessels used in simulation 6 clearly relieved the slaughtering part of the factory, and would thus only require develop-

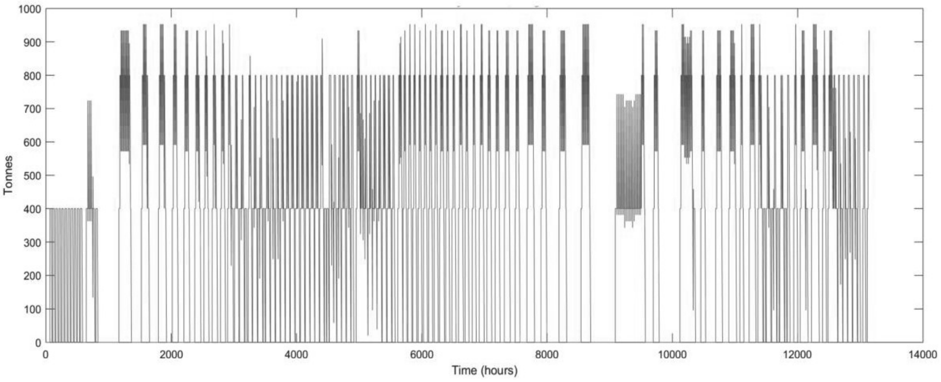


Figure 7.11: Tonnes of fish pending in Jøstenøya waiting cages for simulation number six, where processing vessels were introduced to the system.

ment of the second half of the factory to adapt to the new production levels.

Regardless of production volumes or slaughtering capacity, the time from reaching land to the various measuring points were similar for all six simulations, the last three with slight headway. Processing vessels, in both cases, generated a slight advantage, however, not as much as the 24 hours they save from waiting cages. This was probably a result of being the second priority on advancement to processing on land, relative the fish which was slaughtered on land, and already on the conveyor belt.

7.2.2 Waiting Cage Prohibition

With the recent tone shift in public fora, animal welfare and ethics have been scrutinized in general, including waiting cages. In light of this, it is valuable to asses what effect a ban on traditional waiting cages would have on the supply chain. The ban is introduced to the simulation according to the problem boundaries, by holding the live fish carriers for an extended 24 hours before unloading directly to slaughtering. The simulated ban thus only influenced the live fish carriers, as processing vessels bypassed the waiting cages altogether. Of particular interest was where an added capacity would be required in the supply chain in order to compensate for the vessels being held up. In order to evaluate this, a set of simulations were conducted with input according to Figure 9.34 in Appendix D.6. Table 7.9 shows a selection of the most prominent of these.

Table 7.9: The most prominent input for simulations testing waiting cage ban impact.

Run number	**1**	**2**	**3**	**4**	**5**	**6**	**7**
LFC Capacity	400	400	400	400	400	400	900
Number of LFCs	4	4	2	4	2	4	6
Number of PVs	0	0	2	0	2	0	0
Waiting Cage Ban	No	Yes	Yes	Yes	Yes	Yes	Yes
Number of Berths	1	1	1	10	10	3	1

All simulations included cooperation, had five times today's production levels, and all processing vessels used had 500 tonnes capacity. The first three simulations show how a ban affected a simple chain, with high production volumes, and limited capacity for receiving load from fish carriers. The last four evaluated how berth capacity, vessel capacity and use of processing vessels influenced each other.

A selected set of outputs from the seven simulations are presented in Table 7.10 and 7.11. They contain mean cage volumes, information regarding vessel waiting actions, and production flow. The total output spreadsheet, and selected plots can be found in Figure 9.35 and 9.36 in Appendix D.6.

The first simulation shows how a system works without a waiting cage ban. The same system was then imposed by a ban in simulation number 2. This showed a huge effect of the flow, and the four vessels were no longer able to collect all the fish in the cages, where the stock grows uncontrollably. This was more prominent for the exposed cage than the traditional ones, as the model chose to send vessels to the traditional cages if they were able to fill a full vessel, see Section 6.1.4. The high average in the exposed farm indicate that that is often the case.

The third simulation replaced two of the live fish carriers with processing vessels. This helped reduce the mean volume waiting to be collected from exposed farms to a fourth. Even with a halved fleet, the number of round-trips performed by the

Table 7.10: Selected outputs for waiting cage prohibition simulations, vol 1.

Simulation Run Number	**Unit**	**1**	**2**	**3**
Traditional Farm				
Mean Volume	[tonnes]	200	906	838
Exposed Farm				
Mean Volume	[tonnes]	419	154 170	37 987
LFC				
Round-trips	[weekly]	14.6	7	4.6
Mean waiting to unload	[vessels]	0	2.9	0.5
Mean time waiting to unload	[hours]	0.34	62	8.3
Processing Vessel				
Waiting to unload	[vessels]	-	-	0
Wait time to unload	[hours]	-	-	0
Slaughter Facility				
Delivered to Jøstenøya	[tonnes]	450 000	218 000	386 200
Delivered to Nordskaget	[tonnes]	0	0	0

live fish carriers was only reduced by 35%. The time spent waiting for an available berth was also reduced from 62 hours to 8.3, which is a considerable contribution to the vessel up-time. Without the forced delay at berthing, the processing vessels proceeded through the system unobstructed, and their wait to unload was accordingly zero hours.

Evident from the number of vessels waiting to unload in simulation number 2 is that berthing was a narrow bottleneck, and when this was occupied by a lingering vessel, the wait propagated to all the other vessels waiting to unload. The problem was almost eliminated by introducing processing vessels, upon which the live fish carrier cycle was much better coordinated.

The amount of fish delivered to Jøstenøya in Table 7.10 and 7.11 indicate that neither the second nor third simulations were able to delivered all the required fish within a year, although processing vessels reduced this production hold-up by 168 200 tonnes. The first system, however, was able to transport almost the complete 451 800 tonnes, the remaining 1 800 probably being in transit when the simulation stopped.

Table 7.11: Selected outputs for waiting cage prohibition simulations, vol 2.

Simulation Run Number	**Unit**	**4**	**5**	**6**	**7**
Traditional Farm					
Mean Volume	[tonnes]	202	576	202	449
Exposed Farm					
Mean Volume	[tonnes]	205	474	204	450
LFC					
Roudtrips	[weekly]	14.6	6.5	14.6	6.5
Mean waiting to unload	[vessels]	0	0	0	0.3
Mean time waiting to unload	[hours]	0	0	0	7
Processing Vessel					
Waiting to unload	[vessels]	-	0	-	-
Wait time to unload	[hours]	-	0	-	-
Slaughter Facility					
Max unloading simultaneously	[vessels]	4	4	3	1
Mean unloading simultaneously	[vessels]	2.1	1.1	2.1	0.9
Delivered to Jøstenøya	[1000 tonnes]	450	449	450	422

The results in Table 7.11, shows that system throughput was significantly improved by expanding the berthing capacity, indicating that the vessel hold-up time itself was not the primary source of the problem. More than half of the fleet was stand-by waiting for the fish to settle and unload at all times. Upon introducing processing vessels, the mean volumes in the farms increased slightly. This was probably due to some delay in distributing between the two circuits, which is an artificial problem. The live fish carrier round-trips were more than halved with 2 vessels relative to 4. This was because the cargo was distributed according to the capacity distribution between the two fleets, which was 800 versus 1000 for simulation number 3 and 5. The simultaneous unloading was reduced by almost 50% when processing vessels were included. This corresponds to the short amount of time these vessels spent unloading relative the live fish carriers.

Returning to the second fleet, three berths gave the results of run 6. These indicated similar conditions to the first simulation, with marginally longer unloading waiting times for the vessels. Again were more than half the fleet on average loaded and standing by in berth. This supports the earlier discovery that one berth does not suffice during a restriction on waiting cages.

Simulation number 7 replaced additional berthing capacity with vessel capacity, which worked satisfactorily. Both farm volumes waiting for vessel pick-up averaged on half of the vessel capacity, and the waiting time was approximately 7 hours. 0.9 vessels on average unloaded at the same time, indicating a high utility, and the berthing capacity was well taken advantage of. The total delivered load to Jøstenøya was, however, lower than the previous simulations due to the large amount of cargo held up in the vessel fleet, and the larger volume of fish accumulating in the farms. Thus, increasing vessel capacity might reduce vessel queue to load, compared to simulations with reduced berth capacity, but simultaneously holds up large amounts of cargo.

7.3 Benchmark Fleet

In order to test the scenario influences at five-fold production, a basis fleet had to be decided. This was done in order to test the influences on a realistic fleet, rather than the general effect. The cargo fleet was in Chapter 7.1.3 found to suffice with one 2000 dwt cargo vessel. The trucks are maintained at a high level, ensuring no shortage, as this is considered to be of lesser importance. The final fleet composition to determine is thus that of live fish carriers and processing vessels.

In order to find the best fit for normal operation the model was run iteratively, analyzing the influence of various fleet compositions. Initially were exclusively traditional fish carriers employed. Four vessels were first included in the model, analyzing the queue to load in order to find how many were in fact needed. Figure 7.12 represents the number of live fish carriers waiting to pick up fish from traditional farms (left) and exposed farms (right). These indicate that the traditional farms were provided with sufficient capacity, as there is rarely no vessels available. The exposed cages had lower availability because the production was higher, so the available vessels were loaded and shipped off immediately. The rush to load could imply that the fish were accumulating in the cages. The fish queue-to-load never surpassed one batched entity, however, which implies sufficient availability of vessels as seen by the lack of fish queuing up in Figure 9.39 in Appendix D.7.

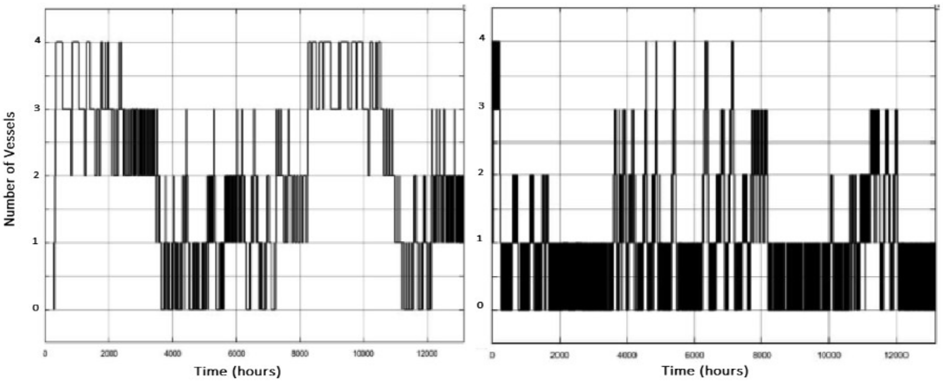


Figure 7.12: Number of live fish carriers waiting to load from traditional (left) and exposed (right) cages during normal operation.

Having established that four vessels would be sufficient to serve the production, a set of simulations were run in order to decide on a benchmark fleet. Serving as KPI for the various fleets was the time measured from servers after separating processing vessel and live fish carrier cargoes, and to completed processing. This is a time-gap which roughly represents the time from the fish was picked up from the cage, until complete processing, but involved some simulation-technical blocks, which could induce artificial queues. It was thus important to ensure that these were at a minimum, which was confirmed by queue scopes in Figure 9.39 in Appendix D.7. Table 7.12 displays KPI figures for various fleet compositions. For a fleet consisting of four live fish carriers, was the average time from cage to processing 45 hours.

Next, the other extrema was tested; no traditional fish carriers, but four processing vessels. These had a preliminary capacity of 500 tonnes, against the 400 tonne fish carrier, due to the reduced amount of water included during transit. This resulted in longer waiting times for the fish during winter months, when the production was lower, subsequently making it harder to fill a larger vessel, see Figure 9.40 in Appendix D.7. During summer, the time until completed processing was lower

Table 7.12: Time used between the fish is ready for pick-up, and to processing is completed for various fleet compositions.

		Time KPI			
LFC	**PV**	**Mean**	**Max**	**Min**	**IQR**
4	0	45	128	39	4
3	1	45	169	18	5
2	2	41	230	16	25
1	3	30	152	16	17
0	4	23	121	16	5

due to the large capacity of the vessels and the sidestepped waiting cags. Running only processing vessels and only live fish carrier key time components can be seen in the first and last row in Table 7.12.

Figure 9.40 indicates that the system was vulnerable during the slow winter months, when long waiting times influenced the system. Using processing vessels was efficient during summer, but the larger capacity causes problems during winter. A heterogeneous fleet was tested in order to utilize the best of both worlds.

Varying the fleet to comprise a total of four vessels from farm to shore, based on results from Chapter 7.1.1, the number of live fish carriers and processing vessels were varied. KPIs from these runs resulted in the rest of the figures in Table 7.12. These indicated that the mean time spent from fish cage to processing decreased with added processing vessels. Because these are expected to be more expensive with added equipment compared to the live fish carriers, the investment should result in pay-out in the shape of increased system performance. In the table, the biggest leap in time reduction is from simulation three to four. This implies going from two of each vessel to three processing vessels and one live fish carrier. Bjørnar Johanessen from Blått Kompetansesenter also commented in conversation that it is anticipated that the land-based facilities might be altogether bypassed in the future, with full factory vessels taking its place. In light of this, it was decided to move forward with a fleet comprising one fish carrier and three processing vessels. The complete basis fleet input is thus as seen in Table 7.13.

Table 7.13: Final composition of benchmark fleet.

	Number [-]	**Capacity** [tonnes]	**Speed** [knots and kph]	**Down-time** [h/year]
Processing vessels	3	500	12	140
Live fish carriers	1	400	12	160
Slaughtering facilities	-	94	-	-
Trucks	100	20	60	-
Cargo vessels	1	2000	16	100

7.3.1 Scenario Testing

When the fleet composition had been decided, it was tested with imposed shock scenarios. The fleet composition was thus as seen in Table 7.13, and the following sections describe the associated simulation results.

Emergency Slaughter

It was a challenge to find an approximate volume for emergency slaughter, because the regulations and challenges of future exposed aquaculture are still unknown. Continuing the five-fold increase, a one-cage slaughter would correspond to approximately 3 000 tonnes from an exposed farm. This was imposed at 6 000 hours.

Figure 7.13 shows the queue of fish waiting to load processing vessels from exposed farms, after the imposed emergency slaughter, with an unit of 500 tonnes of fish. The system worked rapid to clear the added cargo, moving from no queue, to 3000 added tonnes, and taking approximately 300 hours to restore back to system normality.

The observation was supported by the vessels in queue to load from the traditional farms to the same fleet. These usually frequented with 80-90 hours intervals, but when the added fish is imposed, all vessels sail to the exposed farms to clear help, see Figure 7.14. The same tendency was seen in the traditional farms, as shown in Figure 9.41 in Appendix D.7, where accumulation of fish started after the vessels prioritized the exposed farm. This left a 350 hour gap with the traditional farms,

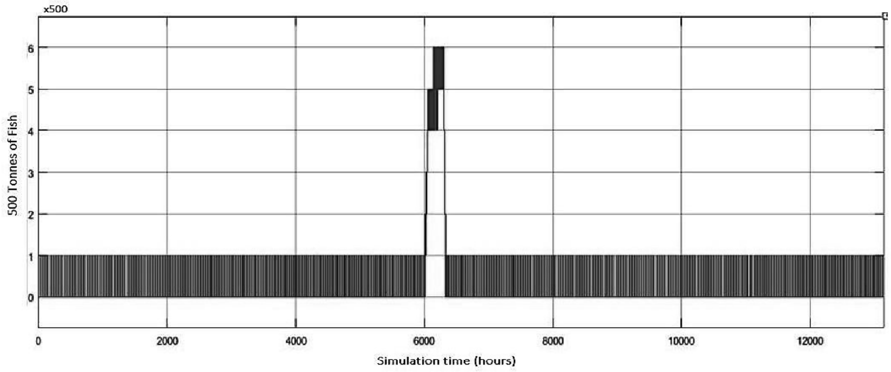


Figure 7.13: Tonnes of fish waiting for processing vessel loading, during emergency slaughter scenario.

which were then visited frequently by vessels to pick up the slack, before the system was restored to normalcy.

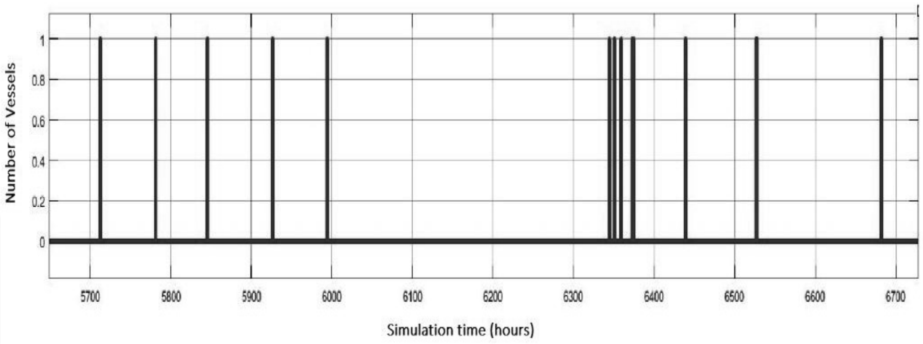


Figure 7.14: Available vessels at traditional farms during emergency slaughter.

The added volumes fish then moved down the supply chain, first meeting the slaughtering facilities. These were relieved by the processing vessels, and had no problem handling the sudden increased pressure. The processing facilities had, however, to incorporate the cargo from the processing vessels in addition to the flowof fish coming from the slaughter facility, which gave a increased utility, see Figure 9.42 in Appendix D.7.

This entailed that the now-processed extra cargo continued down the chain, and was loaded onto transportation. The cargo vessel was unable take on any extra load, and the trucks needed thus take on the full supplement. This can be seen from the departed entities to vessel queue in Figure 9.43 in Appendix D.7, which have a continuous departure-rate. The Europe-bound trucks, however, maintain the added strain from the processing facilities, which can be seen in Figure 7.15. This is replicated in the other trucking routes, and requires a larger fleet than usual, according to Figure 7.16.

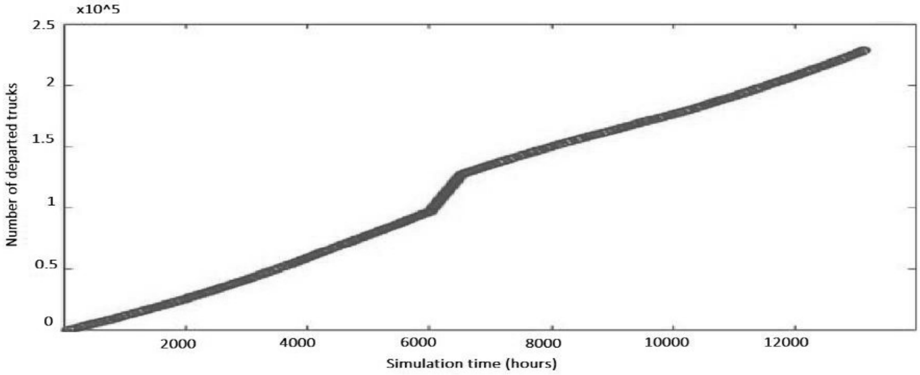


Figure 7.15: Number of departed trucks from Jøstenøya during emergency slaughter scenario for benchmark fleet [x10^5].

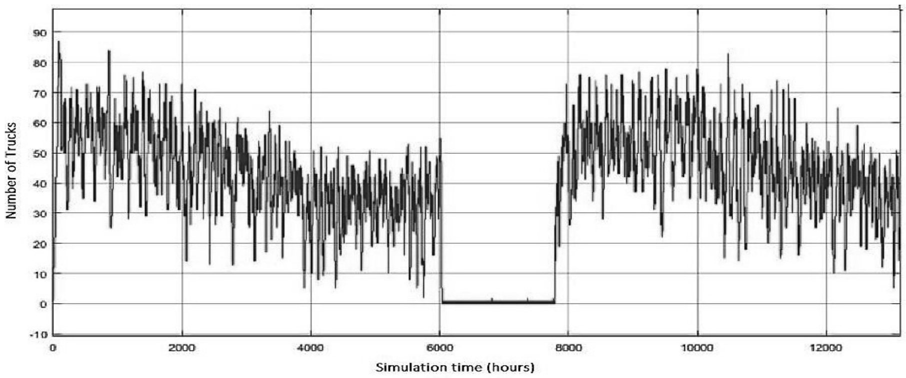


Figure 7.16: Number of trucks waiting to load at Jøstenøya during emergency slaughter scenario.

Waiting Cage Prohibition

As the results from Section 7.2.2 showed, was the number of berths essential during waiting cage prohibition simulations. The required number depends on the total number of fish carriers in the fleet, and the number of processing vessels compared to live fish carriers. Thus, as the number of processing vessels for the benchmark fleet was three of four vessels in total, the number of berths was set to two for this simulation. The remaining input, and full spreadsheet of associating simulation output can be seen i Figure 9.34 and 9.36 in Appendix D.6.

The sample fleet was exposed to a ban on waiting cages, which again implied that the live fish carrier was upheld for 24 hours before unloading. Table 7.14 shows the most prominent results from the simulation.

Table 7.14: Selected outputs for sample fleet during waiting cage prohibition.

Simulation Run Number	**Unit**	
Traditional Farm		
Mean Volume	[tonnes]	780
Exposed Farm		
Mean Volume	[tonnes]	460
LFC		
Round-trips	[weekly]	3.1
Mean waiting to unload	[vessels]	0
Mean time waiting to unload	[hours]	0
Processing Vessel		
Round-trips	[weekly]	8.4
Waiting to unload	[vessels]	0
Wait time to unload	[hours]	0
Slaughter Facility		
Max unloading simultaneously	[vessels]	2
Mean unloading simultaneously	[vessels]	0.7
Delivered to Jøstenøya	[1000 tonnes]	449

The mean time spent in both traditional and exposed farms was somewhat higher than its counterparts in Table 7.10 and 7.11, but they were steady, implying that the fleet was able to clear the required volumes, which is supported by Figure 9.44 and 9.45 in Appendix D.7. There was no accumulation of fish, which indicates a sufficiently large fleet. This, in turn, requires checking that the fleet is not excessive, which was controlled by evaluating the vessels in queue to load scopes.

These show no accumulation of fish wither, and the longest wait for a live fish carrier was 30 hours, see queue plots in Figure 9.46 - 9.45, Appendix D.7.

The live fish carrier made more than three trips weekly, which is more or less continuous operation. This is also supported by the short waits in Figure 9.48. The processing vessels also have low downtime, and more than 8 round-trips per week.

The fleet occasionally utilized both berths simultaneously, supporting the need for more than one, although the average was less than one vessel at a time. The total volumes of 450 000 tonnes over the course of 1.5 years was also delivered to Jøstenøya. All these factors indicate an efficient system, effectively bypassing the waiting cage ban with a relatively large fleet of processing vessels.

7.3.2 Worst Case Scenario

In the interest of covering all aspects, a simulation was also run with both waiting cage ban and emergency slaughter of 3 000 tonnes, representing the worst case scenario. The processing vessels represented the largest part of the fleet, and were largely unaffected by the waiting cage ban. The results are thus similar to those of subsection 7.3.1. The live fish carrier fleet was hit harder by the double influence, and as seen from the amount of cargo waiting to be picked up from exposed farms in Figure 7.18, the system regained stability within approximately 600 hours, roughly 25 days.

Figure 7.18 shows the time used between batches of fish reaching shore, and delivery to market for two simulations: Worst case (circles) and normal operation (triangles). Keeping in mind that the plot disregards all time spent in vessel transit, the figure shows that the delay did not propagate after settling. There was one abrupt phase immediately after the emergency slaughter in the worst case run, but after this, the two simulations were interchangeable regarding time KPIs.

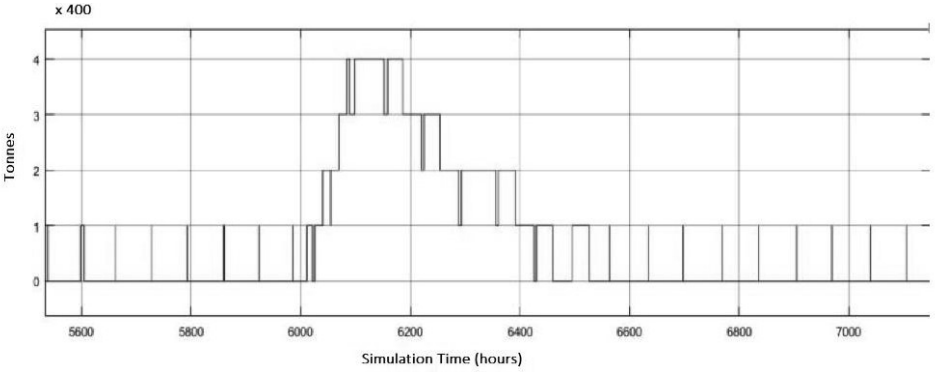


Figure 7.17: Batches of fish waiting to be picked up by the live fish carrier at the exposed farm during a worst case simulation run.

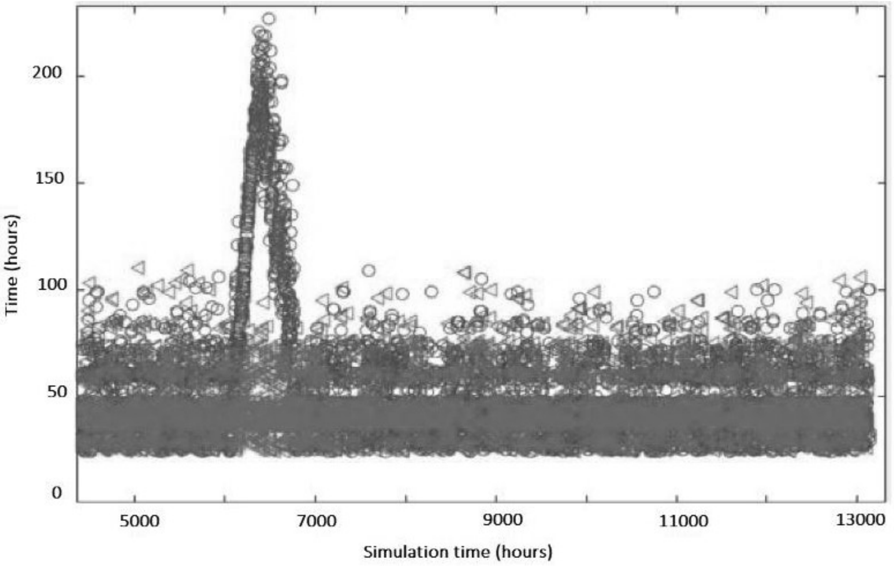


Figure 7.18: Time spent for batches of fish between reaching shore, and being delivered to market, for worst case simulation (circles) and normal operation (triangles).

Key Results

This chapter has shown the results from the simulation runs. Focus was gradually altered throughout the chapter, starting with validation of input and model, and progressing to introducing more elements. This ensures a correct model, and provides an understanding of the influence each element has on the results.

The most significant results included the need for larger production in order to justify introducing cargo vessels, and the considerable effect collaboration had. The knowledge from these results formed basis for moving on to finding a benchmarking fleet, and introducing crash scenarios, where processing vessels were a contribution to continuity in operation.

CHAPTER 8

Discussion

The simulation execution with associated results will be discussed with respect to the problem description and scope in the following sections. Because there are no previous studies using discrete-event simulation to model the future of aquaculture, emphasis has been put on the validity of the model and methodology. Thus, has a considerable part of the discussion been dedicated to evaluating this, as well as the subsequent credibility of the results.

The supply chain will then be broken down, and the various components and their role in the complete system evaluated. Finally, will the thesis objective be discussed, where throughput and system composition will be used to evaluate cohesion, flexibility and vulnerability.

8.1 Testing and Validation

Simulation error can come from several sources, e.g. incorrect modeling assumptions, inaccuracy in apriori information, or just imperfect coding. Murray-Smith writes in his book ”Testing and Validation of Computer Simulation Models” that the modeler essentially has to keep on conducting tests and evaluations until he/she is sufficiently confident that the model is acceptable for the application being considered. This can be a challenging process, as quality assessment of more complex models is difficult (Murray-Smith, 2015). Based on the relationship between reference data and simulation output, it had to be determined to whether a model, and the associated data set, were a reflection of the real world.

Validating a model can be done in several manners. The most easily accessible was through checking scope outputs, as these display an intuitive and time-continuous representation of the entity movement through a block. This included making sure no entities were delayed in incorrect blocks, and that departures were present where they should be.

Chapter 7.1.1 presented results retrieved from a normal operation, where the existing supply chain was used for preliminary model validation. After constantly monitoring model behavior, a number of observations supported the assertion of a sufficient model; all fish generated in the farms were delivered to market, transport actors were capable of handling the incoming batches of fish, and vessels and vehicles preform their round-trips in predictable patterns. The total time used from arrival at shore, until reaching its consumer was 2-4 days on average, depending on destination, proving that the model reads the time attribute values correctly. The market distribution of products presented in Chapter 5.2.4 was very well replicated during simulation.

It is, however, important to emphasize that although several validations have been performed, there still may be basic mistakes in the code, causing it to deviate from the system it represents. Some of the validations were performed based on common sense and the authors' understanding of what the output should look like. A number of assumptions and simplifications were made during the development process, which influenced the simulation output, and should thus be considered when evaluating the project results. One of the most prominent being that the model does not incorporate continuous adjustments in the event of production changes. Where the real world supply chain management would adjust factory capacity by e.g. increasing the personnel at the and utilizing all available resources, according to continuous operation, the model needs to have predetermined priorities. Due to this were the utility measures somewhat off, and were disregarded as a performance measure.

Additionally is a simulation only as strong as its input. All data used to construct the supply chain, and used as simulation input, had to be evaluated in order to substantiate their credibility, subsequently the robustness of the results. This means that all sources used were checked for relevance or other aspects which could influence the content. Furthermore, was the source of information considered in terms of when it was published, why was it published, and where was it published. All articles- and reports used in this thesis were accessed through NTNU's database Oria or other relevant web-pages, and information from experts and aquaculture stakeholders was received first-hand. The information based on conversations with industry actors are naturally subject to exaggerations or personal perception.

Discrete-Event Simulation

Using simulation as tool in order to analyze system interactions proved to be fitting for the scope of the thesis, and the interactive interface of Simulink allowed for fairly intuitive model structuring. Allowing the developer to mold the system block by block, using MATLAB to incorporate unit behaviours, made it possible to customize the complex supply chain according to project scope and limitations.

Although all natural flows are continuous, the model was built in a discrete space, implying a sectioning of the time periods. This is a simplification of the real world, but with small enough intercepts, the estimations become reliable. If the time unit was set to months instead of hours, only running for 12-24 units, the entities would be artificially batched together, and the results unreliable. If it was set to seconds, it would mimic continuous time very closely, but the amount of calculations would make the model unnecessarily slow to run.

However, the challenge of using Simulink lies in the same feature, with endless possible models. By developing a model from scratch, there were many pitfalls. These became particularly evident during result generation, which prompted several alterations, in turn making earlier results incorrect. The process thus became a continuous iteration, where one final model had to incorporate all system variations.

Many of these challenges were bypassed by adding blocks, data stores or other work-arounds, while the ones which could not be sidestepped within the scope of this project were accounted for in simulation limitations. These were deviation from normal operation, such as cases of slaughter facility downtime, injury, technical unplanned failure, etc. These are incidents which will be a part of any production process, and will not influence one system composition more than the other. The comparison between these was thus unaffected, but the total time to market will need to incorporate a safety margin. One of the most influential limitations was the self-regulating nature of the model, which could not be affected during simulation. This removed the possibility to manually intervene during untypical production, and particularly affected the distribution of cargo between live fish carrier and processing vessel fleets.

8.2 Fleet Composition

Live Fish Carrier

The live fish carriers have generally not been evaluated for their effectiveness in this chapter, as they are currently in real-life operation, and are known to function. The model replicated their operational pattern sufficiently, and were thus concluded to be a good representation of the real-life system. The shipbuilding trend indicates that all vessels grow in size, and live fish carriers have not been an exception. It is thus possible that these have been under-dimensioned in the futuristic simulations in this thesis.

All vessels had incorporated a downtime value in the input file, in order to include service time and maintenance. But, as it affected all vessels equally, and the actions would typically be scheduled in calmer periods, the annual downtime was disregarded for comparative purposes. It slightly influenced the total time to market for some batches of fish, but not enough to influence the over-all results. The maintenance time can, however, be used if correct figures are implemented in the input file, and scheduled according to season.

Processing Vessels

A considerable part of this thesis was devoted to assess the value of processing vessels. Many farmers are signaling that slaughter during transit is an interesting possibility, and the goal was thus to understand the perks and repercussions of including this feature in a fleet. A processing vessel will have the benefit that it can bypass the waiting cages and part of the land-based process. This was done by moving segments of the process on board. Killing the fish immediately has advantages related to fish welfare regulations, which can be disregarded when there is no transportation of live biomass. This includes, among others, that the amount of water which has to be brought along for the transit can be drastically reduced. There is no advantage in transporting large amounts of water, on the contrary, it adds to the dead-weight of the vessel, but not the payload. Utilizing a processing vessel also reduces the amount of times one has to handle the fish whilst alive, reducing risk, stress and loss from loading. During normal operation will the employment of processing vessels cut the total time from cage to market. This time is, however, time saved before the fish has been slaughtered, and does, as such, not influence the total time to market. The value of introducing processing vessels is thus rather in quality than effectiveness.

A processing vessel is, however, a significant upgrade relative to a traditional live fish carrier, introducing a series of new technology, and in practice incorporating a small factory onto the vessel. This implies moving a large, centralized function to more wide-spread facilities, which in classic supply chain theory is a disadvantageous development. The equipment is also expensive and heavy, adding to the outfitting and fuel costs.

From a supply chain perspective is the value of a processing vessel the reduction of chain length, and the independence from waiting cages. These were thus the perspectives which were included in this study, where all commercial aspects were disregarded. By implementing processing vessels during a ban on waiting cages, the value became evident, allowing the vessels to serve as a reliable source in a supply chain in crisis. The added capacity which the processing vessels were assumed to bring was favorable during high production months, requiring fewer trips to transport the same amount of cargo. During slower winter months, however, the added capacity became a burden, forcing the earliest arriving cargo to wait for longer periods for the vessel to fill up and depart.

Although this study only evaluated a very narrow part of processing vessels' contribution to an aquaculture supply chain, it proved to be a trustworthy addition, providing valuable flexibility in extraordinary situations.

Slaughter and Processing Facilities

Chapter 2.4 stated that one way to ensure system reliability would be to either ensure constant high capacity, or introduce redundancy as a parallel standby system. In this supply chain, there was capacity flexibility for all of the transportation legs by chartering additional units, whereas the land-based facilities have a rigid potential. The former project thesis focused on how the factory served as a focal point for the supply chain, and limited the total throughput. The conclusion was that measures like reducing staff, increasing service time, or increasing production volumes could ensure a higher utility. But since a 100 % simulation utility was virtually only possible when there was a constant queue ahead, it significantly increased the number of fish in the waiting cages, and it was concluded that was was more important to ensure a steady stream to and from the factory.

It was thus made sure that the factory capacity in this thesis was held high throughout most simulation runs, in order to spot additional system bottlenecks. Because all management adjustment possibilities regarding factory, e.g. shift work, was disregarded, would the utility measure be misleading regardless. This assumption also caused the level of cooperation between Jøstenøya and Nordskaget to be somewhat off. As a possible cooperation was based on the level of fish waiting for slaughter, and the capacity of the factory was somewhat oversized, could the need for help sometimes be greater than the results show.

Other bottlenecks which became evident during the simulations, were the the loading- and unloading capacities at the factory. To ensure a steady flow of entities during times of high production volumes, one would have to make infrastructural upgrades to every step of the supply chain. This included providing terminals and ports with higher capacities. The number of required berthing spots was found to be closely linked to the number of live fish carriers in the fleet. This was particularly important during an imposed waiting cage ban. The trucks also need to have sufficient loading capacity available at the factory, and were found to be approximately minimum 3 spots during five-fold production. These numbers have to be seen in relation to the project limitations and system simplifications, and should be treated as illustrating numbers. However, up-scaling the aquaculture industry by five-fold would lead to necessary upgrading of associated infrastructure in addition to altering the transport modes' capacity and quantity.

Transportation by Road

The utilization of trucks was not considered an important factor for this analysis. These were assumed chartered on a when-needed-basis, and lingering vehicles were thus not an issue. However, longer contracts are cheaper, and a certain level of trucks needs to be kept in action at all times. Subsequently, for projects including monetary considerations, should discussions regarding whether spot-chartering of trucks was a sufficient assumption be involved.

The trucks were effective, and were run without interruptions and delays. This was a simplification from the real world, as the interview with Lerøy Midt CEO, Sven Amund Fjeldvær revealed that fragile roads, traffic jams and mechanical unreliability occasionally lead to transport delays (pers.comm., 6 October, 2016) .

Production numbers for today's level are seen in 5.2.1, causing approximately 3000 truckloads to leave Jøstenøya every year, or 58 every week. As this simulation was supposed to represent one out of three factories in the region, the total flow of loaded trucks driving from Hitra was approximately 173 every week. These numbers need to be treated as illustrations, but were approximately in line with numbers shown in Chapter 2.3.2. A five-fold production increase can thus cause problems for the Norwegian road network. Section 7.1.3 states that approximately 2158 of the 3000 annual truckloads were sent from Jøstenøya bound for East- and West Europe. This number was reduced by as much as 80 percent when a cargo vessel was included in the supply chain.

Although the need for trucks was reduced upon implementation of cargo vessels were these an important feature during abnormal situation, and has to be maintained as a backup system. Without the parallel truck node, the cargo vessel is much more exposed to alterations, and has to be able to handle all situations independently. With trucks as a secondary means of transport, however, are the vessels allowed to focus on carrying the bulk of the volume, but deviations are picked up elsewhere.

Cargo Vessel

Through the analyses, there was no doubt that from a practical supply chain point of view, the trucks were close to unbeatable in efficiency. In light of the current focus on sustainability, and considering the fragile infrastructure, however, will an increase in truck traffic be unwise. Consumers are more conscious than ever, and green operation has become an important trademark, particularly for biological products.

Cargo vessels were introduced as part of the simulated supply chain, despite the farmers' reluctance to introduce these. Their main concern is regarding risk and delay. Delays were included here only due to weather, and predetermined maintenance downtime. Chapter 7.1.4 presents that the average time spent sailing throughout the year is stable. Only 3% of the time during the winter months was the traffic stopped due to large specific wave heights. These numbers need to be evaluated in relation to the limitations presented in Chapter 3.2. In normal operation will there also be a series of other aspects which could cause a cargo vessel to miss schedule. These range from bureaucracy to technical failures, and could cause delays from hours to weeks. The extent of these influences depend largely on the shipowner, and effort should be put into finding a reliable transporter.

The results show that implementation of cargo vessels with today's production volumes would require using smaller vessels in order to avoid excessive waiting times in storage. Production has to approximately double before cargo vessels add value to the supply chain, and even then it was hard to challenge the traditional trucks. This assumes decent size vessels, as smaller ones could practically be possible already, but would not be able to serve as economies of scale, and thus be too expensive.

By cooperating on transportation to market, it was proved that the waiting time was reduced. At Hitra/Frøya, this will typically be between Marine Harvest AS, Lerøy Midt AS and SalMar AS. These are competitors in operation, but by cooperating on transport, all three benefit from a possibility which was unavailable to any one of them alone. As the results in Section 7.1.3 show, would cooperation today make it possible to introduce cargo vessels, whereas the future could allow the farmers to independently incorporate cargo vessels into their supply chain.

At a five-fold increase in production were cargo vessels almost inevitable, and competitive with trucks with regard to time, see Section 7.1.3. However, the flexibility was significantly reduced due to the large volume per unit. Production stops or delays will propagate, and the vessel will either have to sail with reduced cargo, or wait for normalcy to be restored, possibly causing tardy deliveries. These are unpopular alternatives to a Norwegian farmer who trademarks himself as efficient and reliable.

Including cargo vessels means adding an extra link, and mode, to the supply chain. This requires more planning between the preceding and proceeding units in order for the chain to flow smoothly. A cargo vessel would be the chain's biggest entity, and thus requires large amounts of smaller units to supersede it, which should be coordinated as closely in time as possible. Upon unloading, this entails 100 trucks on stand-by, immediately ready to take their share to their final destination. This added link is also an added source of risk and complication, making the chain more vulnerable.

Another aspect which has caused skepticism towards the introduction of cargo vessels is the risk of delay. This system risk was mitigated by maintaining a fleet of trucks operating in parallel. When there was no available cargo vessel in the near future, all cargo was sent by truck, ensuring continuous flow and predictability to the customers. The value of this substitute arrangement became particularly evident during periods with emergency slaughter. Chapter 7.2.1 showed that all the additional cargo was sent by truck, and it is thus important to maintain this redundancy measure. Delay which occurs after the vessel is fully laden is unaffected by this measure, and remains a concern for the industry which has to be handled otherwise.

The use of composite entity creators in order to model merging of transporters and cargo, has had a few drawbacks. The most prominent being that although the two entities merged do not need to have the same structure, they needed to be of a consistent 1-to-1 size thought the simulation. This means that one was unable to vary the degree to which the transporters were loaded between round-trips, and made it necessary to assume 100% full load every run. In turn, this required the cargo vessel to wait for a full shipload every time, which affected the mean time spent delivering the products to the market. A cargo vessel can theoretically carry all types of cargo, as long as they do not have a negative impact on one another. This implies that a vessel would not necessarily be filled fish fish products, but could include other export goods. As with the return ballasting leg was this consideration outside the scope of the project, but could have an impact on the profitability and feasibility of cargo vessels, as well as allow for earlier introduction.

8.3 System Interaction

The results of the simulation runs were then considered with regard to the scope of the thesis, assessing system flexibility and vulnerability based on throughput and composition. This incorporated results from all analyses, and evaluated the system from a meta-perspective.

Cohesion and Flexibility

The overall analyses gave the impression that system cohesion is paramount. Small alterations had large repercussions down the supply chain if they created bottlenecks or insufficient dimensions. The opposite case, excessive dimensions, allowed fluctuations to propagate. This was particularly evident with respect to emergency slaughter, when the wave of additional cargo was maintained all the way to transport to market.

Evident from both emergency slaughter and waiting cage ban case was that parallel modes provide greater flexibility and robustness throughout the system. When emergency slaughter had to be conducted, it was necessary to utilize additional trucks for transportation to Europe, as the cargo vessel was already running on full capacity. Without this redundancy, the sudden extra strain on the system would have propagated in time, affecting all elements of the system for longer.

In the case of a sudden waiting cage ban was the redundant berthing capacity vital in allowing the supply chain to uphold efficiency. This also showed that the flexibility in employing processing vessels, which were unaffected by the ban, maintained stability during an unprecedented period. The influence of this measure was probably understated, as the model had some trouble distributing correctly when one fleet was significantly quicker than the other, which most likely influenced the results. The cargo should have been sent to the processing vessel in greater scale when they were faster, instead of being distributed based on total capacity share.

Although processing vessels can eliminate the need to develop the slaughtering capacity, the processing capacity still has to be in line with the level of production. This becomes even more important with incorporation of processing vessels, as these will bring already slaughtered fish with a hurry to be processed. In this regard, the flexibility of waiting cages is eliminated, and fish from two channels will meet at a critical phase. This needs to be handled through planning as the arrival of slaughtered fish is predictable and controllable.

In Chapter 7.2.2 it was also found that berthing capacity had the ability to act as a bottleneck when production was increased. The same was found for loading docks for trucks. These are cheap measures to ensure sufficient availability of, and should not be allowed to be the restricting element in the supply chain. Upon developing the chain, focus should always be on increasing, or bypassing, the weakest link, which is the only investment which will have an effect.

Vulnerability

The real world supply chain is vulnerable to a number of hazards such as technical failures, human errors, environmental impacts, variation in product- and energy prices, etc. Many supply chains are particularly vulnerable because management is not fully aware of the threats that the system is exposed to, and the vulnerable situation these threats could impose on the supply chain (Asbjørnslett, 2009). This study has limited its vulnerability analysis to comprehend external impulses from emergency slaughter actions and waiting cage bans, and how these effect the product flow and system interactions.

By imposing disruptions to the system, deviations to the system throughput became evident. Results presented in Chapter 7.2.1 showed that a sudden volume increase easily set the system out of balance. With an excessive fleet capacity, will a sudden additional 1000 tonnes of fish make the system flow unstable for approximately 150 hours. However, the sample fleet with four vessels in Section 7.3, used 300 hours in order to obtain the same result after a volume increase of

3000 tonnes. This indicated that the fleet reduction had a considerable effect on the system flexibility. Upon reducing a fleet further, cost and benefit needs to be weighted against each other, finding an equilibrium state where further investment is not beneficial. The factory capacity influence during sudden volume increases became important relative to vessel capacity, as an increase in the latter did not influence the systems ability to absorb shock without an equivalent upgrade of the former. Fleet capacity needs to be in line with the capacity of the next system node, and highlights the importance of the factory as the system vulnerability focal point.

According to Table 7.8 and Figure 7.9 was the backup factory at Nordskaget used. This occurred after emergency slaughter had been included and where the factory capacity was limited. This could imply that if the factory already was struggling to keep the throughput of fish at a sufficient level, external system impulses, such as emergency slaughter, could put the system out of balance. Upon introducing processing vessels, the waiting cages were bypassed, and the need for borrowed capacity was eliminated, clearly relieving the slaughter part of the factory. Subsequently, for systems where the the factory utility already ran high, will incorporation of processing vessels increase the system robustness without having to reconstruct the whole factory.

The extra strain following sudden volume increase during emergency slaughter was eventually loaded onto the trucks, as the cargo vessel already was operating on its full potential. The system thus needed sufficient capacity in every chain element in order to handle the extra volumes. Having a cargo vessel operating in parallel with a fleet of flexible trucks, ensured that the flow of cargo was absorbed without influencing the time spent to market.

By evaluating production flow during waiting cage prohibition simulations, Section 7.2.2 highlights the importance of ensuring sufficient unloading capacity for the fish carriers by the factory. This was the point where the supply chain was narrowed down, and a shortage of infrastructure resources could thus have an impact on the rest of the chain, and consequently, the total cargo throughput. Added berthing capacity is an example of supply chain restructuring which does not necessarily have to be very costly, but can still be very effective, and the knowledge of this bottleneck could avoid unnecessary and expensive procurement of vessels or factory capacity.

8.4 Further Work

This thesis has focused on assessing the practicalities regarding a considerable supply chain expansion, including the feasibility of introducing two new vessel types. It has, however, disregarded all business and operational aspects of the development. These are considerations which need to be taken into account before any alterations are enforced, and could have a large impact on an finding an optimum. This also includes considerations regarding environmental impact, future legislation, etc. Constructing a simulation model which takes all these parameters into account and aims for an optimal solution could provide valuable information.

Along with the solution for the ballasting return legs will evaluations on ship routing be important. This is crucial to the cargo vessel round-trip time and flexibility, and should be assessed both with regard to vessel type and required volumes to get a complete understanding of vessel logistics and availability. The industry also has to perform thorough studies on arrival port. Esbjerg was used as the tentative harbor for this thesis based on current data, but this is not necessarily the most efficient route.

The thesis has assumed a set of vessels with their respective features, but the future vessels could be drastically different. It is thus necessary to evaluate these changes and their impact on the results. In conjunction with vessel and process development will also the handling of fish, and the range of products potentially change. This should be assessed, as new product may need new procedures or trading patterns.

The input data is assumed to be correct, but in order to get precise and reliable results should inputs regarding weather, downtime, fleet size, etc. be exact. An equivalent upstream analysis could also be of interest in order to provide specific data on production volume and fluctuations. Furthermore should a more precise model be developed, and one should aim to avoid the limitations which have been accepted here. This should particularly be with emphasis on creating a realistic integration between live fish carriers and processing vessels, in order to accurately model the interaction between these.

CHAPTER 9

Conclusion

This thesis has incorporated a series of different elements in order to shed light on the challenges and possibilities facing the future of Norwegian aquaculture. The scope has been limited to the downstream chain, starting from the farms and including all significant aspects until reaching the market. By using discrete-event simulation, a realistic model was developed, and various input tested, providing intel about the process. Emphasis has not been on finding an explicit solution, but rather to understand the supply chain interactions, as well as its response to fluctuations.

The study is relevant in conjunction with the fast pace development of the industry, which is expected to boom when/if technology allows the farms to move to more exposed locations. During a period of growth, it is important to be conscious of the total chain throughput and vulnerability. In a feasibility stage is simulation a powerful tool to ascertain the sustainability of a system. This allows for extensive testing without full-scale development. Using discrete event simulation is restricted to analyzing the practical aspect of development, and all commercial, HSE and environment matters are omitted.

In a development phase is understanding system interactions of paramount importance. As a system administrator, one has to know the repercussions of developing one link in the chain, investing in an additional vessel, or expanding the number of loading slots. Upgrading an element which is preceded by a bottleneck will not have an effect of total system throughput. This requires a continuous focus on locating the system weak-link in order to implement effective system upgrades.

A Simulink model was developed block by block, with a series of features embedded in each one. This flexibility allows for a variety of different models, but gives no guarantees as to the correctness of these. It is thus the responsibility of the developer to ensure a correct representation. This can be challenging, as there are many places entities can accumulate, or disappear from, and this needs to be checked for every run. When used correctly and within its limitations, however, is Simulink a very powerful tool, well suited to evaluate throughput and system dynamics.

The results show various noteworthy relations. First, they support Bjørnar Johansen's statement that collaboration is a considerable contributor to the feasibility of introducing cargo vessels. This study incorporated collaboration only through shared capacity in transit, but this alone significantly reduced total time to market, and thus opened up to introducing seaborne transportation at lower production volumes. This is valuable from a sustainability and development perspective.

The introduction of processing vessels also proved to add value to the chain through redundancy and flexibility. These features were particularly evident during crash scenarios, when the steady system was pushed beyond its limitations. This was also the case for parallel trucking routes to Europe, serving as secondary transportation, and backup to the cargo vessels.

Further analyses also showed that the production levels currently are in the early stages of safe introduction of cargo vessels. The implementation of these implies adding a much larger mode than any previously found in the system, and it thus requires the supporting system to be compatible. With the current production level, it was found that long waiting periods in storage had to be expected, and that the time saved maintaining trucks as transportation is considerable. The model does not, however, take into account the strain on the roadways, which is an important element in considering a modal switch.

Upon imposing shock scenarios, the model was subject to drastic changes in operation. Emergency slaughter was simulated by adding a sudden wave of additional cargo, and a potential ban on waiting cages required the live fish carriers to stand by fully laden, allowing the fish to settle on-board after transit. These challenged the system to adapt to a new reality, all the while maintaining a steady flow. The results favored cohesion between all supply chain elements in order to utilize all elements in the best possible way. Shocks in supply chains without bottlenecks were transported safely throughout the system, whereas chains which were strained in the middle were unable to take advantage of proceeding capacity. The value of collaboration manifested itself again through borrowed slaughtering capacity when the primary source failed to clear the total volumes.

The thesis has confirmed that a supply chain is only as strong as its weakest link, which will be important for the industry to keep in mind during further development. Any upgrade must be supported by equivalent infrastructure in order to increase system performance. It was also shown that collaboration and nodal redundancy are important aspects to include in a larger, future supply chain. These make the system more resilient to external influences, and ensure system up-time.

Bibliography

Amble, A. and Ferraz, M. (2012). Sjømat som del av lastgrunnlaget for rorolinje midt-norge – kontinentet. Minimarin for the NSL/WR Project.

Asbjørnslett, B. E. (2009). Assessing the vulnerability of supply chains. In Zsidisin, G. A. and Ritchie, B., editors, *Supply Chain Risk; A Handbook of Assessment, Management, and Performance*, volume 124, chapter 2, pages 15–32. Springer US.

Asbjørnslett, B. E. and Rausand, M. (1999). Assess the vulnerability of your production system. In *Production Planning & Control*, volume 10, pages 219–229. Taylor Francis Group, 3 edition. DOI: 10.1080 / 095372899233181.

Baker, R. J. (2006). *Measure What Matters to Customers*. Wiley, Hoboken, NJ, USA.

Bjørshol, D. R., Bjerkestrand, J. O., Dekkerhus, P. I., and Netter, J. E. (2015). 'Bærekraftige transportløsninger for fersk fisk fra Midt-Norge til kontinentet; prosjektpresentasjon og resultater', *ENOVA, Kysthavnalliansen*, 10.08.2015. Available from: `https://hitra-naeringsforening.no`. [Accessed 20-March-2017].

Caia, J., Liub, X., Xiaob, Z., and Liub, J. (2009). Improving supply chain performance management: A systematic approach to analyzing iterative kpi accomplishment. In *Decision Support Systems*, volume 46, page 512–521. Elsevier B.V., 2 edition.

Campuzano, F. and Mula, J. (2011). *Supply Chain Simulation*. Springer-Verlag, London, 3 edition.

Chandra, C. and Grabis, J. (2016). *Supply chain configuration: Concepts, Solutions, and Applications*. Springer New York, 2 edition.

Cigolini, R., Pero, M., Rossi, T., and Sianesi, A. (2014). Linking supply chain configuration to supply chain performance: A discrete event simulation model. In *Simulation Modelling Practice and Theory*, volume 40, pages 1–11. Elsevier B.V., 1 edition.

Ellingsen, H., Emanuelsson, A., Hognes, E. S., Ziegler, F., and Wither, U. (2009). Energy consumption and greenhouse gas emissions in norwegian seafood export. Technical report, SINTEF Fiskeri og Havbruk AS.

Fiskeri- og Havbruksnæringens Landsforening (2009). 'Moderne Slaking av Laks', *Fiskeri- og Havbruksnæringens Forskningsfond (FHF)*. Available from: `https://sjomatnorge.no`. [Accessed 20-September-2016].

Fon, M. (2017). *EXPOSED Annual report 2016*. Available from `http://exposedaquaculture.no`. [Accessed 15-May-2017].

Hanssen, T.-E. S., Solvoll, G., Nerdal, S., Oddvar Runderem, L. A., and Mathisen, T. A. (2014). Transportstrømmer av fersk laks og ørret fra norge. Technical Report 5, Senter for Innovasjon og Bedriftsøkonomi. ISSN 1890-3584.

Kant, G., Quak, H., Peeters, R., and van Woensel, T. (2016). *Logistics and Supply Chain Innovation*. Springer International Publishing.

Kartverket, G. A. (2017). *Thematic Maps*. Available from: `https://www.kystinfo.no`. [Accessed 03-Feb-2017].

Kvistad, A. (2014). 'Etterlyser offensiv for norsk sjømat', *Sjømatalliansen*, 08

October 2014. Available from: `http://www.sjomatalliansen.no/etterlyser-offensiv-for-norsk-sjomat/`. [Accessed 10-March-2017].

Marine Harvest (2016). *2016 Salmon Industry Handbook*. Available from: `http://marineharvest.com/`. [Accessed 01-April-2017].

Marr, B. (2012). *Key Performance Indicators: the 75 Measures every Manager needs to know*. Pearson, Harlow, Essex, England.

Mathisen, T., Nerdal, S., Solvoll, G., Jørgensen, F., and Hanssen, T.-E. S. (2009). Ferskfisktransporter fra norge til kontinentet transportstrømmer og utfordringer ved bruk av intermodale transportopplegg. Master's thesis, Handelshøgskolen i Bodø, Senter for Innovasjon og Bedriftsøkonomi (SIB AS).

MathWorks (2017). *Simulation and Model-Based Design*. Available from: `https://se.mathworks.com/products/simulink.html`. [Accessed 22-March-2017].

Meteorologisk Institutt (2010). *Significant wave height*. Available from: `https://metlex.met.no/wiki/Signifikant_b%C3%B8lgeh%C3%B8yde`. [Accessed 10-April-2017].

Mindling, K. , Harris, S., Humborstad, O.-B., Akse, L., Chris Noble, Tor Evensen, R. J., and Tobiasen, T. (2011). Slakting direkte fra oppdrettsmerd; tauranga - fase 3. Technical Report 44, Nofima.

Murray-Smith, D. J. (2015). *Testing and Validation of Computer Simulation Models: Principles, Methods and Applications*. Springer International Publishing, 1 edition. DOI 10.1007/978-3-319-15099-4.

Møre Maritime AS (2016). *Future Ship Design; Aquaculture Vessels, Fishing Vessels, Other Specialized Vessels*. IDesign Industrikopi, Dalagata 137 Kristiansund, 2 edition. Company Design Catalogue in Norwegian.

Nordtveit, T. S. (2009). Superkjøling av fisk - en litteraturstudie og prosjektoversikt. Technical report, SINTEF Energiforskning AS. ISBN: 978-82-594-3399-2.

Nygaard, I. (2014). *Havmerd– Lakseoppdrett til havs*. SINTEF Fiskeri og Havbruk. Available from: `http://www.sintef.no/siste-nytt/havmerd-lakseoppdrett-til-havs/` [Accessed 10-April-2017].

Nærings- og Fiskeridepartementet (2015). 'Meld. St. 16 (2014-2015): Stortingsmelding om vekst i havbruksnæringen'. Available at: `https://regjeringen.no`. [Accessed 01-May-2017].

Olafsen, T., Winther, U., Olsen, Y., and Skjermo, J. (2012). Value created from productive oceans in 2050. Technical Report A23299, Royal Norwegian Society of Sciences and Letters (DKNVS) and the Norwegian Academy of Technological Sciences (NTVA).

Pawlewsk, P. (2013). Multimodal approach to model and design supply chain. In *IFAC Proceedings Volumes*, volume 46, pages 2110–2115. Elsevier B.V., 9 edition.

Rausand, M. and Høysand, A. (2004). *System Reliability Theory - Models, Statistical Methods and Applications*. John Wiley Sons, Inc., Hoboken, New Yersey, 2 edition.

Rodrigue, J.-P. and Notteboom, T. (2013). Containerized freight distribution in north america and europe. In Bookbinder, J. H., editor, *International Series in Operations Research Management Science*, volume 181, chapter 10, pages 219–246. Springer New York.

Ross, S. M. (2013). *Simulation*. Elsevier, CA, USA, 5 edition.

Rørtveit, R. and Lilienthal, R. (2016). From fish farm to market; a multimodal simulation of norwegian aquaculture supply chain. Technical report, Institute for Marine Technology, NTNU, Otto Nielsens veg 10, 7052 Trondheim.

SalMar (2017). *Innovamar*. Available from: `http://www.salmar.`

`no/innovamar-fra-drom-til-virkelighet`. [Accessed 02-April-2017].

SFI EXPOSED (2017). *Marin Harvest Exposed Seawatch Bouy*. Available at: `http://exposedaquaculture.no/boyedata/`. [Accessed 06-May-2017].

SFI EXPOSED and SINTEF Ocean (2015). *SFI EXPOSED Home Page*. Available from `http://exposedaquaculture.no`. [Accessed 30-September-2016].

Srai, J. S. and Gregory, M. (2008). A supply network configuration perspective on international supply chain development. In *International Journal of Operations Production Management*, volume 28, pages 386–411. Emerald Group Publishing Limited, 5 edition.

Statistisk Sentralbyrå (2017). *Utenriksøkonomi; Eksport av laks*. Available from: `https://www.ssb.no/statistikkbanken/`. [Accessed 24-March-2017].

Steinum, S. and Budalen, A. (2013). '69.000 laks må nødslaktes i Lofoten', *NRK*, 30 April 2013. Available at: `https://www.nrk.no`. [Accessed 03-May-2017].

Vilko, J. P. and Hallikas, J. M. (2012). Risk assessment in multimodal supply chains. In *International Journal of Production Economics*, volume 140, page 586–595. Elsevier B.V., 2 edition.

Waller, M. and Fawcett, S. (2013). Data science, predictive analytics, and big data: A revolution that will transform supply chain design and management. In *Journal of Buisness Logistics*, volume 34, page 77–84. John Wiley Sons, Inc., 2 edition. DOI 10.1111/jbl.12010.

Appendix

NTNU Trondheim
Norwegian University of Science and Technology
Department of Marine Technology

MASTER'S THESIS IN MARINE TECHNOLOGY

SPRING 2017

Discrete-Event Simulation of a Multimodal Downstream Supply Chain for Future Norwegian Aquaculture

Stud. Tech. Ronja Eide Lilienthal and Ragni Rørtveit

Background

Norway has unique natural advantages for aquaculture production. The SINTEF report "Value created from productive oceans in 2050", emphasizes that, by managing our resources properly, focusing on education and research, and providing our industries with good and predictable framework conditions, we could raise the potential added value in the marine sector to 550 billion NOK in 2050.

Consequently, the Norwegian government has posed an ambitious goal to increase production by five-fold. This brings a lot of new challenges, the first being finding sufficient locations for the farms, moreover, on-shore infrastructure will require significant development. This includes both factories and transportation to market, which today is done by trucking directly to the consumers.

Salmon is fresh produce, and time is of essence. With a tight schedule and large production volumes come the need for constant high system efficiency and large production- and transportation capabilities. The system throughput and flexibility is highly dependent on value chain composition and utilization. In connecting different transportation modes, the slaughter- and processing facilities serves as a supply chain focal point.

With increased production, the rest of the supply chain has to expand correspondingly. The Norwegian aquaculture industry has grown at an exceptional rate, leaving very little time for long-term elements, such as slaughter facilities, to adapt. Today the farmers of the Hitra/Frøya area solve this discrepancy by sharing slaughter capacity amongst each other. However, if the five-fold production goal is going to be met, the supply chain must be more streamlined and standardized.

The industry is currently at a point of no expansion due to strict government regulations imposed in order to control problems related to disease and eco-pollution. However, the willingness to experiment is at an all-time high, proportional to recent salmon prices, and it is thus assumed that the technological breakthrough is imminent. If/when this happens, the logistics infrastructure has to be ready for a rapid expansion, and it is this new market, our master thesis aims to comprehend.

NTNU Trondheim
Norwegian University of Science and Technology
Department of Marine Technology

Objective

This master's thesis aimed to serve as a feasibility study, using discrete- event simulation as a tool to give intel into the downstream supply chain interaction, and subsequently unveil potential challenges facing Norwegian aquaculture in the future. The thesis was a continuation of the authors' preliminary project thesis, which emphasized the flow to and from the slaughter- and processing facilities. This project continued the preceding work by expanding to the complete multimodal downstream supply chain composition, evaluating system dynamics and flexibility based on composition and throughput.

Modelling a continuous movement of entities made it possible to locate system weak links, as well as evaluate whether new transportation modes provided added value to the supply chain. The goal of this project was not to find an optimal solution, but rather to provide an understanding of the system as a whole, and as such be useful when the industry expands.

Tasks

The thesis aims to cover the following tasks:

a. Describe the motivation and relevance behind the thesis.

b. Present the multimodal downstream aquaculture supply chain in mid-Norway, and possible future developments based on the assumption that production will increase by five-fold over the next decades.

c. Perform a state of the art analysis, both regarding multimodal supply chains, and the use of discrete-event simulation to evaluate these. Emphasis should be put on assessing information about the method's possibilities and limitations as a flow evaluation tool.

d. Access simulation input data through industry reports and contact with relevant actors.

e. Use Simulink's discrete-event library to develop a model which represents the supply chain as realistically as possible, including future concepts.

f. Validate the model and outputs, based on real-life data.

g. Retrieve results based on simulation runs with various fleet compositions and scenarios. Include two main approaches, one evaluating a steady system, and one imposing operation fluctuations.

h. Discuss the results with regard to system interaction, vulnerability and flexibility.

Supervision:

Main supervisor: Bjørn Egil Asbjørnslett

Deadline: 09.06.2017

A MATLAB Codes

A.1 MATLAB Simulation Run Script

```
%--------- Read Input From Excel--------------------------
 Input1 = xlsread('inputdata.xlsx');
 Input2 = xlsread('inputdata.xlsx', 'Sheet2');

 %-------- Run weather generation ------------------------
 run('mcwave')
 run('mcwave_exp')
 run('mcwave_CV')

 %---------- SCENARIO -------------------------------------
%Production Volume
    Sc_production = Input1(1,2);
%Weather
    %CV
    Sc_weather_CV_per_1 = Input1(5,2);
    Sc_weather_CV_per_2 = Input1(6,2);
    Sc_weather_0_CV = Input1(7,3);
    Sc_weather_CV_Hs_1 = Input1(5,3);
    Sc_weather_CV_Hs_2 = Input1(6,3);
    %LFC and Proc
    Sc_weather_per_1 = Input1(5,5);
    Sc_weather_per_2 = Input1(6,5);
    Sc_weather_Hs_1 = Input1(5,6);
    Sc_weather_Hs_2 = Input1(6,6);
    Sc_weather_0 = Input1(7,6);
    %Cage operations
    Sc_weather_trad_limit = Input1(5,8);
    Sc_weather_exp_limit = Input1(6,8);
    Sc_weather_exp_trad_ratio = Input1(8,8);
%Cooperation
    Sc_coop = Input1(11,2);
    Sc_coop_wait_limit = Input1(12,2);
```

```
   if Sc_coop == 1
       CoopFactor = 0.333;
   else
       CoopFactor = 1;
   end

% Emergency Slaughter
   Sc_emergency_tonnes_trad = Input1(18,2);
   Sc_emergency_nr_trad = Input1(19,2);
   Sc_emergency_tonnes_exp = Input1(18,5);
   Sc_emergency_nr_exp = Input1(19,5);

   %Vestor Creation
    SlaughterVec = zeros(1,Sc_emergency_tonnes_trad+1);

    if Sc_emergency_tonnes_trad == 0
        SlaughterVec(1)=0;
    else %Impose Signal after 4380 hours
       SlaughterVec(1) = 4380 + round(8760*rand(1,1));
    end


    SlaughterVec_exp = zeros(1,Sc_emergency_tonnes_exp+1);
    if Sc_emergency_tonnes_exp == 0
        SlaughterVec_exp(1)=0;
    else %Impose Signal after 4380 hours
       SlaughterVec_exp(1) = 4380 + round(8760*rand(1,1));
    end

% Waiting cage prohib
   Sc_waitcage = Input1(15,2);


% ------------- FLEET ------------------------------------

   % PROCESSING VESSEL
   Proc_amount = Input1(29,2);
   Proc_cap = Input1(29,3);
```

```
  Proc_speed = Input1(29,4);
  Proc_down = Input1(29,5);

  % LIVE FISH CARRIER
  LFC_amount = Input1(30,2);
  LFC_cap = Input1(30,3);
  LFC_speed = Input1(30,4);
  LFC_down = Input1(30,5);

  % SLAUGHTERING FACILITIES
  SF_cap = Input1(31,3);
  SF_down = Input1(31,5);

  % TRUCKS
  Truck_amount = Input1(32,2);
  Truck_cap = Input1(32,3);
  Truck_speed = Input1(32,4);
  Truck_down = Input1(32,5);

  % LIVE FISH CARRIER
  CC_amount = Input1(33,2);
  CC_cap = Input1(33,3);
  CC_speed = Input1(33,4);
  CC_down = Input1(33,5);


%------------- MARKET DISTRUBUTION -------------------

    M_TotWest = Input2(11,9);
    M_TotGardemoen = Input2(12,9);
    M_TotEast = Input2(13,9);
    M_TotNordic = Input2(14,9);

    M_DK = Input2(4,9);
    M_BL_NL = Input2(5,9);
    M_GER = Input2(6,9);
    M_FR = Input2(7,9);
    M_ES = Input2(8,9);
    M_UK = Input2(9,9);
```

```
     M_PT = Input2(10,9);

     M_TotWest_II = Input2(11,10);
     M_TotEast_II = Input2(13,10);
     M_TotGardemoen_II = Input2(12,11);
     M_TotNordic_II = Input2(14,11);

     M_DK_II = Input2(4,10);
     M_BL_NL_II = Input2(5,10);
     M_GER_II = Input2(6,10);
     M_FR_II = Input2(7,10);
     M_ES_II = Input2(8,10);
     M_UK_II = Input2(9,10);
     M_PT_II = Input2(10,10);

%----------Driving Distances ---------------------

% Nutical Miles, Cargo Vessel, Sandstand - Esbjerg
    Nm_Hitra_DK = Input1(67,1);
   %From J sten ya
    Km_DK = Input2(4,6);
     Km_BL_NL = Input2(5,6);
     Km_GER = Input2(6,6);
     Km_FR = Input2(7,6);
     Km_ES = Input2(8,6);
     Km_UK = Input2(9,6);
     Km_PT = Input2(10,6);

     Km_Gardemoen = Input2(12,6);
     Km_PL = Input2(13,6);
     Km_North = Input2(14,6);

     %From Esbjerg
     Km_DK_II = Input2(4,7);
     Km_BL_NL_II = Input2(5,7);
     Km_GER_II = Input2(6,7);
     Km_FR_II = Input2(7,7);
     Km_ES_II = Input2(8,7);
     Km_UK_II = Input2(9,7);
```

```
     Km_PT_II = Input2(10,7);

     Km_PL_II = Input2(13,7);

   %------ Berths, Manouvering and loading operations----------
   CC_moor = Input1(70,1);
   Proc_moor = Input1(74,1);
   LFC_moor = Input1(72,1);

   Esbjerg_cap = Input1(74,4);
   Josten_cap = Input1(70,4);
   Nord_cap = Input1(72,4);


   %------FARM LOCATIONS-----------------
   Jostenoya = zeros(1,18);
   Nordskaget = zeros(1,18);

   for i = 1:18
   Jostenoya(i) = Input1(40+i,2);
   Nordskaget(i) = Input1(40+i,3);
   i = i+1;
   end

   Exposed_Jostenoya = zeros(1,1);
   Exposed_Nordskaget = zeros(1,1);
for i = 1:1
   Exposed_Jostenoya(i) = Input1(40+i,6);
   Exposed_Nordskaget(i) = Input1(40+i,7);
   i = i+1;
end
```

A.2 MATLAB Code for Markov Chain modelling of Weather Conditions

Original code constructed by engineer Knut Støvler in conjunction with the course TMR4565, fall 2016, for the Department of Marine Technology NTNU.

```
tic;
% Reads the csv file into a matrix
A = xlsread('wave_data_trad.xlsx');

% Set first and last month. Can be the same

% Loop over all the rows in the matrix
for i = 1:size(A,1)
Hs(i) = A(i,1);
end

% Set number of states in the markov chain
numStates = 10;

% Find upper limit for Hs values and divide the values into even bins
ul = max(Hs);
% Find state ranges - first state [0,stateRange] and so on
stateRange = ul / numStates;
% State values - stateRange, 2xstateRange and so on up til ul
stateValues = stateRange:stateRange:ul;
% Initialize 1D-matrix holding the state of each data point
HsState = zeros(length(Hs),1);

% Find each data points state
for i = 1:length(Hs)
    % For each data point
    for j = 1:numStates
        % For each state
        if Hs(i) <= stateValues(j)
            % Data point is in state j
            HsState(i) = j;
```

```
            % This data point is categorized, so we break and move to the
            % next data point
            break;
        end
    end
end

% Find transitions
transitions = zeros(numStates);
for t = 1:length(HsState)-1
    % HsState(t) represents the state and HsState(t+1) represents the state
    % it transitions to
transitions(HsState(t),HsState(t+1))=transitions(HsState(t),HsState(t+1))+1;
end

P = transitions;
% Normalize each row in the transition matrix so each row sums to 1
for i = 1:numStates
    P(i,:) = P(i,:) / sum( P(i,:) );
end

% Check to see if there are any absorbing states
% i.e. P(i,j) == 1 where i=j
absorbstate = zeros(numStates);
for i = 1:numStates
    for j = 1:numStates
        if P(i,j) == 1
            absorbstate(i,j) = absorbstate(i,j) + 1;
        end
    end
end
if sum(sum(absorbstate)) >= 1
    error('Absorbing states. Consider reducing number of states.');
end

%% Transition matrix is now ready in P

% How many state transitions to perform
% Lower this number to show how fewer replications affects results
```

```
% for example, 100, 1000, length(Hs), 10000
numReplications = 10000;

% Random number seed
rng(1235);

% Set starting state - should sample randomly
state = randi(numStates);

states = zeros(numReplications,1);

for i = 1:numReplications
    % Sample a new random value in range [0,1]
    r = rand();

    for j = 1:numStates
        prob = 0;
        % Accumulate probabilities
        for k = 1:j
            prob = prob + P(state,k);
        end

        if r <= prob
            % New state is found, j
            state = j;

            % Store the state we transition to
            states(i) = j;

            % Break ends the current for loop, and returns to the outer
            % loop, which will sample a new random value and start over
            break;
        end
    end
end
% If needed, Hs can be compared directly to the simulated results
simValues = zeros(numReplications,1);
for i = 1:numReplications
    simValues(i) = (states(i) * stateRange) - stateRange/2;
```

```
end
toc;
```

A.3 Codes for calculating Weather Influence on Cargo Vessel Speed.

```
%Separates data set into different seasons; spring, summer, fall and winter
%Separates the data into different seasons

%Spring = March (3), April (4) and May (5)
%Summer = June (6), July (7) and August (8)
%Fall = September (9), October (10) and November (11)
%Winter = December (12), January (1) and February (2)

%Read in wanted text-file. Make sure HS-column is read correctly from
%Excel.
A = csvread('wave_data_CV.csv',23,0);

SpringT = 1;    %counter
SummerT = 1;    %counter
FallT = 1;      %counter
WinterT = 1;    %counter

Hs = A(:,7);
for i = 1:length(A(:,2))
    if A(i,2) == 3 || A(i,2) == 4 || A(i,2) == 5
        spring(SpringT) = Hs(i);
        SpringT = SpringT + 1;
    elseif A(i,2) == 6 || A(i,2) == 7 || A(i,2) == 8
        summer(SummerT) = Hs(i);
        SummerT = SummerT + 1;
    elseif A(i,2) == 9 || A(i,2) == 10 || A(i,2) == 11
        fall(FallT) = Hs(i);
        FallT = FallT + 1;
    elseif A(i,2) == 12 || A(i,2) == 1 || A (i,2) == 2
        winter(WinterT) = Hs(i);
        WinterT = WinterT + 1;
    end
end

%Set Waiting Periods based on Hs criterion:
```

```
Hs = 9; % Set operable criterion for Hs


I = 3; %time interval between each data
Tr = 12; % reference Period
year = 1992:2014;
hours = 30*24; %hours each month

%% Waiting periods Spring
StormS = 0; %Counter for hours within one storm period
CalmS = 0; %Counter for hours within one calm period
dummyS = 0; %Counter for place in SpringStorm vector
dummyC = 0; %Counter for place in SpringCalm vector
SpringStorm = []; %The total number of hours for each storm period
SpringCalm = []; %The total number of hours for each calm period

for i = 1:length(spring)
    if spring(i) > Hs %Storm
        if CalmS == 0
            CalmS = 0;
            StormS = StormS + I; %add on time interval
        else
            dummyC = dummyC + 1;
            SpringCalm(dummyC) = CalmS;
            CalmS = 0;
            StormS = StormS + I;
        end
    else
        if StormS == 0
            StormS = 0;
            CalmS = CalmS + I;
        else
            dummyS = dummyS + 1;
            SpringStorm(dummyS) = StormS;
            StormS = 0;
            CalmS = CalmS + I;
        end
    end
end
```

```
%counts for last storm and calm period
if CalmS ˜= 0
    dummyC = dummyC + 1;
    SpringCalm(dummyC) = CalmS;
end

if StormS ˜= 0
    dummyS = dummyS + 1;
    SpringStorm(dummyS) = StormS;
end

WaitS = sum(SpringStorm); %Waiting time equals the length of storm period
% + the calm period that is shorter than the reference period
for j = 1:length(SpringCalm)
    if SpringCalm(j) < Tr
        WaitS = WaitS + SpringCalm(j);
    end
end

Downtime_S = WaitS/length(year)/3; %downtime each year each month in hours
DownPercent_S = Downtime_S/hours*100; %downtime each year each
                                       % month in percent


%% Summer
StormSum = 0; %Counter for hours within one storm period
CalmSum = 0; %Counter for hours within one calm period
dummyS = 0; %Counter for place in SummerStorm vector
dummyC = 0; %Counter for place in SummerCalm vector
SummerStorm = []; %The total number of hours for each storm period
SummerCalm = []; %The total number of hours for each calm period

for i = 1:length(summer)
    if summer(i) > Hs %Storm
        if CalmSum == 0
            CalmSum = 0;
            StormSum = StormSum + I; %add on time interval
        else
```

```
            dummyC = dummyC + 1;
            SummerCalm(dummyC) = CalmSum;
            CalmSum = 0;
            StormSum = StormSum + I;
        end
    else
        if StormSum == 0
            StormSum = 0;
            CalmSum = CalmSum + I;
        else
            dummyS = dummyS + 1;
            SummerStorm(dummyS) = StormSum;
            StormSum = 0;
            CalmSum = CalmSum + I;
        end
    end
end

%counts for last storm and calm period
if CalmSum ~= 0
    dummyC = dummyC + 1;
    SummerCalm(dummyC) = CalmSum;
end

if StormSum ~= 0
    dummyS = dummyS + 1;
    SummerStorm(dummyS) = StormSum;
end

WaitSum = sum(SummerStorm); %Waiting time equals the length of storm
   %period + the calm period that is shorter than the reference period
for j = 1:length(SummerCalm)
    if SummerCalm(j) < Tr
        WaitSum = WaitSum + SummerCalm(j);
    end
end

Downtime_Sum = WaitSum/length(year)/3; %downtime each year each month
                                        %in hours
```

```
DownPercent_Sum = Downtime_Sum/hours*100; %downtime each year each month
                                         % in percent

%% Fall
StormF = 0; %Counter for hours within one storm period
CalmF = 0; %Counter for hours within one calm period
dummyS = 0; %Counter for place in FallStorm vector
dummyC = 0; %Counter for place in FallCalm vector
FallStorm = []; %The total number of hours for each storm period
FallCalm = []; %The total number of hours for each calm period

for i = 1:length(fall)
    if fall(i) > Hs %Storm
        if CalmF == 0
            CalmF = 0;
            StormF = StormF + I; %add on time interval
        else
            dummyC = dummyC + 1;
            FallCalm(dummyC) = CalmF;
            CalmF = 0;
            StormF = StormF + I;
        end
    else
        if StormF == 0
            StormF = 0;
            CalmF = CalmF + I;
        else
            dummyS = dummyS + 1;
            FallStorm(dummyS) = StormF;
            StormF = 0;
            CalmF = CalmF + I;
        end
    end
end

%counts for last storm and calm period
if CalmF ~= 0
    dummyC = dummyC + 1;
    FallCalm(dummyC) = CalmF;
```

```
end

if StormF ~= 0
    dummyS = dummyS + 1;
    FallStorm(dummyS) = StormF;
end

WaitF = sum(FallStorm); %Waiting time equals the length of storm period
            % + the calm period that is shorter than the reference period
for j = 1:length(FallCalm)
    if FallCalm(j) < Tr
        WaitF = WaitF + FallCalm(j);
    end
end

Downtime_F = WaitF/length(year)/3; %downtime each year each month
                                    % in hours
DownPercent_F = Downtime_F/hours*100; %downtime each year each month
                                    % in percent

%% Winter
StormW = 0; %Counter for hours within one storm period
CalmW = 0; %Counter for hours within one calm period
dummyS = 0; %Counter for place in WinterStorm vector
dummyC = 0; %Counter for place in WinterCalm vector
WinterStorm = []; %The total number of hours for each storm period
WinterCalm = []; %The total number of hours for each calm period

for i = 1:length(winter)
    if winter(i) > Hs %Storm
        if CalmW == 0
            CalmW = 0;
            StormW = StormW + I; % add on time interval
        else
            dummyC = dummyC + 1;
            WinterCalm(dummyC) = CalmW;
            CalmW = 0;
            StormW = StormW + I;
        end
```

```
    else
        if StormW == 0
            StormW = 0;
            CalmW = CalmW + I;
        else
            dummyS = dummyS + 1;
            WinterStorm(dummyS) = StormW;
            StormW = 0;
            CalmW = CalmW + I;
        end
    end
end

%counts for last storm and calm period
if CalmW ~= 0
    dummyC = dummyC + 1;
    WinterCalm(dummyC) = CalmW;
end

if StormW ~= 0
    dummyS = dummyS + 1;
    WinterStorm(dummyS) = StormW;
end

WaitW = sum(WinterStorm); %Waiting time equals the length of storm period
            % + the calm period that is shorter than the reference period
for j = 1:length(WinterCalm)
    if WinterCalm(j) < Tr
        WaitW = WaitW + WinterCalm(j);
    end
end

Downtime_W = WaitW/length(year)/3; %downtime each year each month
                                    %in hours
DownPercent_W = Downtime_W/hours*100; %downtime each year each month
                                    % in percent

%% Print downtime in hours and percent
Downtime(1,:) = [Downtime_S Downtime_Sum Downtime_F Downtime_W];
```

```
Downtime(2,:) = [DownPercent_S DownPercent_Sum DownPercent_F DownPercent_W];

printmat(Downtime,'DOWNTIME each month pr. season','[hours] [%]','Spring Summer



%Separates the data into different seasons

%Spring = March (3), April (4) and May (5)
%Summer = June (6), July (7) and August (8)
%Fall = September (9), October (10) and November (11)
%Winter = December (12), January (1) and February (2)

%Read in wanted text-file. Make sure HS-column is read correctly from
%Excel.
A = csvread('wave_data_CV.csv',23,0);

SpringT = 1;    %counter
SummerT = 1;    %counter
FallT = 1;      %counter
WinterT = 1;    %counter

Hs = A(:,7);
for i = 1:length(A(:,2))
    if A(i,2) == 3 || A(i,2) == 4 || A(i,2) == 5
        spring(SpringT) = Hs(i);
        SpringT = SpringT + 1;
    elseif A(i,2) == 6 || A(i,2) == 7 || A(i,2) == 8
        summer(SummerT) = Hs(i);
        SummerT = SummerT + 1;
    elseif A(i,2) == 9 || A(i,2) == 10 || A(i,2) == 11
        fall(FallT) = Hs(i);
        FallT = FallT + 1;
    elseif A(i,2) == 12 || A(i,2) == 1 || A (i,2) == 2
        winter(WinterT) = Hs(i);
        WinterT = WinterT + 1;
    end
end
```

```
I = 3; %time interval between each data
Hs_start = 0; %minimum Hs plotted
Hs_max = 10; %maximum Hs plotted
Tr = 12; %Reference Period
year = 1992:2014;
months = 3; % months each season

%% Waiting periods Spring
StormS = 0; %Counter for hours within one storm period
CalmS = 0; %Counter for hours within one calm period
dummyS = 0; %Counter for place in SpringStorm vector
dummyC = 0; %Counter for place in SpringCalm vector
SpringStorm = []; %The total number of hours for each storm period
SpringCalm = []; %The total number of hours for each calm period
t = 1; %Makes us able to separate each Hs
hours = (31+31+30)*24/3; %hours each month

for Hs = Hs_start:0.25:Hs_max
    for i = 1:length(spring)
        if spring(i) > Hs %Storm
            if CalmS == 0
                CalmS = 0;
            else
                dummyC = dummyC + 1;
                SpringCalm(dummyC) = CalmS;
                CalmS = 0;
            end
            StormS = StormS + I; %add on time interval
        else
            if StormS == 0
                StormS = 0;
            else
                dummyS = dummyS + 1;
                SpringStorm(dummyS) = StormS;
                StormS = 0;
            end
            CalmS = CalmS + I;
        end
```

```
    end

    %counts for last storm and calm period
    if CalmS ~= 0
        dummyC = dummyC + 1;
        SpringCalm(dummyC) = CalmS;
    end

    if StormS ~= 0
        dummyS = dummyS + 1;
        SpringStorm(dummyS) = StormS;
    end

    WaitS = sum(SpringStorm); %Waiting time equals the length of storm
    %period + the calm period that is shorter than the reference period
    for j = 1:length(SpringCalm)
        if SpringCalm(j) < Tr
            WaitS = WaitS + SpringCalm(j);
        end
    end

    Downtime_S(t) =WaitS/length(year)/months;  %downtime each year pr.
                                               %month in percent
    DownPercent_S(t) =Downtime_S(t)/hours*100; %downtime each year pr.
                                               % month in percent

    t = t+1;
    StormS = 0; %all counters are set to zero
    CalmS = 0;
    dummyS = 0;
    dummyC = 0;
    SpringStorm = [];
    SpringCalm = [];
end

%% Summer
StormSum = 0; %Counter for hours within one storm period
CalmSum = 0; %Counter for hours within one calm period
dummyS = 0; %Counter for place in SummerStorm vector
```

```
dummyC = 0; %Counter for place in SummerCalm vector
SummerStorm = []; %The total number of hours for each storm period
SummerCalm = []; %The total number of hours for each calm period
t = 1; %Makes us able to separate each HsOp
hours = (31+31+30)*24/3; %hours each month

for Hs = Hs_start:0.25:Hs_max
    for i = 1:length(summer)
        if summer(i) > Hs %Storm
            if CalmSum == 0
                CalmSum = 0;
                StormSum = StormSum + I; %add on time interval
            else
                dummyC = dummyC + 1;
                SummerCalm(dummyC) = CalmSum;
                CalmSum = 0;
                StormSum = StormSum + I;
            end
        else
            if StormSum == 0
                StormSum = 0;
                CalmSum = CalmSum + I;
            else
                dummyS = dummyS + 1;
                SummerStorm(dummyS) = StormSum;
                StormSum = 0;
                CalmSum = CalmSum + I;
            end
        end
    end

    %counts for last storm and calm period
    if CalmSum ~= 0
        dummyC = dummyC + 1;
        SummerCalm(dummyC) = CalmSum;
    end

    if StormSum ~= 0
        dummyS = dummyS + 1;
```

```
        SummerStorm(dummyS) = StormSum;
    end

    WaitSum = sum(SummerStorm); %Waiting time equals the length of storm
%period + the calm period that is shorter than the reference period
    for j = 1:length(SummerCalm)
        if SummerCalm(j) < Tr
            WaitSum = WaitSum + SummerCalm(j);
        end
    end

    Downtime_Sum(t) = WaitSum/length(year)/months; %downtime each year
                                                   %each month in percent
    DownPercent_Sum(t) = Downtime_Sum(t)/hours*100;%downtime each year
                                                   %each month in percent

    t = t+1;
    StormSum = 0;  %all counters are set to zero
    CalmSum = 0;
    dummyS = 0;
    dummyC = 0;
    SummerStorm = [];
    SummerCalm = [];
end

%% Fall
StormF = 0; %Counter for hours within one storm period
CalmF = 0; %Counter for hours within one calm period
dummyS = 0; %Counter for place in FallStorm vector
dummyC = 0; %Counter for place in FallCalm vector
FallStorm = []; %The total number of hours for each storm period
FallCalm = []; %The total number of hours for each calm period
t = 1; %Makes us able to separate each HsOp
hours = (30+31+30)*24/3; %hours each month

for Hs = Hs_start:0.25:Hs_max

    for i = 1:length(fall)
        if fall(i) > Hs %Storm
```

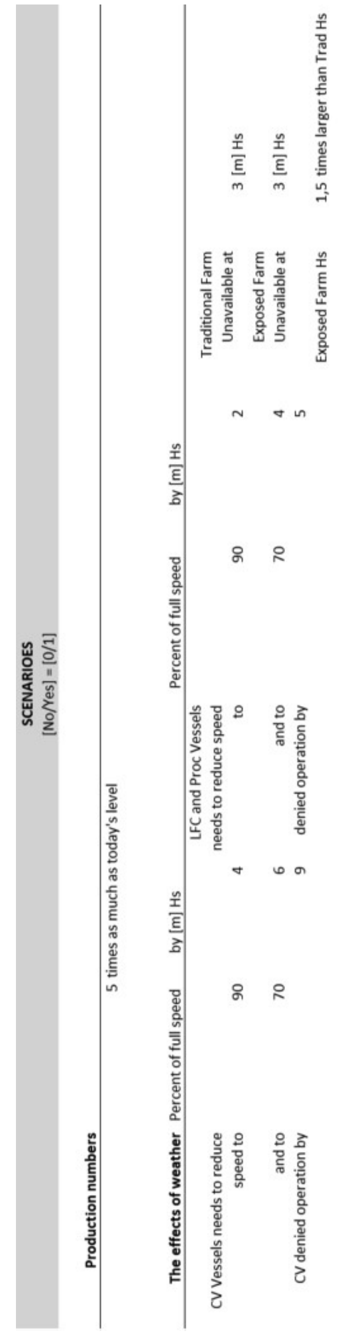
```
        if CalmF == 0
            CalmF = 0;
            StormF = StormF + I; %add on time interval
        else
            dummyC = dummyC + 1;
            FallCalm(dummyC) = CalmF;
            CalmF = 0;
            StormF = StormF + I;
        end
    else
        if StormF == 0
            StormF = 0;
            CalmF = CalmF + I;
        else
            dummyS = dummyS + 1;
            FallStorm(dummyS) = StormF;
            StormF = 0;
            CalmF = CalmF + I;
        end
    end
end

%counts for last storm and calm period
if CalmF ~= 0
    dummyC = dummyC + 1;
    FallCalm(dummyC) = CalmF;
end

if StormF ~= 0
    dummyS = dummyS + 1;
    FallStorm(dummyS) = StormF;
end

WaitF = sum(FallStorm); %Waiting time equals the length of storm
%period + the calm period that is shorter than the reference period
for j = 1:length(FallCalm)
    if FallCalm(j) < Tr
        WaitF = WaitF + FallCalm(j);
    end
```

```
    end

    Downtime_F(t) = WaitF/length(year)/months; %downtime each year
                                               %each month in percent
    DownPercent_F(t) = Downtime_F(t)/hours*100;%downtime each year
                                               %each month in percent

    t = t+1;
    StormF = 0;  %all counters are set to zero
    CalmF = 0;
    dummyS = 0;
    dummyC = 0;
    FallStorm = [];
    FallCalm = [];
end

%% Winter
StormW = 0; %Counter for hours within one storm period
CalmW = 0; %Counter for hours within one calm period
dummyS = 0; %Counter for place in WinterStorm vector
dummyC = 0; %Counter for place in WinterCalm vector
WinterStorm = []; %The total number of hours for each storm period
WinterCalm = []; %The total number of hours for each calm period
t = 1; %Makes us able to separate each HsOp
hours = (31+31+30)*24/3; %hours each month

for Hs = Hs_start:0.25:Hs_max
    for i = 1:length(winter)
        if winter(i) > Hs %Storm
            if CalmW == 0
                CalmW = 0;
                StormW = StormW + I; % add on time interval
            else
                dummyC = dummyC + 1;
                WinterCalm(dummyC) = CalmW;
                CalmW = 0;
                StormW = StormW + I;
            end
        else
```

```
            if StormW == 0
                StormW = 0;
                CalmW = CalmW + I;
            else
                dummyS = dummyS + 1;
                WinterStorm(dummyS) = StormW;
                StormW = 0;
                CalmW = CalmW + I;
            end
        end
    end

    %counts for last storm and calm period
    if CalmW ~= 0
        dummyC = dummyC + 1;
        WinterCalm(dummyC) = CalmW;
    end

    if StormW ~= 0
        dummyS = dummyS + 1;
        WinterStorm(dummyS) = StormW;
    end

    WaitW = sum(WinterStorm); %Waiting time equals the length of storm
%period + the calm period that is shorter than the reference period
    for j = 1:length(WinterCalm)
        if WinterCalm(j) < Tr
            WaitW = WaitW + WinterCalm(j);
        end
    end

    Downtime_W(t) = WaitW/length(year)/months;  %downtime each year
                                                %each month in percent
    DownPercent_W(t) = Downtime_W(t)/hours*100; %downtime each year each
                                                %month in percent

    t = t+1;
    StormW = 0; %all counters are set to zero
    CalmW = 0;
```

```
    dummyS = 0;
    dummyC = 0;
    WinterStorm = [];
    WinterCalm = [];
end

%% Visualisation
Hs = Hs_start:0.25:Hs_max;
figure(5);clf
subplot(1,2,1)
plot(Hs,Downtime_S,'k*-')
hold on
plot(Hs,Downtime_Sum, 'k>-')
plot(Hs,Downtime_F, 'k--')
plot(Hs,Downtime_W,'k')

title('Average Time w/Necessary Speed Reduction pr Month pr Season[h]')

axis([0 Hs_max 0 800])
legend('Spring','Summer','Fall','Winter')
xlabel('Significant wave Height [m]')
ylabel('Up-Time [hours]')
set(gca,'fontsize',15)
box on
set(gcf,'color','w')

subplot(1,2,2)
plot(Hs,DownPercent_S,'k*-')
hold on
plot(Hs,DownPercent_Sum, 'k>-')
plot(Hs,DownPercent_F, 'k--')
plot(Hs,DownPercent_W,'k')

title('Average Time w/Necessary Speed Reduction pr Month pr Season [%]')
axis([0 Hs_max 0 100])
legend('Spring','Summer','Fall','Winter');
xlabel('Significant wave Height [m]')
ylabel('Up- Time [%]')
set(gca,'fontsize',15)
```

B Excel Input Spreadsheet



SCENARIOES
[No/Yes] = [0/1]

Production numbers

5 times as much as today's level

The effects of weather	Percent of full speed	by [m] Hs		Percent of full speed	by [m] Hs		
CV Vessels needs to reduce speed to	90	4	LFC and Proc Vessels needs to reduce speed to	90	2	Traditional Farm Unavailable at	3 [m] Hs
and to	70	6	and to	70	4	Exposed Farm Unavailable at	3 [m] Hs
CV denied operation by		9	denied operation by		5	Exposed Farm Hs	1,5 times larger than Trad Hs

Cooperation

Cooperation between actors	1	[No/Yes] = [0/1]
Waiting cage capasity limit Jøstenøya	1 500	tonnes

Waiting Cage prohibition

Prohibition against waiting cages	0	[No/Yes] = [0/1]

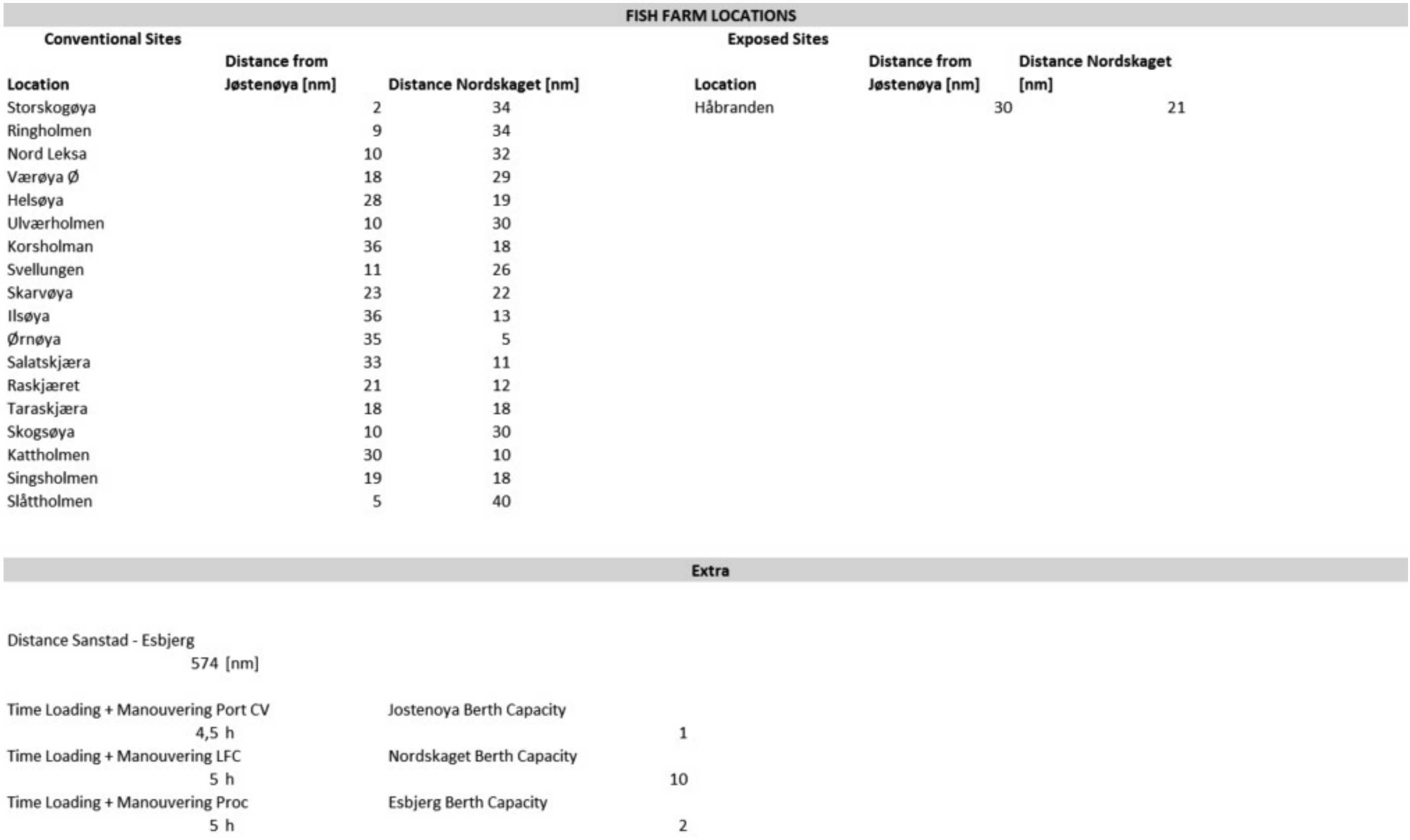
Emergency Slaughter

Traditional Locations	0 tonness of fish 0 times pr year	Exposed Locations	0 tonnes of fish 0 times pr year

FLEET COMPOSITIONS

	Amount *[number of vessels]*	**Capacity** [tonnes]	**Speed** [knots and km]	**Scheduled down-time** [hours/year]
Processing vessels	3	500	12	140
Live fish carriers	1	400	12	160
Slaughterhouses	-	95	-	-
Trucks	100	20	60	-
Cargo carriers	1	2000	16	100

Figure 9.1: Scenario Input and Fleet Composition.



FISH FARM LOCATIONS

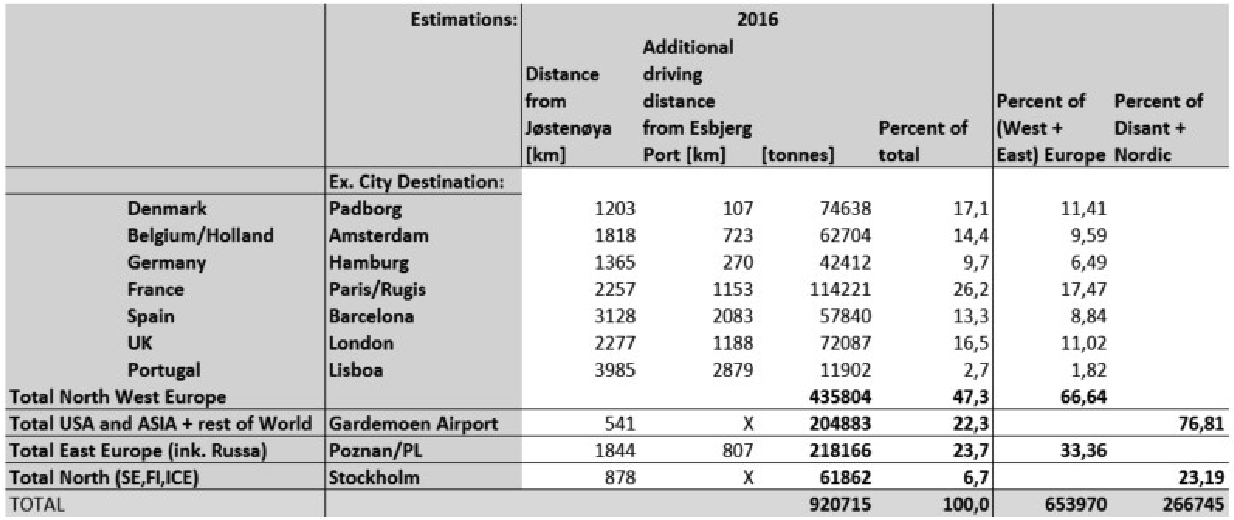
Conventional Sites			Exposed Sites		
Location	Distance from Jøstenøya [nm]	Distance Nordskaget [nm]	Location	Distance from Jøstenøya [nm]	Distance Nordskaget [nm]
Storskogøya	2	34	Håbranden	30	21
Ringholmen	9	34			
Nord Leksa	10	32			
Værøya Ø	18	29			
Helsøya	28	19			
Ulværholmen	10	30			
Korsholman	36	18			
Svellungen	11	26			
Skarvøya	23	22			
Ilsøya	36	13			
Ørnøya	35	5			
Salatskjæra	33	11			
Raskjæret	21	12			
Taraskjæra	18	18			
Skogsøya	10	30			
Kattholmen	30	10			
Singsholmen	19	18			
Slåttholmen	5	40			

Extra

Distance Sanstad - Esbjerg
574 [nm]

Time Loading + Manouvering Port CV 4,5 h	Jostenoya Berth Capacity 1
Time Loading + Manouvering LFC 5 h	Nordskaget Berth Capacity 10
Time Loading + Manouvering Proc 5 h	Esbjerg Berth Capacity 2

Figure 9.2: Farm Locations and Infrastructure Capacities.



	Estimations:	2016					
		Distance from Jøstenøya [km]	Additional driving distance from Esbjerg Port [km]	[tonnes]	Percent of total	Percent of (West + East) Europe	Percent of Disant + Nordic
	Ex. City Destination:						
Denmark	Padborg	1203	107	74638	17,1	11,41	
Belgium/Holland	Amsterdam	1818	723	62704	14,4	9,59	
Germany	Hamburg	1365	270	42412	9,7	6,49	
France	Paris/Rugis	2257	1153	114221	26,2	17,47	
Spain	Barcelona	3128	2083	57840	13,3	8,84	
UK	London	2277	1188	72087	16,5	11,02	
Portugal	Lisboa	3985	2879	11902	2,7	1,82	
Total North West Europe				**435804**	**47,3**	**66,64**	
Total USA and ASIA + rest of World	**Gardemoen Airport**	541	X	**204883**	**22,3**		**76,81**
Total East Europe (ink. Russa)	**Poznan/PL**	1844	807	**218166**	**23,7**	**33,36**	
Total North (SE,FI,ICE)	**Stockholm**	878	X	**61862**	**6,7**		**23,19**
TOTAL				**920715**	**100,0**	**653970**	**266745**

Figure 9.3: Market Distribution, excluding List of Individual Country Imports.

C Simulink Model

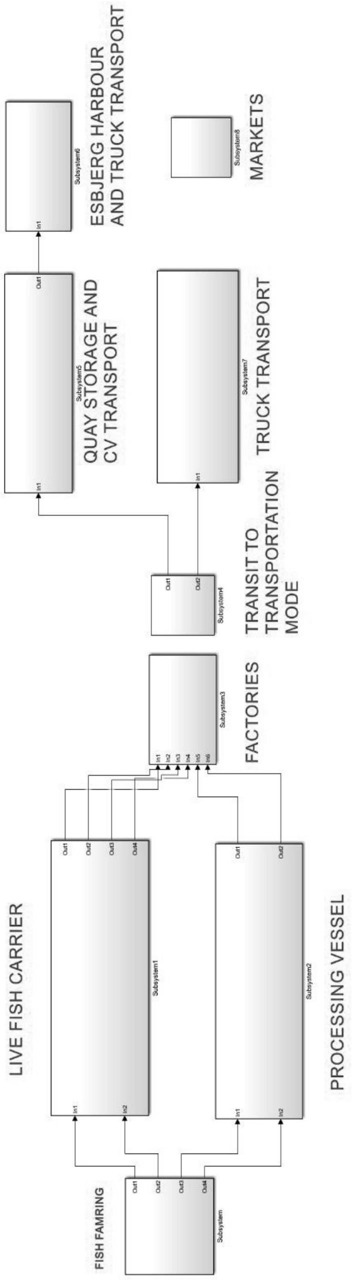


Figure 9.4: Simulink model divided in subsystems.

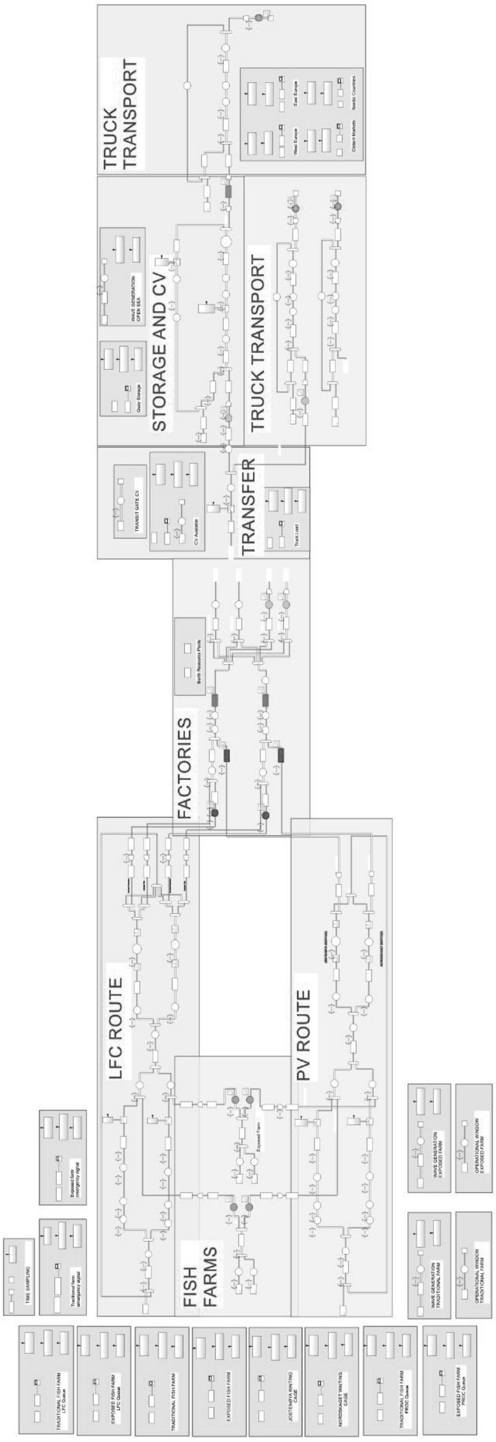


Figure 9.5: Complete Simulink model.

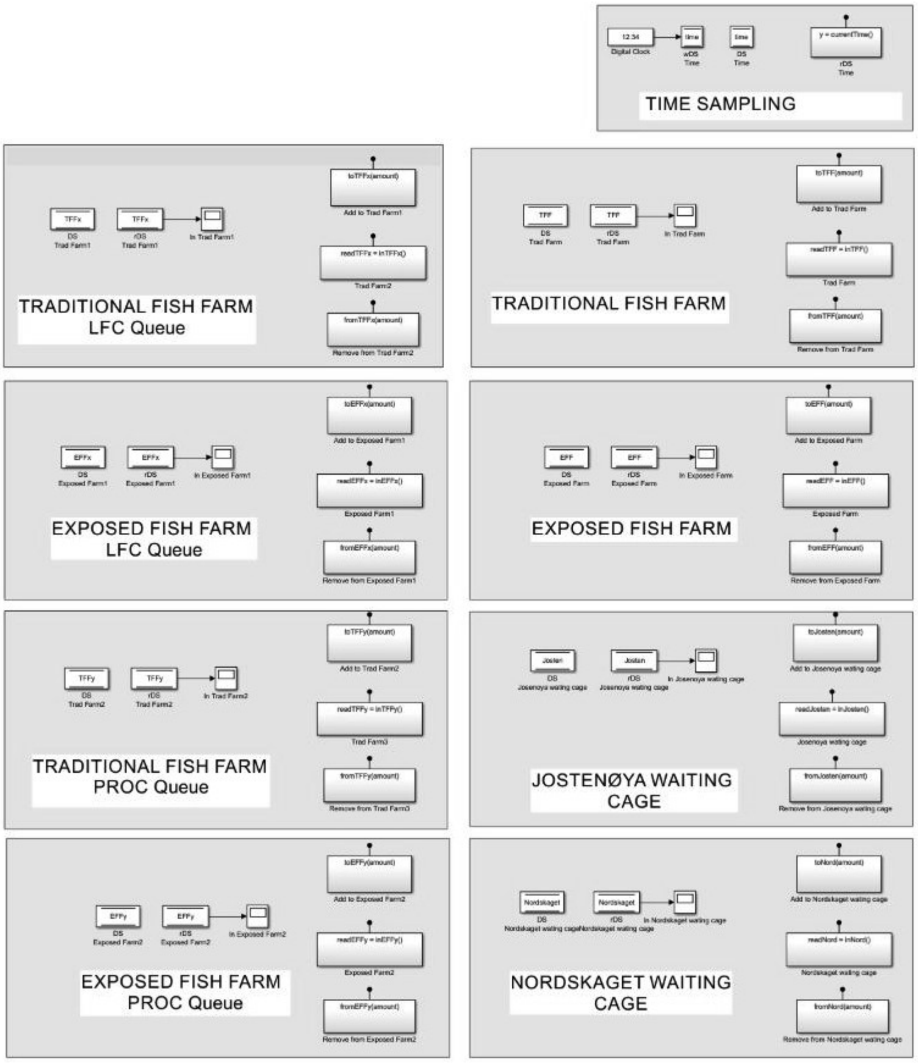


Figure 9.6: Global stores used for monitoring cages and vessel queues.

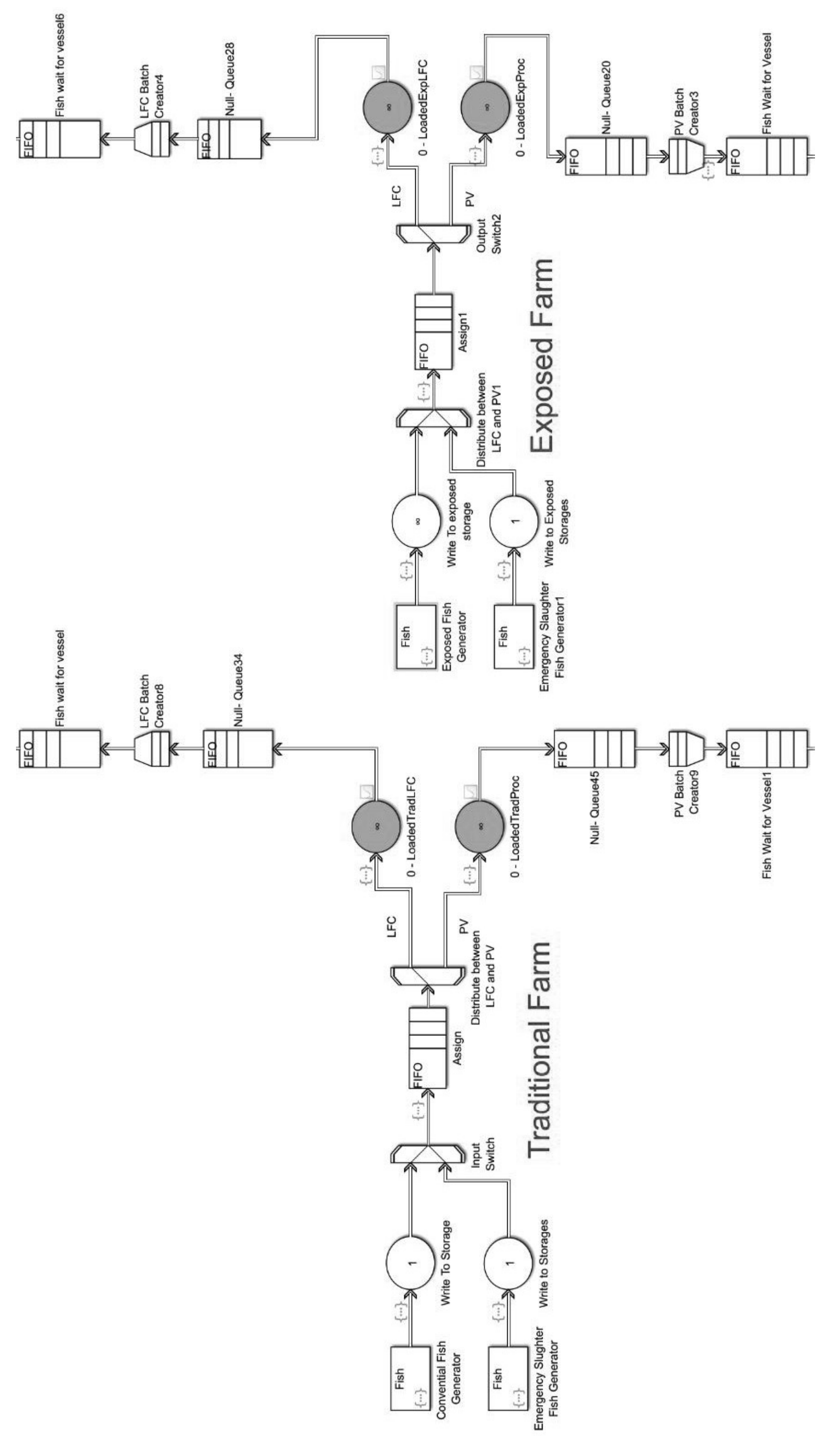


Figure 9.7: Fish farms.

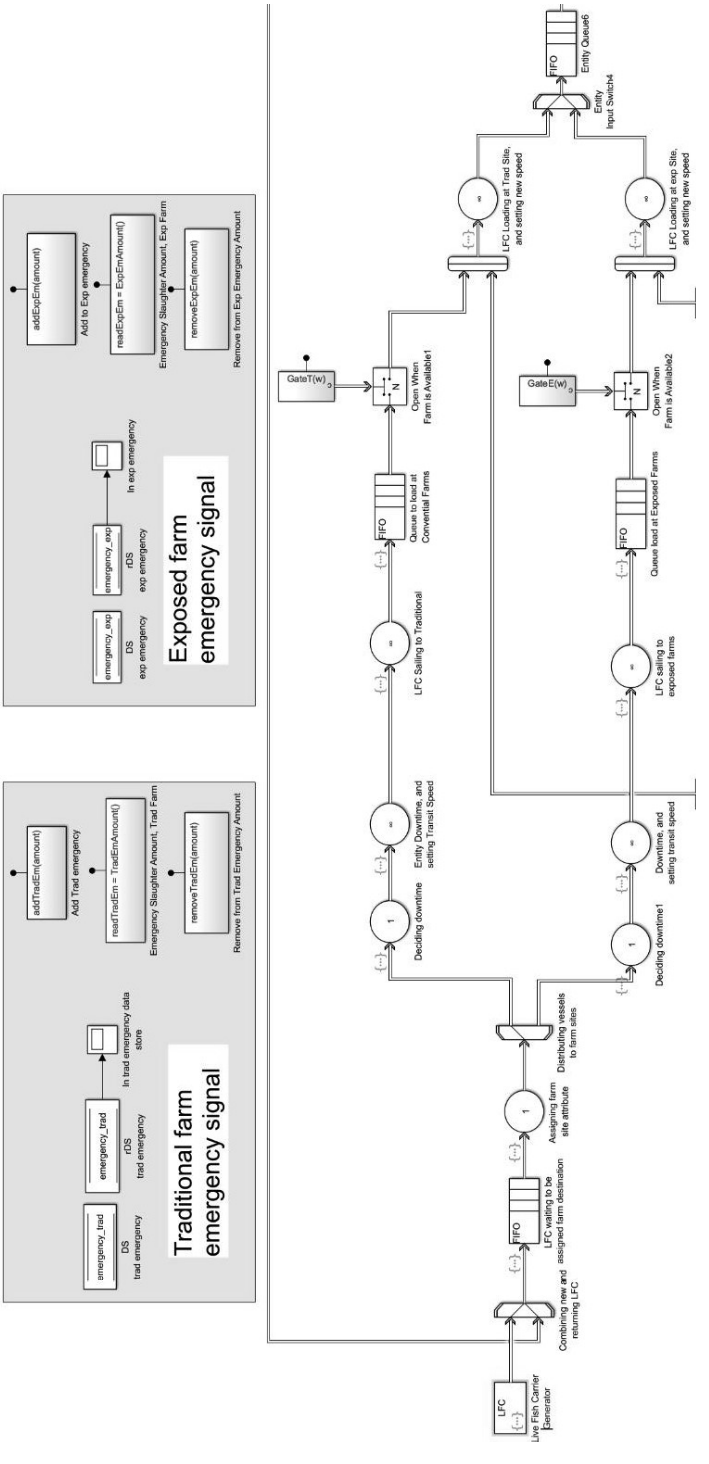


Figure 9.8: Live fish carriers loading at fish farms.

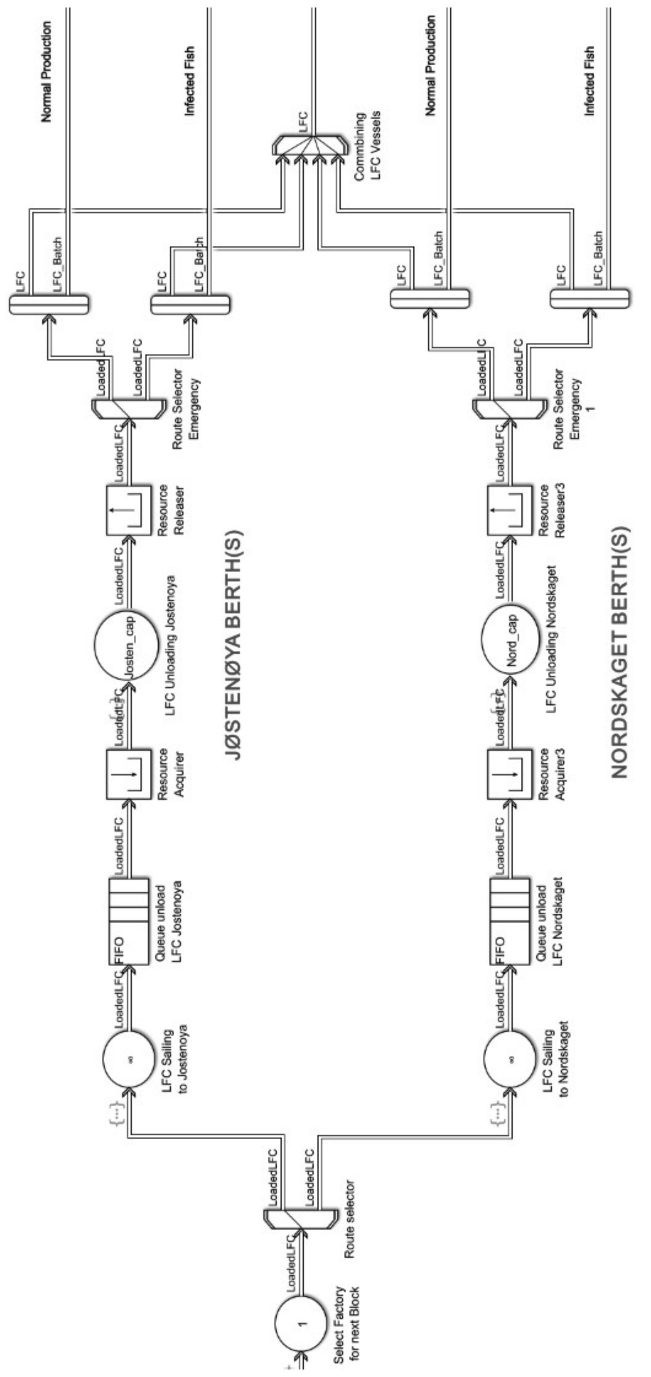


Figure 9.9: Live fish carrier unloading at factory, where infected and ”healthy” fish are sent in different routes after vessel unloading.

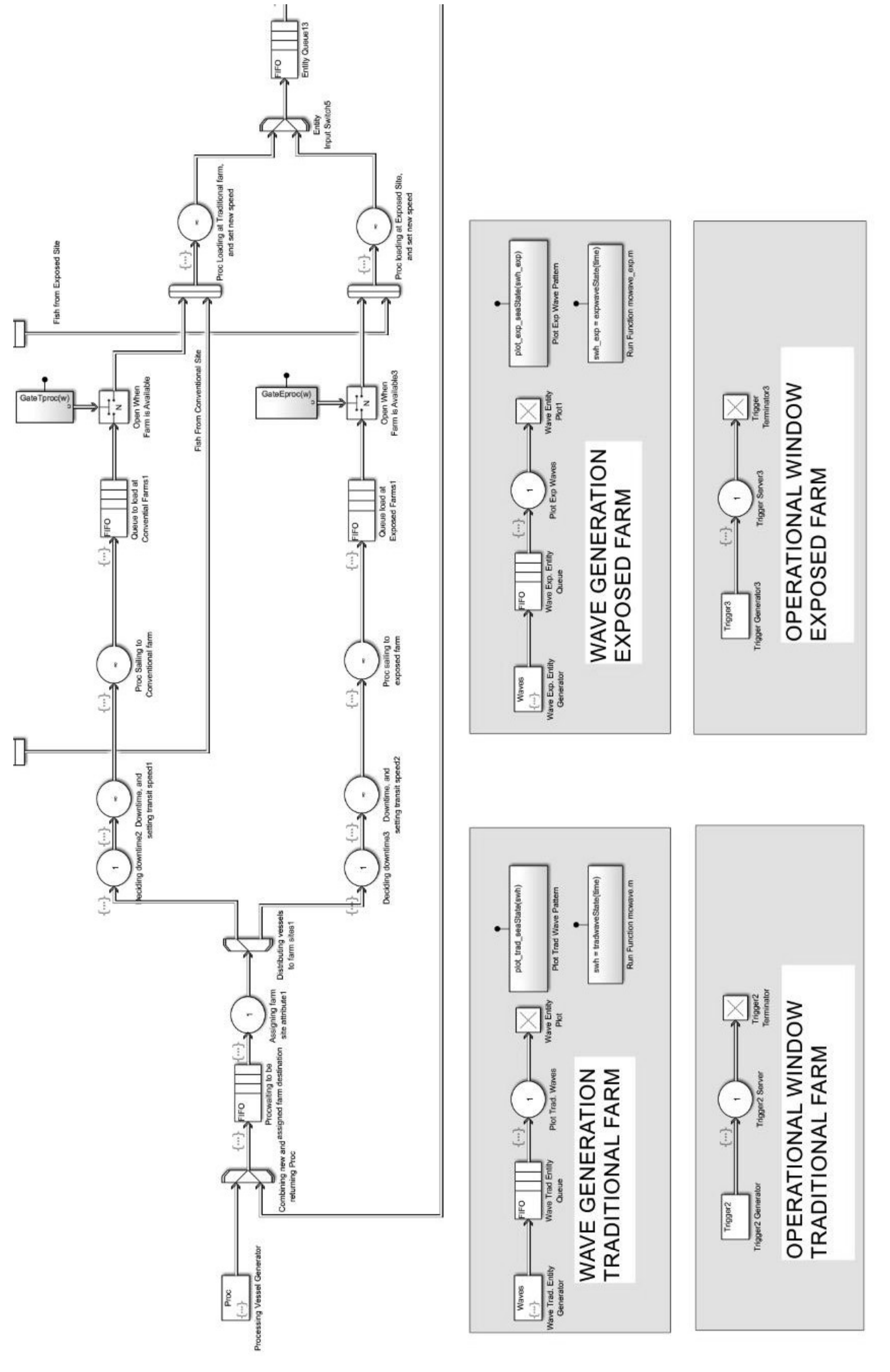


Figure 9.10: Processing vessels loading at the farms.

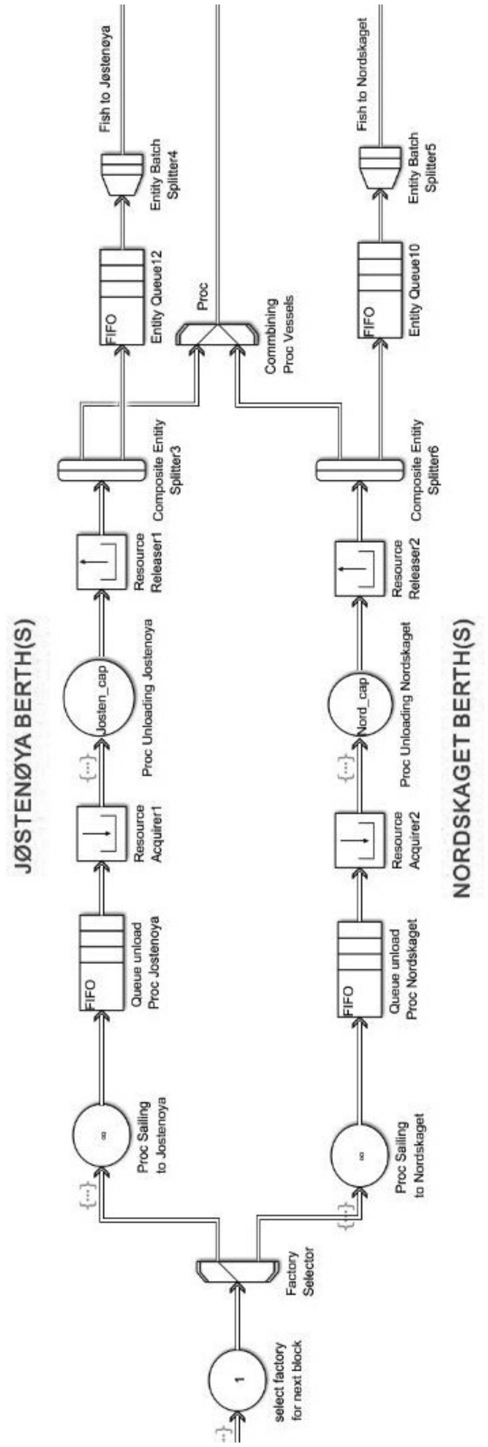


Figure 9.11: Processing vessels unloading at the factories.

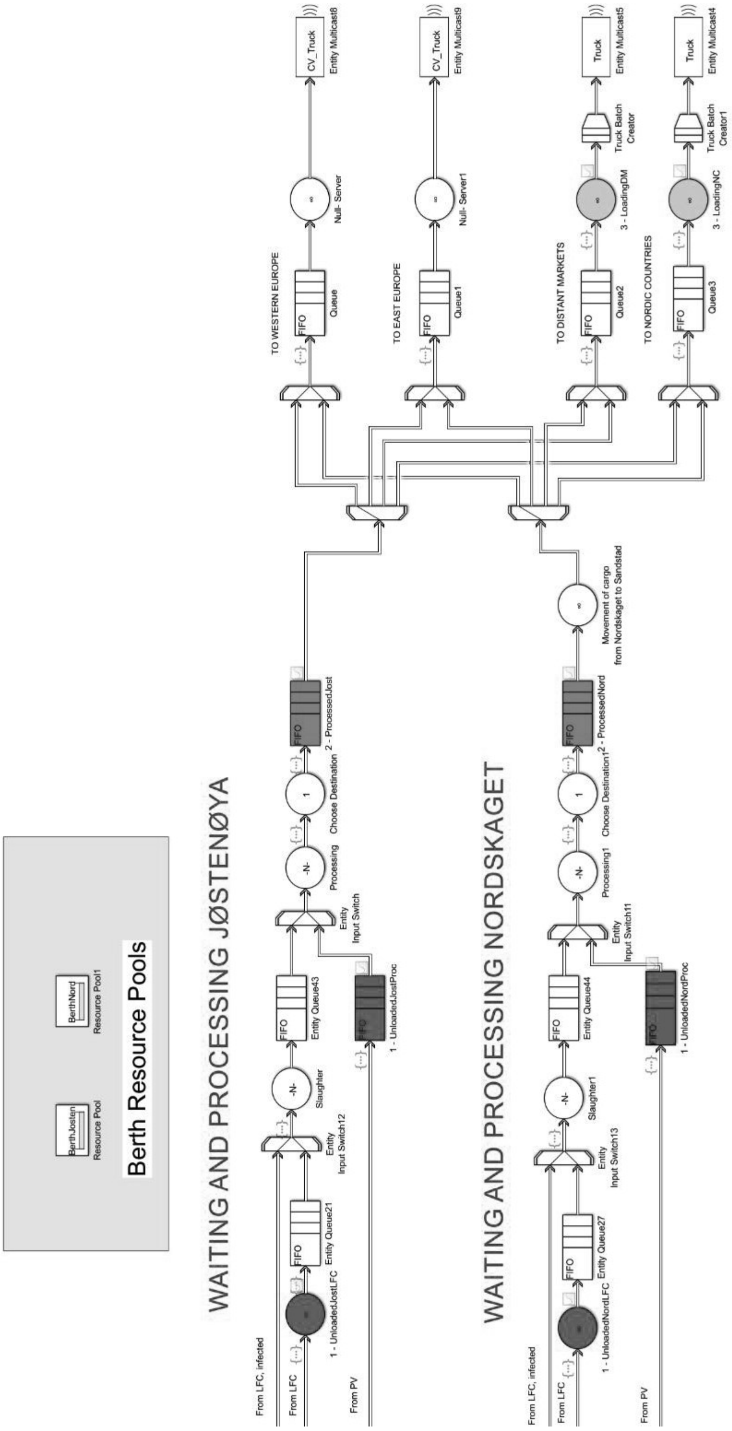


Figure 9.12: Jøstenøya and Nordskaget Factories.

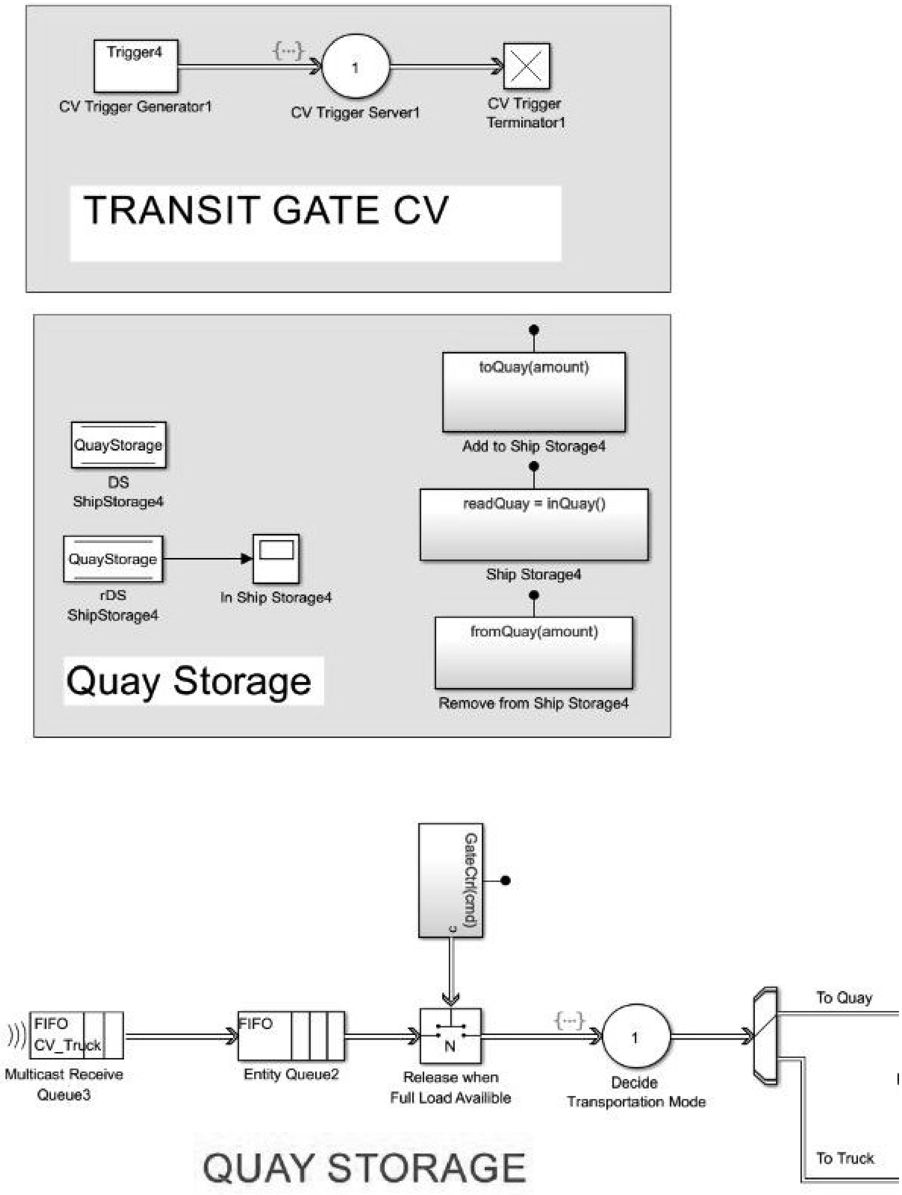


Figure 9.13: Transit to quay storage and truck loading.

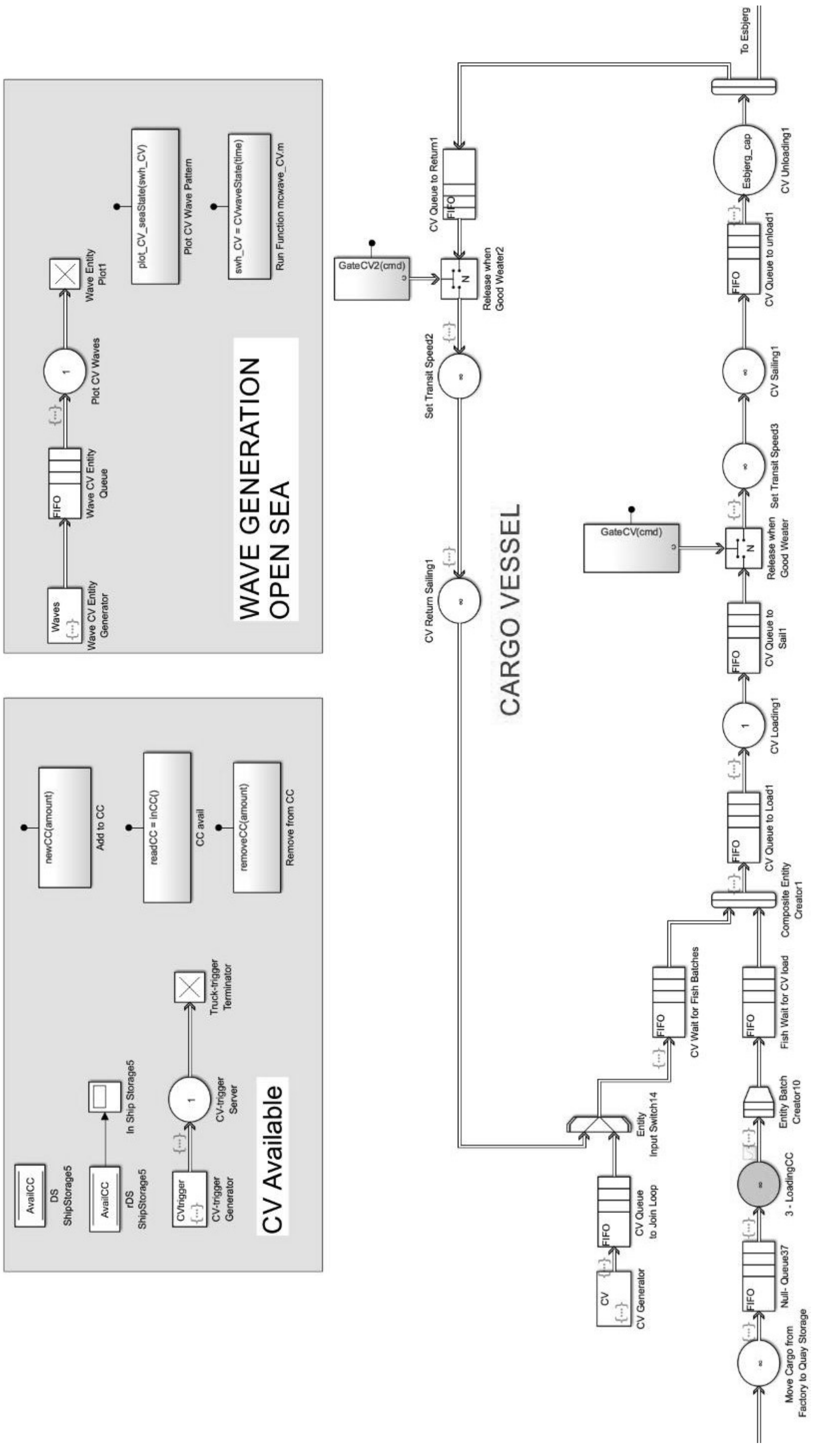


Figure 9.14: Cargo vessel route.

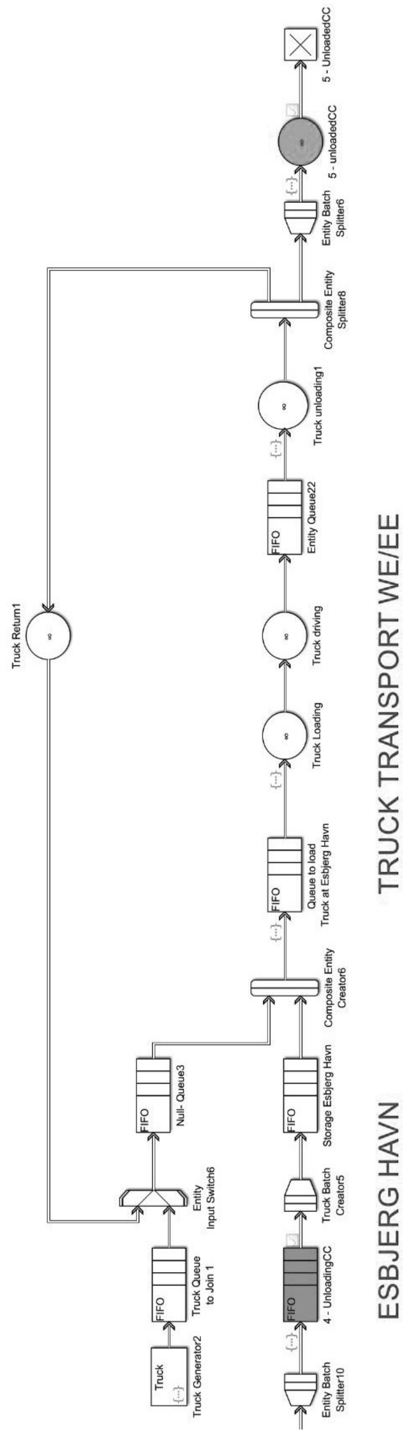


Figure 9.15: Esbjerg harbour and final truck transport to West- and East Europe.

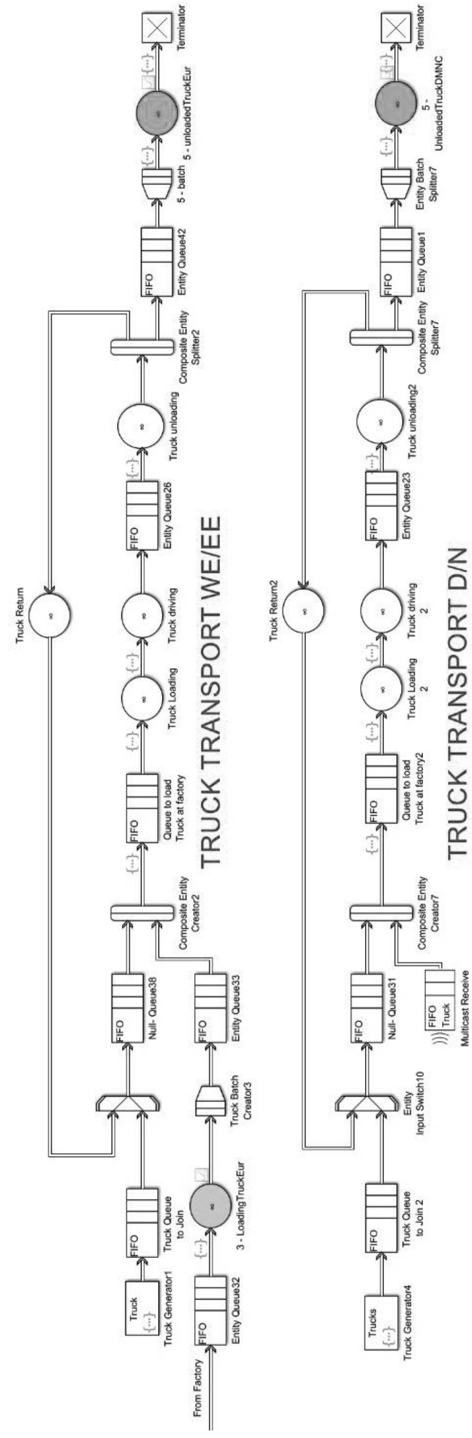


Figure 9.16: Truck transport to market.

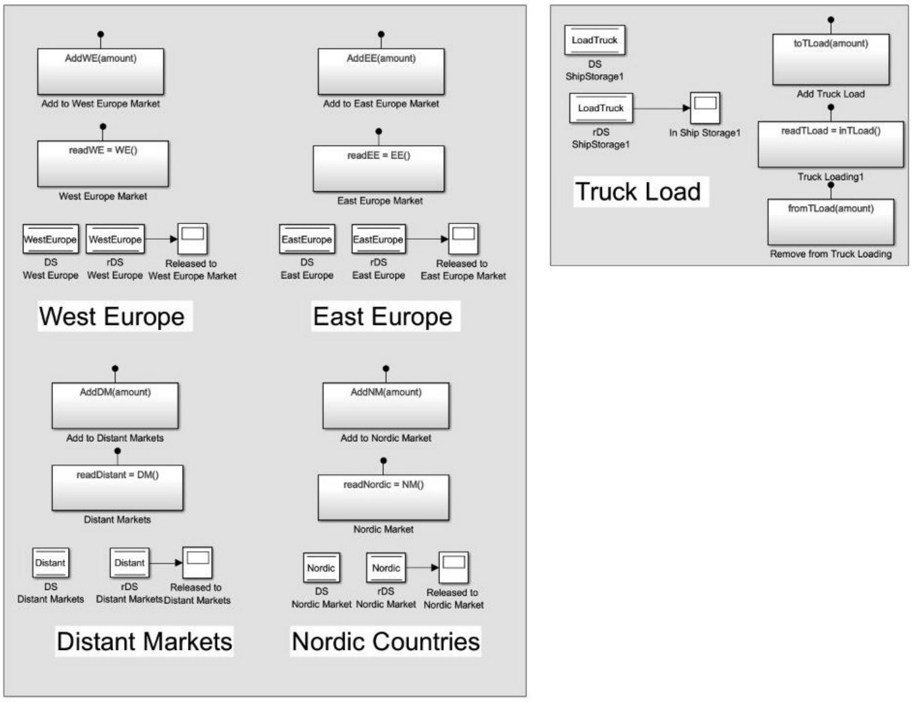


Figure 9.17: Global stores used for monitoring market delivery, and the number of trucks loading simultaneously at Jøstenøya.

D Results

D.1 Normal Operation

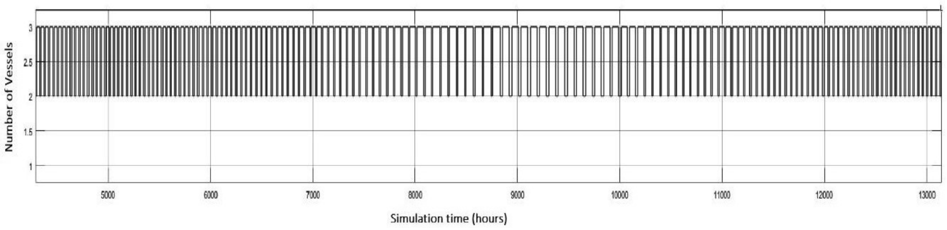


Figure 9.18: Live fish carriers waiting to load during normal operation.

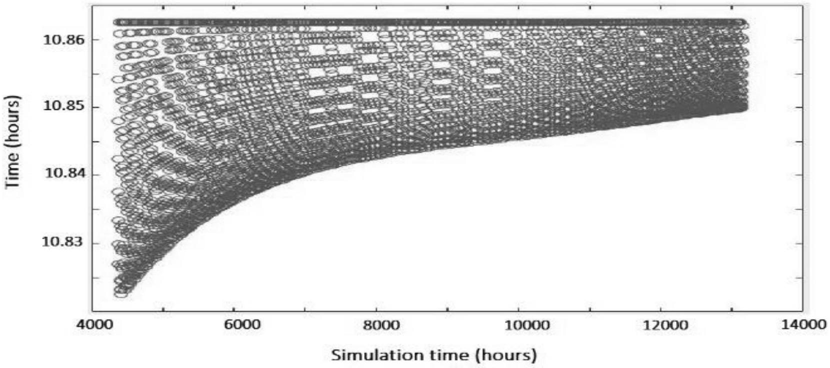


Figure 9.19: Average time spent waiting in waiting cage after the required 24 hours, and until slaughter.

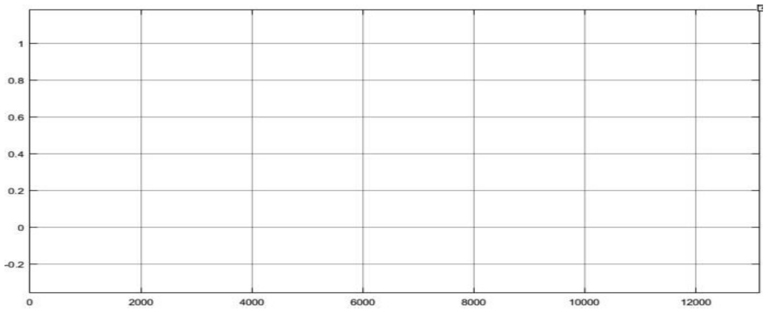


Figure 9.20: Simulink scope from quay storage block during simulation with no generated cargo vessels.

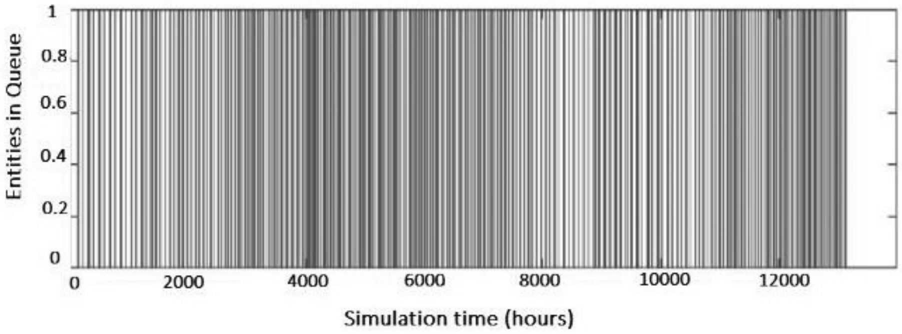


Figure 9.21: Fish entities in queue before the quay storage gate do not accumulate.

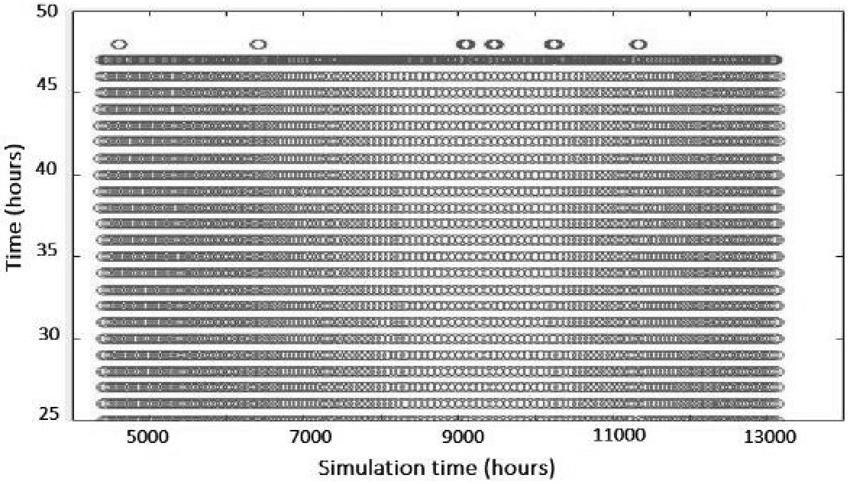


Figure 9.22: Time used for fish entities until tuck loading.

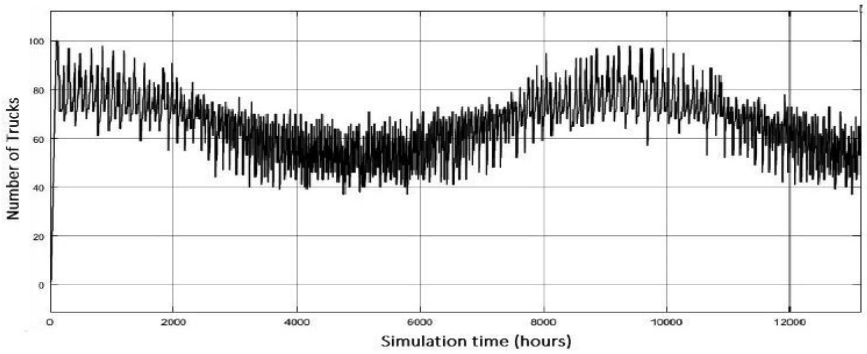
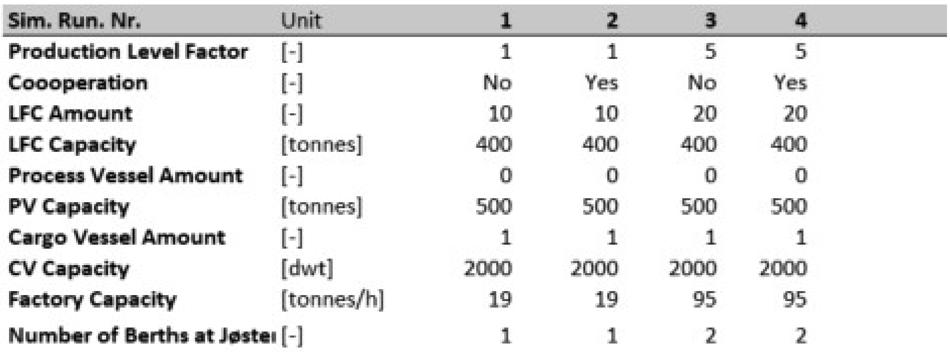


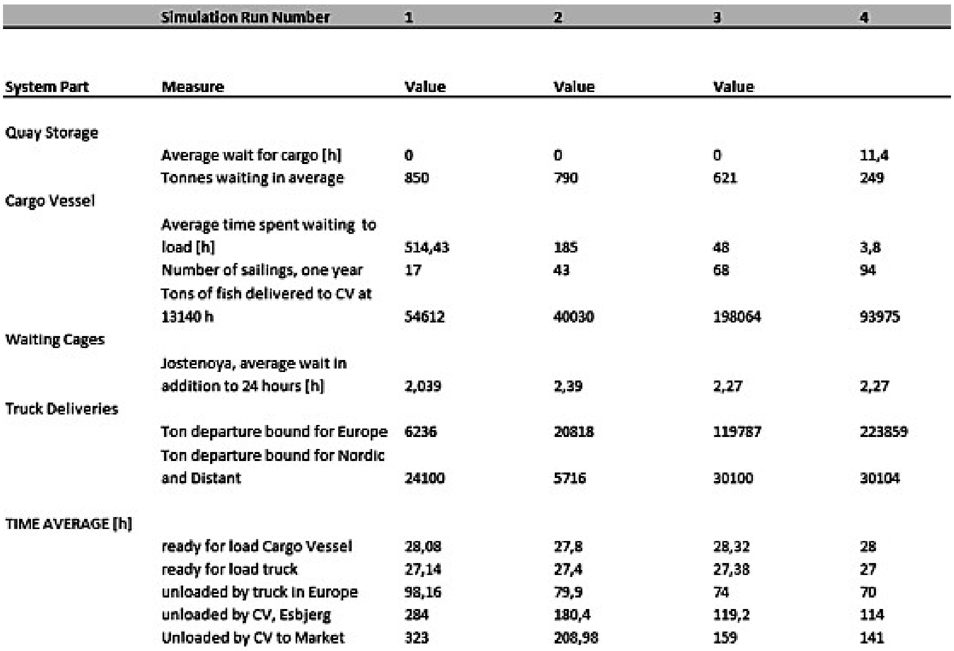
Figure 9.23: Number of trucks waiting to load for Europe, no cargo vessels in parallel.

D.2 Regional Cooperation



Sim. Run. Nr.	Unit	1	2	3	4
Production Level Factor	[-]	1	1	5	5
Coooperation	[-]	No	Yes	No	Yes
LFC Amount	[-]	10	10	20	20
LFC Capacity	[tonnes]	400	400	400	400
Process Vessel Amount	[-]	0	0	0	0
PV Capacity	[tonnes]	500	500	500	500
Cargo Vessel Amount	[-]	1	1	1	1
CV Capacity	[dwt]	2000	2000	2000	2000
Factory Capacity	[tonnes/h]	19	19	95	95
Number of Berths at Jøste	[-]	1	1	2	2

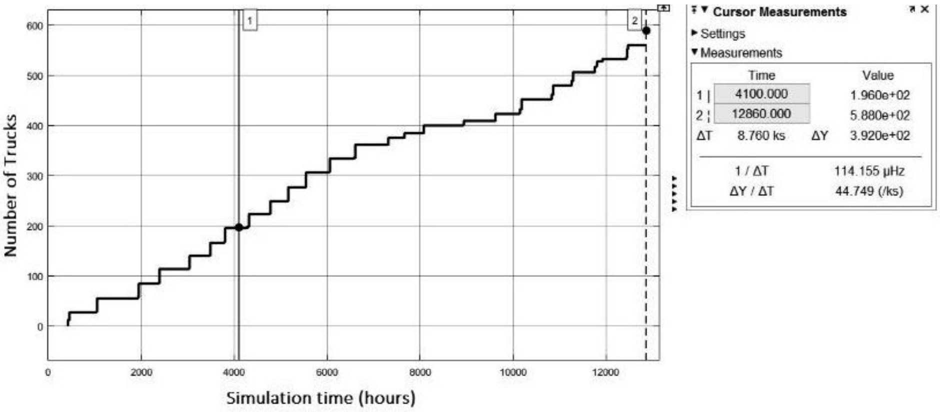
Figure 9.24: Input spreadsheet for simulations testing the impact of regional cooperation.



	Simulation Run Number	1	2	3	4
System Part	Measure	Value	Value	Value	
Quay Storage					
	Average wait for cargo [h]	0	0	0	11,4
	Tonnes waiting in average	850	790	621	249
Cargo Vessel					
	Average time spent waiting to load [h]	514,43	185	48	3,8
	Number of sailings, one year	17	43	68	94
	Tons of fish delivered to CV at 13140 h	54612	40030	198064	93975
Waiting Cages					
	Jostenoya, average wait in addition to 24 hours [h]	2,039	2,39	2,27	2,27
Truck Deliveries					
	Ton departure bound for Europe	6236	20818	119787	223859
	Ton departure bound for Nordic and Distant	24100	5716	30100	30104
TIME AVERAGE [h]					
	ready for load Cargo Vessel	28,08	27,8	28,32	28
	ready for load truck	27,14	27,4	27,38	27
	unloaded by truck in Europe	98,16	79,9	74	70
	unloaded by CV, Esbjerg	284	180,4	119,2	114
	Unloaded by CV to Market	323	208,98	159	141

Figure 9.25: Output spreadsheet for simulations testing the impact of regional cooperation.

D.3 Introduction of Cargo Vessel



Cursor Measurements

Measurements

	Time		Value
1	4100.000		1.960e+02
2	12860.000		5.880e+02
ΔT	8.760 ks	ΔY	3.920e+02

1 / ΔT	114.155 μHz
ΔY / ΔT	44.749 (/ks)

Figure 9.26: Annual truck departures for simulations for current production level, with one cargo vessel in parallel.

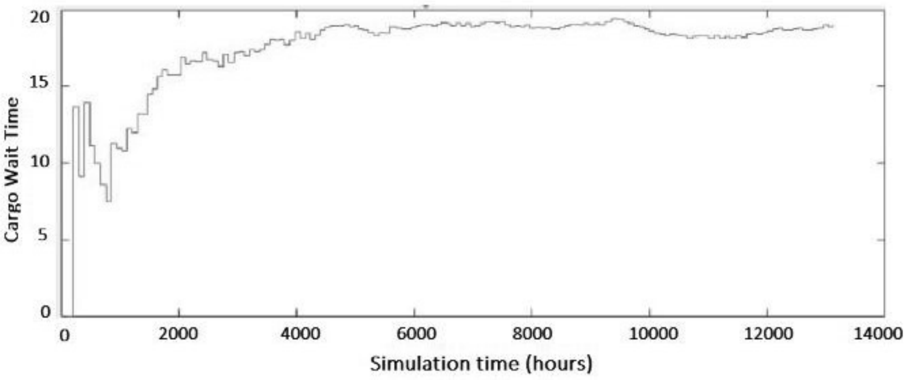


Figure 9.27: Average time a cargo entity waits to load cargo vessel during x2 production level.

Table 9.1: The number of trucks unloading at the same time at Jøstenøya during one year.

Production level	CC Amount	Max Amount	Mean Amount
5x	0	8	1.7
5x	1	6	1.4

D.4 Weather Influence

Table 9.2: Calculated downtime each month based on Hs limit, for cargo vessel during roundtrip transit

Hs limit	Unit	CV downtime				Annually
		Spring	Summer	Fall	Winter	
4 m	[hours]	183.4	25.5	190.3	376.7	2328.1
	[%]	25.4	3.5	26.4	52.3	0.3
6 m	[hours]	52.6	1.1	45.3	140.3	718.1
	[%]	7.3	0.2	6.3	19.5	0.1
9 m	[hours]	7.9	0	4.1	19.5	94.5
	[%]	1.1	0	0.6	10.5	2.7

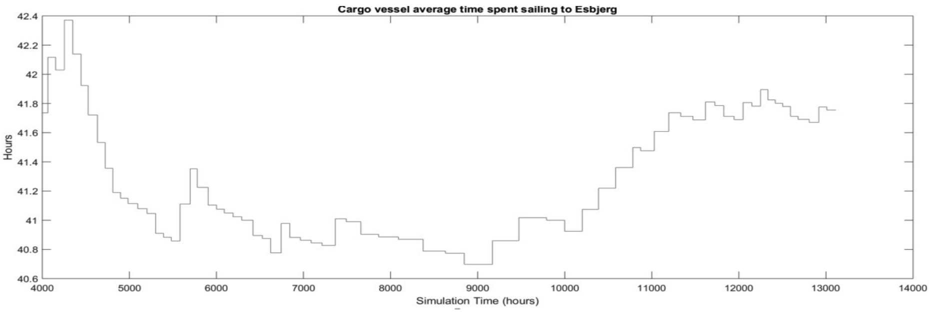
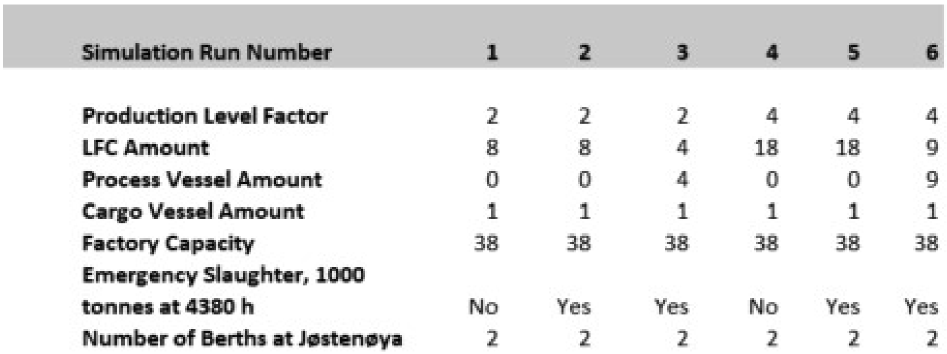


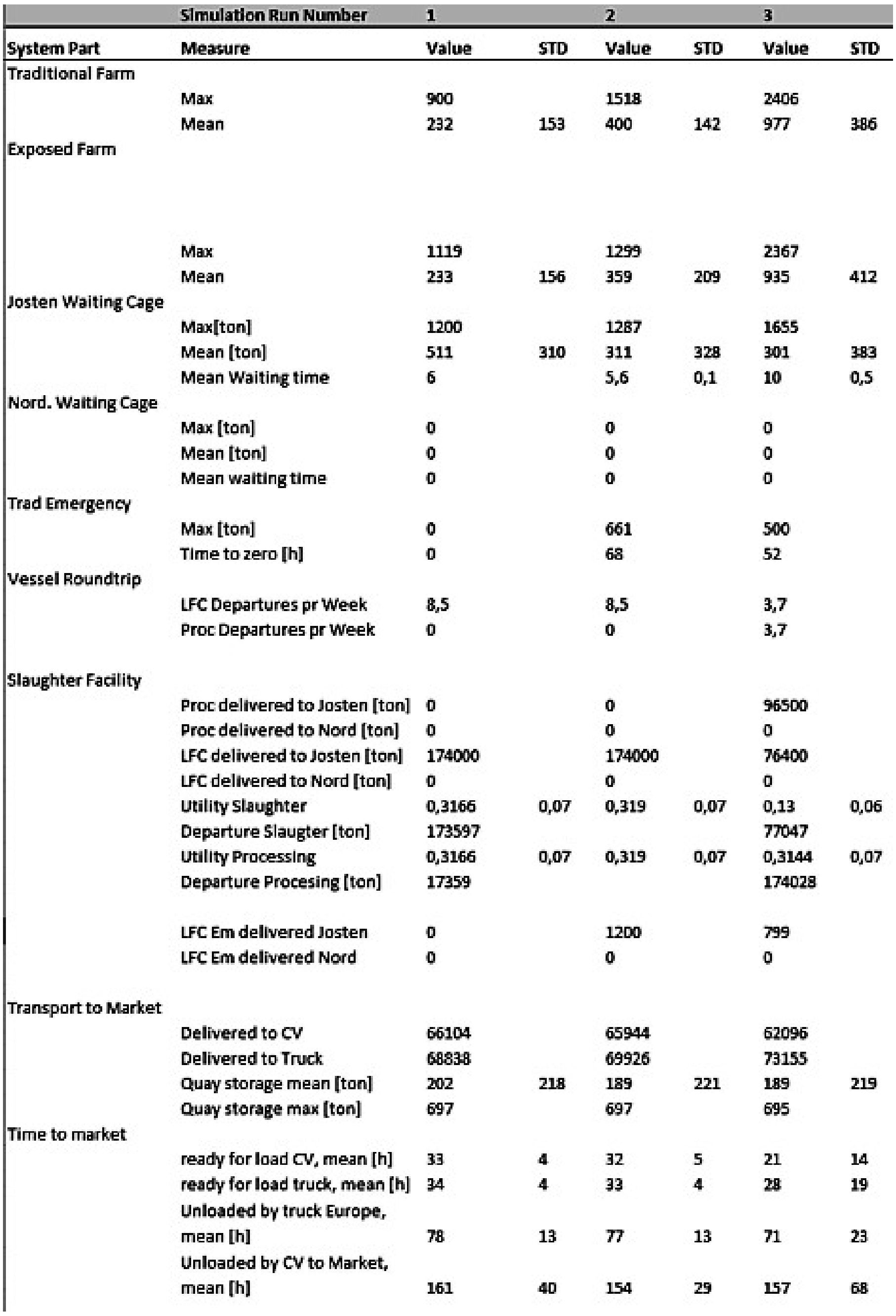
Figure 9.28: The average time spent sailing for cargo vessels between Jøstenøya and Esbjerg does not vary much throughout the year.

D.5 Emergency Slaughter



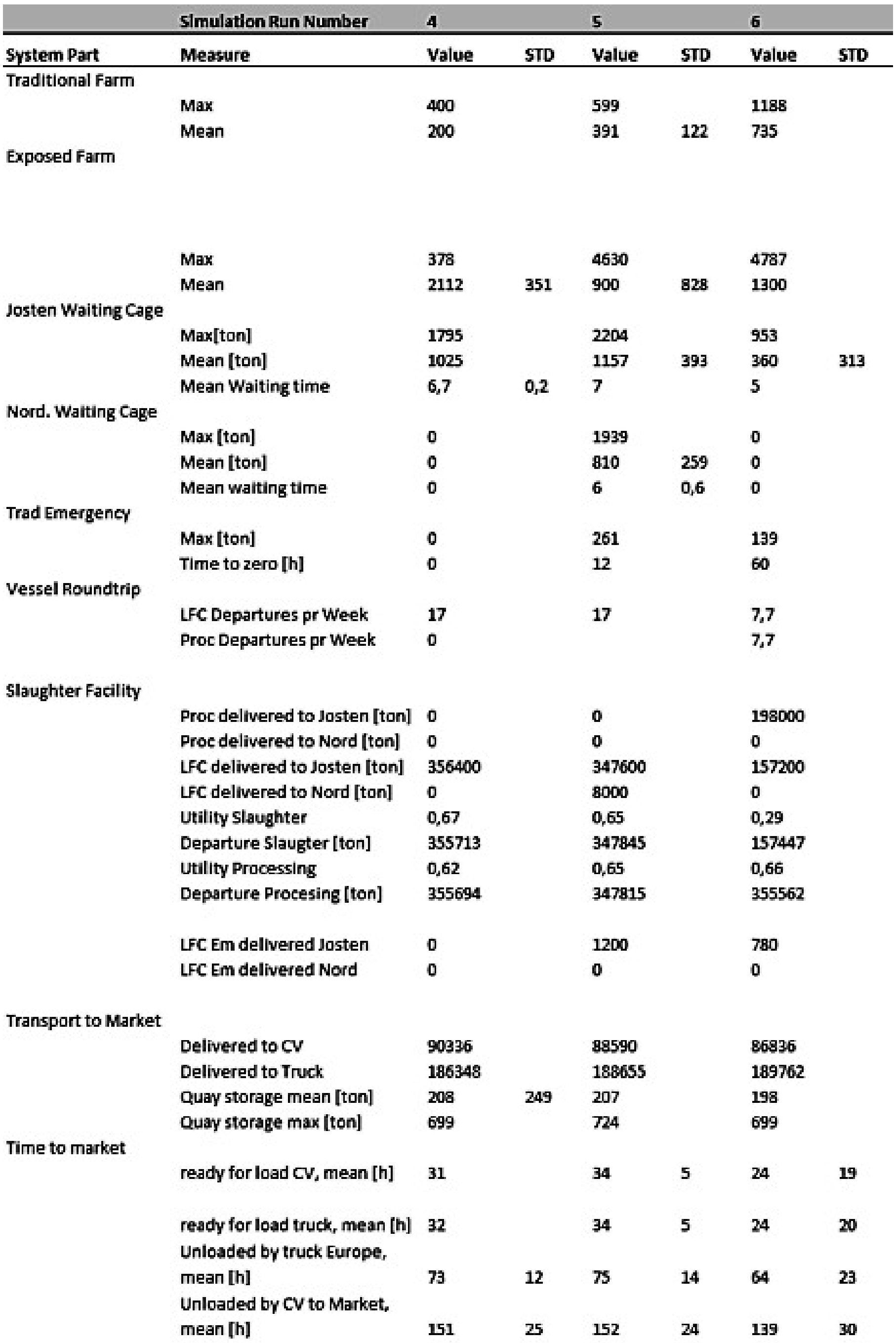
Simulation Run Number	1	2	3	4	5	6
Production Level Factor	2	2	2	4	4	4
LFC Amount	8	8	4	18	18	9
Process Vessel Amount	0	0	4	0	0	9
Cargo Vessel Amount	1	1	1	1	1	1
Factory Capacity	38	38	38	38	38	38
Emergency Slaughter, 1000 tonnes at 4380 h	No	Yes	Yes	No	Yes	Yes
Number of Berths at Jøstenøya	2	2	2	2	2	2

Figure 9.29: Input spreadsheet for simulations testing the impact of emergency slaughter



	Simulation Run Number	1		2		3	
System Part	Measure	Value	STD	Value	STD	Value	STD
Traditional Farm							
	Max	900		1518		2406	
	Mean	232	153	400	142	977	386
Exposed Farm							
	Max	1119		1299		2367	
	Mean	233	156	359	209	935	412
Josten Waiting Cage							
	Max[ton]	1200		1287		1655	
	Mean [ton]	511	310	311	328	301	383
	Mean Waiting time	6		5,6	0,1	10	0,5
Nord. Waiting Cage							
	Max [ton]	0		0		0	
	Mean [ton]	0		0		0	
	Mean waiting time	0		0		0	
Trad Emergency							
	Max [ton]	0		661		500	
	Time to zero [h]	0		68		52	
Vessel Roundtrip							
	LFC Departures pr Week	8,5		8,5		3,7	
	Proc Departures pr Week	0		0		3,7	
Slaughter Facility							
	Proc delivered to Josten [ton]	0		0		96500	
	Proc delivered to Nord [ton]	0		0		0	
	LFC delivered to Josten [ton]	174000		174000		76400	
	LFC delivered to Nord [ton]	0		0		0	
	Utility Slaughter	0,3166	0,07	0,319	0,07	0,13	0,06
	Departure Slaugter [ton]	173597				77047	
	Utility Processing	0,3166	0,07	0,319	0,07	0,3144	0,07
	Departure Procesing [ton]	17359				174028	
	LFC Em delivered Josten	0		1200		799	
	LFC Em delivered Nord	0		0		0	
Transport to Market							
	Delivered to CV	66104		65944		62096	
	Delivered to Truck	68838		69926		73155	
	Quay storage mean [ton]	202	218	189	221	189	219
	Quay storage max [ton]	697		697		695	
Time to market							
	ready for load CV, mean [h]	33	4	32	5	21	14
	ready for load truck, mean [h]	34	4	33	4	28	19
	Unloaded by truck Europe, mean [h]	78	13	77	13	71	23
	Unloaded by CV to Market, mean [h]	161	40	154	29	157	68

Figure 9.30: Output spreadsheet for simulations testing the impact of emergency slaughter, vol. 1



	Simulation Run Number	4		5		6	
System Part	Measure	Value	STD	Value	STD	Value	STD
Traditional Farm							
	Max	400		599		1188	
	Mean	200		391	122	735	
Exposed Farm							
	Max	378		4630		4787	
	Mean	2112	351	900	828	1300	
Josten Waiting Cage							
	Max[ton]	1795		2204		953	
	Mean [ton]	1025		1157	393	360	313
	Mean Waiting time	6,7	0,2	7		5	
Nord. Waiting Cage							
	Max [ton]	0		1939		0	
	Mean [ton]	0		810	259	0	
	Mean waiting time	0		6	0,6	0	
Trad Emergency							
	Max [ton]	0		261		139	
	Time to zero [h]	0		12		60	
Vessel Roundtrip							
	LFC Departures pr Week	17		17		7,7	
	Proc Departures pr Week	0				7,7	
Slaughter Facility							
	Proc delivered to Josten [ton]	0		0		198000	
	Proc delivered to Nord [ton]	0		0		0	
	LFC delivered to Josten [ton]	356400		347600		157200	
	LFC delivered to Nord [ton]	0		8000		0	
	Utility Slaughter	0,67		0,65		0,29	
	Departure Slaugter [ton]	355713		347845		157447	
	Utility Processing	0,62		0,65		0,66	
	Departure Procesing [ton]	355694		347815		355562	
	LFC Em delivered Josten	0		1200		780	
	LFC Em delivered Nord	0		0		0	
Transport to Market							
	Delivered to CV	90336		88590		86836	
	Delivered to Truck	186348		188655		189762	
	Quay storage mean [ton]	208	249	207		198	
	Quay storage max [ton]	699		724		699	
Time to market							
	ready for load CV, mean [h]	31		34	5	24	19
	ready for load truck, mean [h]	32		34	5	24	20
	Unloaded by truck Europe, mean [h]	73	12	75	14	64	23
	Unloaded by CV to Market, mean [h]	151	25	152	24	139	30

Figure 9.31: Output spreadsheet for simulations testing the impact of emergency slaughter, vol. 2

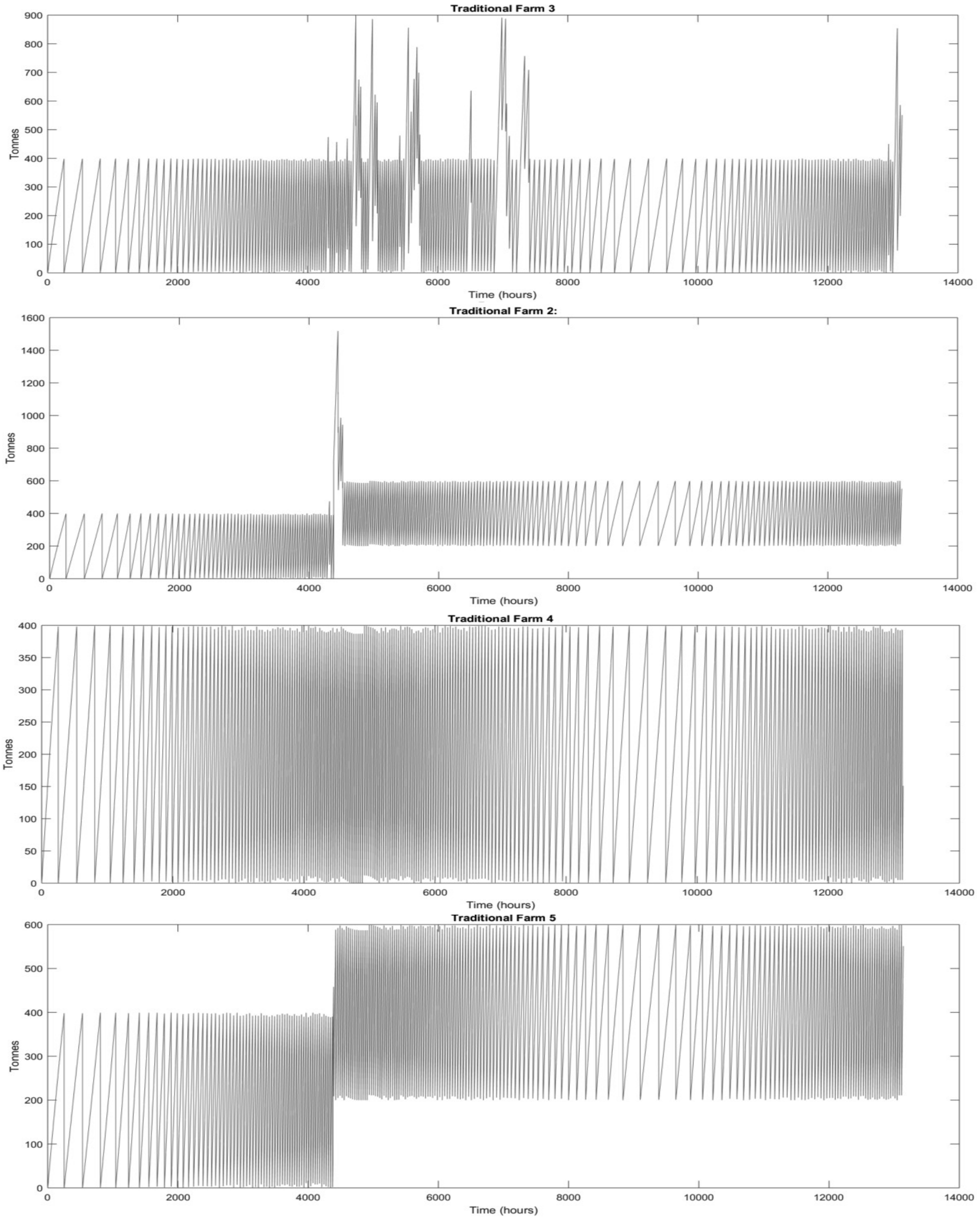


Figure 9.32: Traditional farm volumes during emergency slaughter simulations.

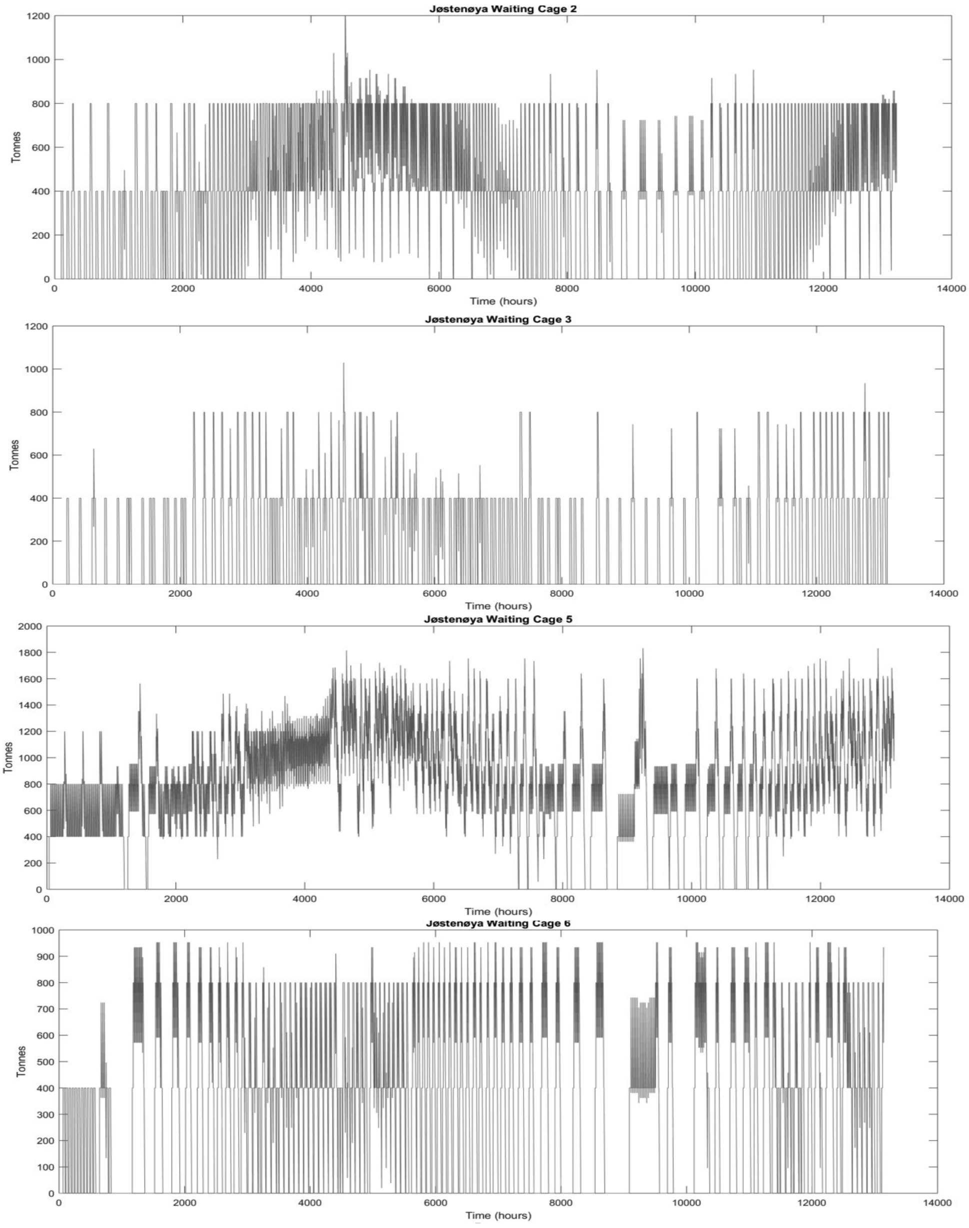
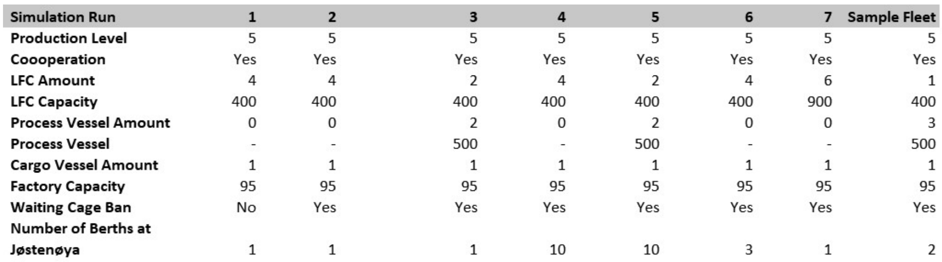


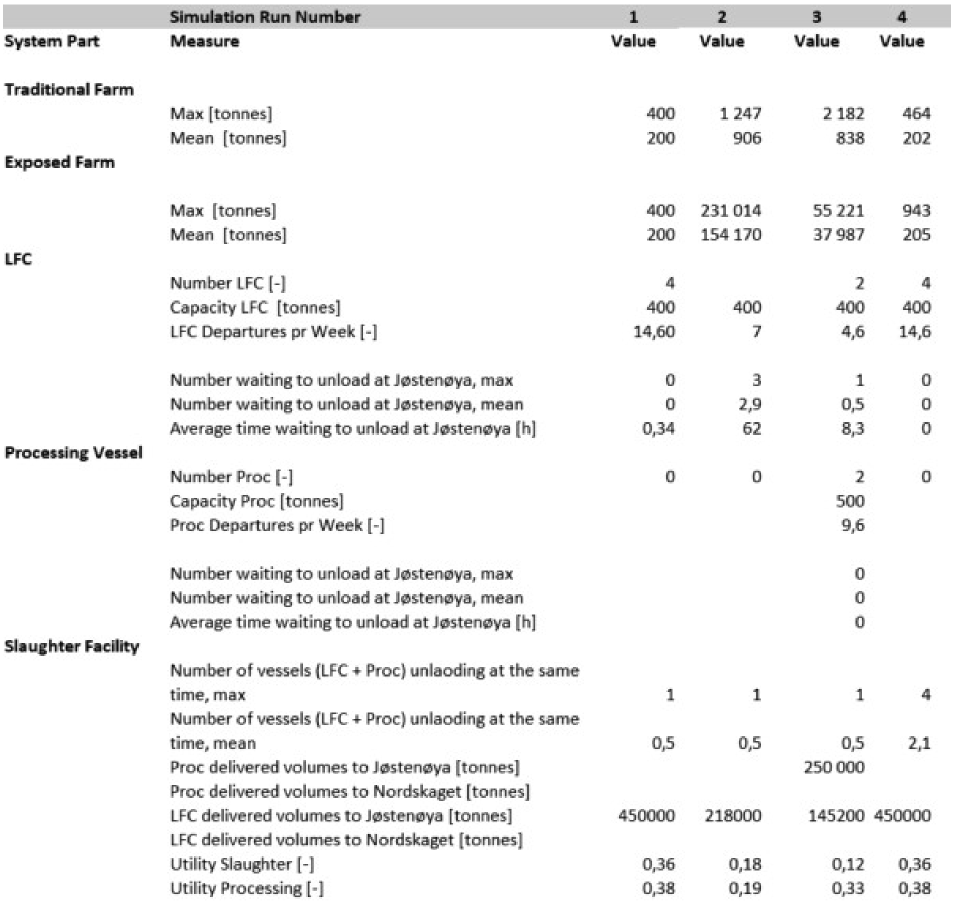
Figure 9.33: Jøstenøya waiting cage volumes during emergency slaughter simulations.

D.6 Waiting Cage Prohibition



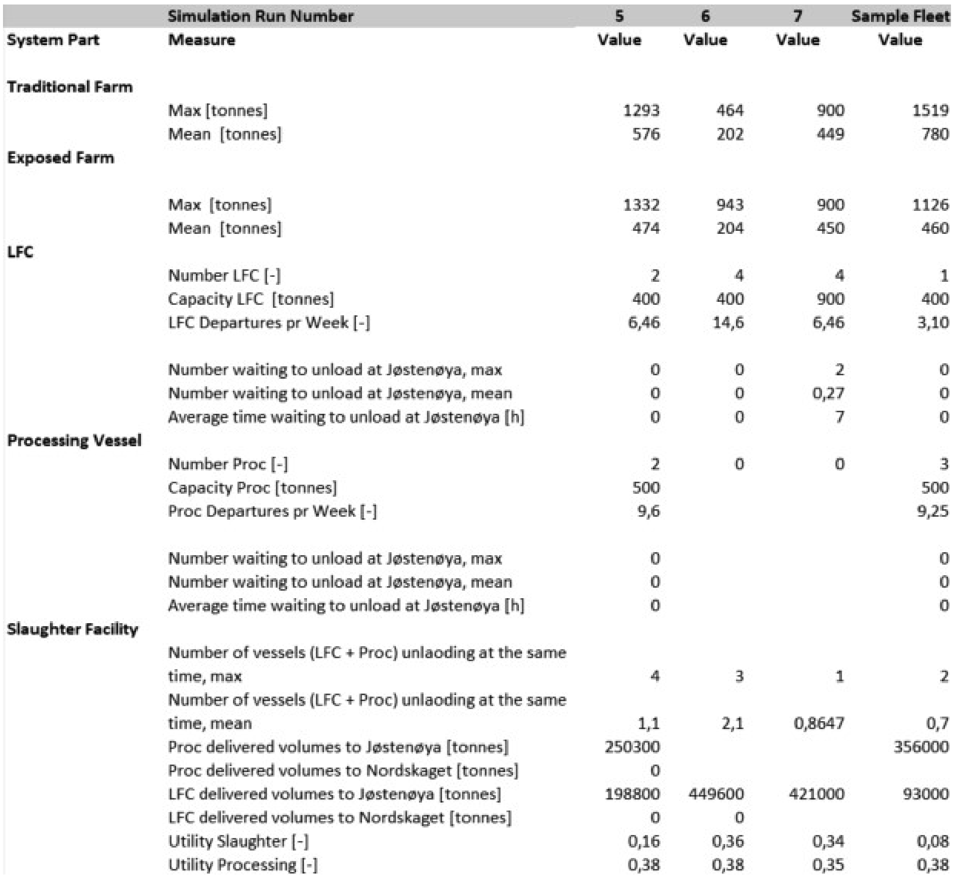
Simulation Run	1	2	3	4	5	6	7	Sample Fleet
Production Level	5	5	5	5	5	5	5	5
Coooperation	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
LFC Amount	4	4	2	4	2	4	6	1
LFC Capacity	400	400	400	400	400	400	900	400
Process Vessel Amount	0	0	2	0	2	0	0	3
Process Vessel	-	-	500	-	500	-	-	500
Cargo Vessel Amount	1	1	1	1	1	1	1	1
Factory Capacity	95	95	95	95	95	95	95	95
Waiting Cage Ban	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Number of Berths at Jøstenøya	1	1	1	10	10	3	1	2

Figure 9.34: Input spreadsheet for simulations testing the impact of open waiting cage ban, vol. 2



	Simulation Run Number	1	2	3	4
System Part	**Measure**	**Value**	**Value**	**Value**	**Value**
Traditional Farm					
	Max [tonnes]	400	1 247	2 182	464
	Mean [tonnes]	200	906	838	202
Exposed Farm					
	Max [tonnes]	400	231 014	55 221	943
	Mean [tonnes]	200	154 170	37 987	205
LFC					
	Number LFC [-]	4		2	4
	Capacity LFC [tonnes]	400	400	400	400
	LFC Departures pr Week [-]	14,60	7	4,6	14,6
	Number waiting to unload at Jøstenøya, max	0	3	1	0
	Number waiting to unload at Jøstenøya, mean	0	2,9	0,5	0
	Average time waiting to unload at Jøstenøya [h]	0,34	62	8,3	0
Processing Vessel					
	Number Proc [-]	0	0	2	0
	Capacity Proc [tonnes]			500	
	Proc Departures pr Week [-]			9,6	
	Number waiting to unload at Jøstenøya, max			0	
	Number waiting to unload at Jøstenøya, mean			0	
	Average time waiting to unload at Jøstenøya [h]			0	
Slaughter Facility					
	Number of vessels (LFC + Proc) unlaoding at the same time, max	1	1	1	4
	Number of vessels (LFC + Proc) unlaoding at the same time, mean	0,5	0,5	0,5	2,1
	Proc delivered volumes to Jøstenøya [tonnes]			250 000	
	Proc delivered volumes to Nordskaget [tonnes]				
	LFC delivered volumes to Jøstenøya [tonnes]	450000	218000	145200	450000
	LFC delivered volumes to Nordskaget [tonnes]				
	Utility Slaughter [-]	0,36	0,18	0,12	0,36
	Utility Processing [-]	0,38	0,19	0,33	0,38

Figure 9.35: Output spreadsheet for simulations testing the impact of open waiting cage ban, vol. 1



	Simulation Run Number	5	6	7	Sample Fleet
System Part	**Measure**	**Value**	**Value**	**Value**	**Value**
Traditional Farm					
	Max [tonnes]	1293	464	900	1519
	Mean [tonnes]	576	202	449	780
Exposed Farm					
	Max [tonnes]	1332	943	900	1126
	Mean [tonnes]	474	204	450	460
LFC					
	Number LFC [-]	2	4	4	1
	Capacity LFC [tonnes]	400	400	900	400
	LFC Departures pr Week [-]	6,46	14,6	6,46	3,10
	Number waiting to unload at Jøstenøya, max	0	0	2	0
	Number waiting to unload at Jøstenøya, mean	0	0	0,27	0
	Average time waiting to unload at Jøstenøya [h]	0	0	7	0
Processing Vessel					
	Number Proc [-]	2	0	0	3
	Capacity Proc [tonnes]	500			500
	Proc Departures pr Week [-]	9,6			9,25
	Number waiting to unload at Jøstenøya, max	0			0
	Number waiting to unload at Jøstenøya, mean	0			0
	Average time waiting to unload at Jøstenøya [h]	0			0
Slaughter Facility					
	Number of vessels (LFC + Proc) unlaoding at the same time, max	4	3	1	2
	Number of vessels (LFC + Proc) unlaoding at the same time, mean	1,1	2,1	0,8647	0,7
	Proc delivered volumes to Jøstenøya [tonnes]	250300			356000
	Proc delivered volumes to Nordskaget [tonnes]	0			
	LFC delivered volumes to Jøstenøya [tonnes]	198800	449600	421000	93000
	LFC delivered volumes to Nordskaget [tonnes]	0	0		
	Utility Slaughter [-]	0,16	0,36	0,34	0,08
	Utility Processing [-]	0,38	0,38	0,35	0,38

Figure 9.36: Output spreadsheet for simulations testing the impact of open waiting cage ban, vol. 2

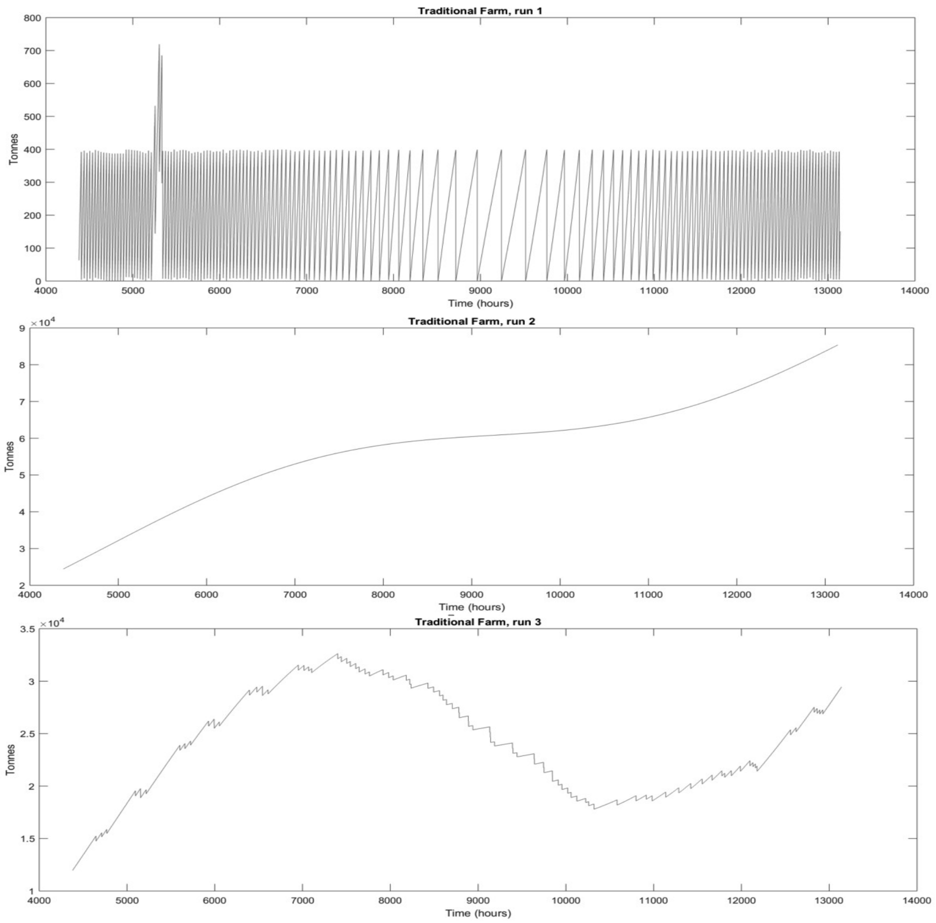


Figure 9.37: Volumes pending in the traditional farms during simulation 1-3, testing impact of waiting cage prohibition. Simulation two and three shows accumulation of fish in the farms.

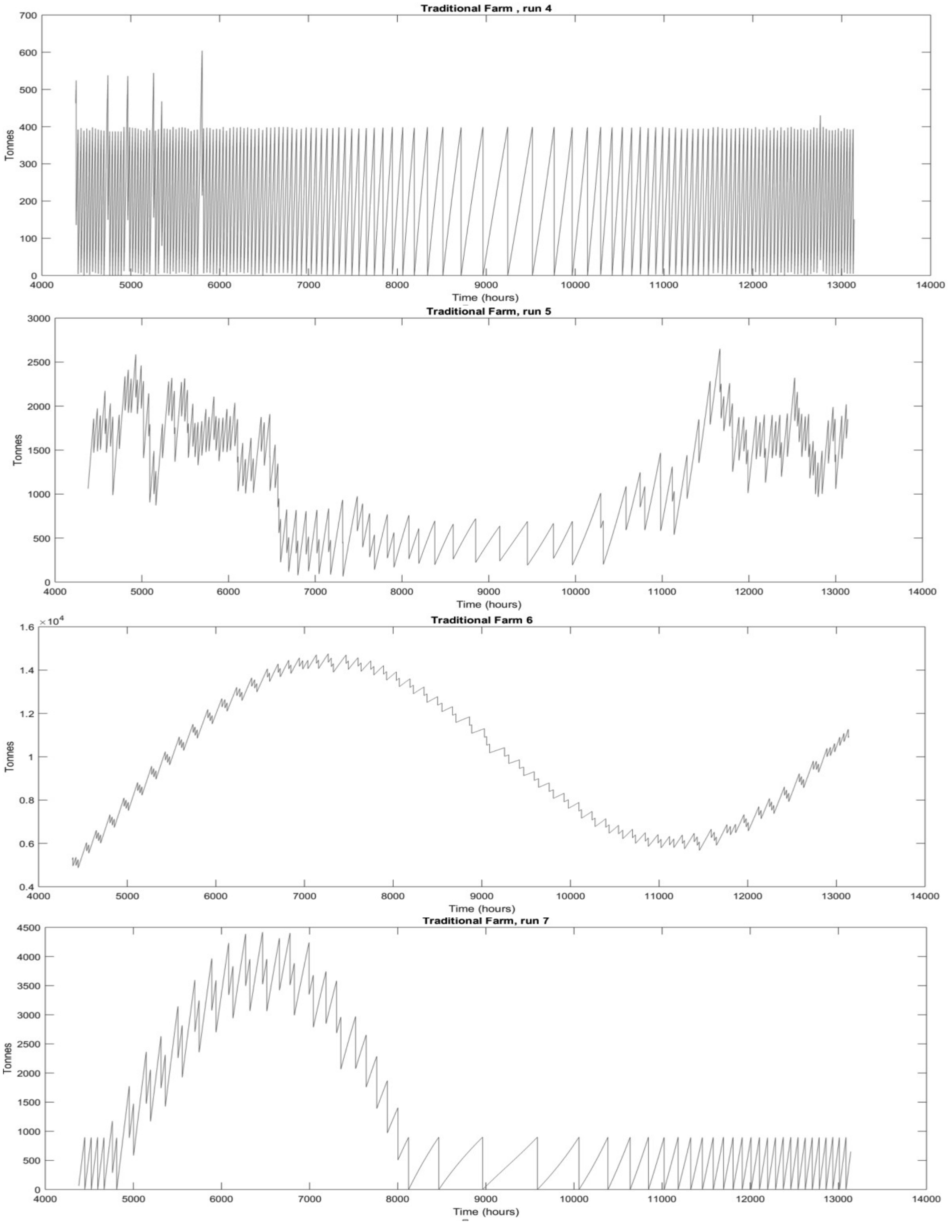


Figure 9.38: Volumes pending in the traditional farms during simulation 4-7, testing impact of waiting cage prohibition. Accumulation of fish higher for simulations imposing prohibition, and during summer months.

D.7 Benchmarking Fleet

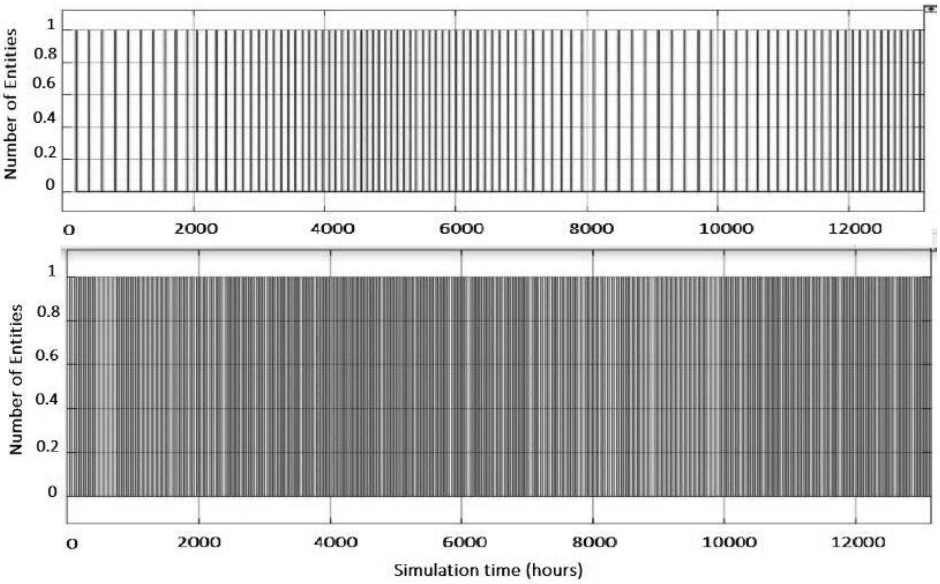


Figure 9.39: Shiploads of fish waiting to be picked up at the traditional farms (top) and for the exposed farm (below).

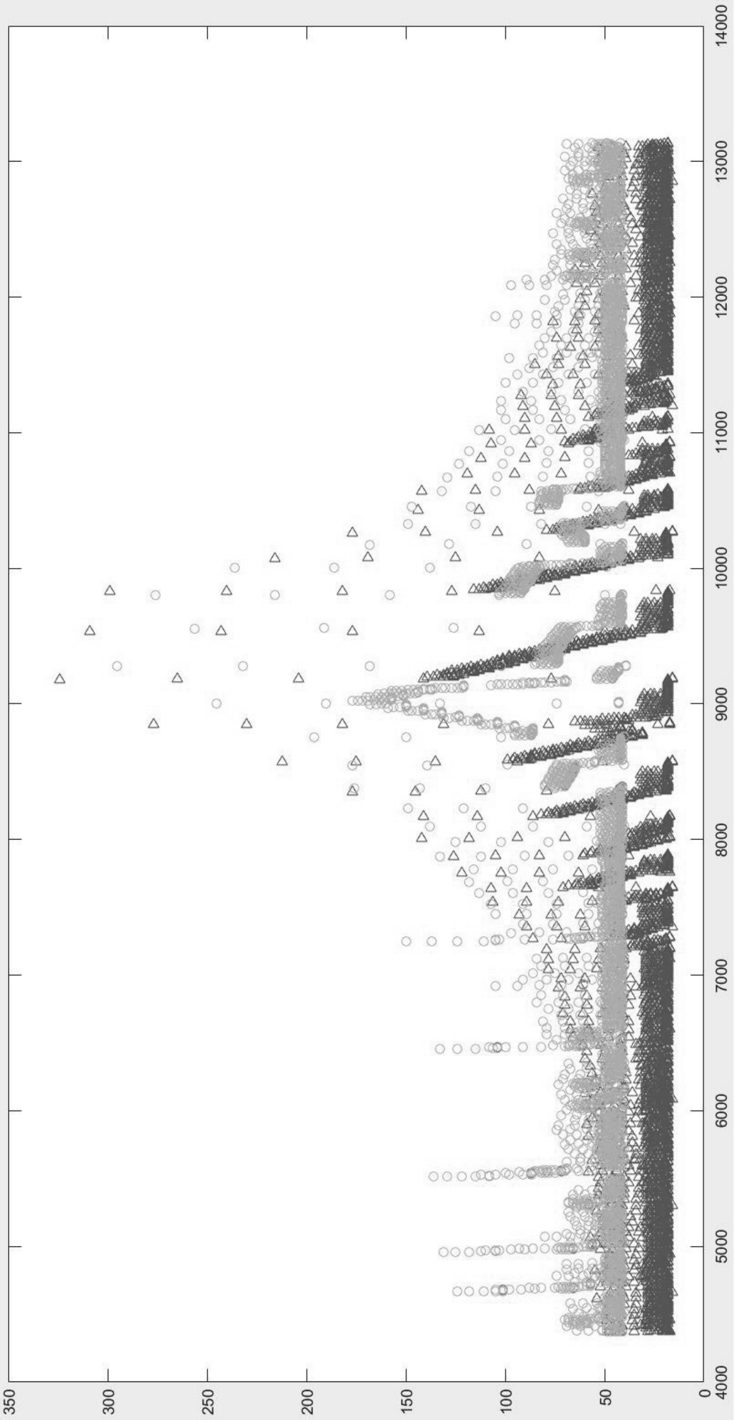


Figure 9.40: Time used for fish batches between the fish is ready for pick-up, and to processing is completed for a fleet consisting of four processing vessels (triangles) and four live fish carriers (circles).

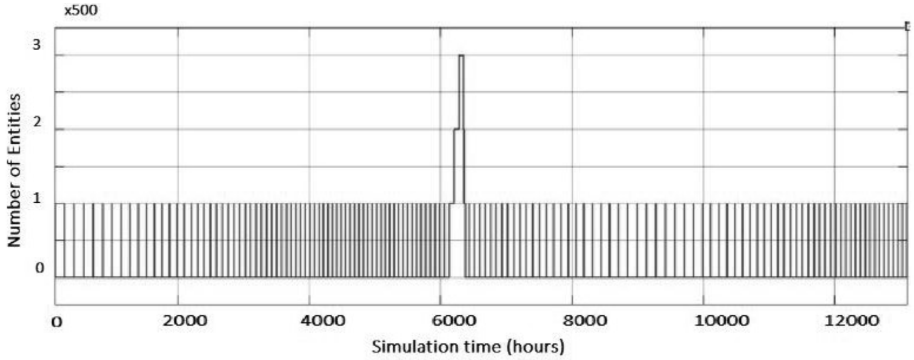


Figure 9.41: 500 tonne units of fish waiting for vessels at the traditional farms following an imposed emergency slaughter at the exposed farm at 6000 hours.

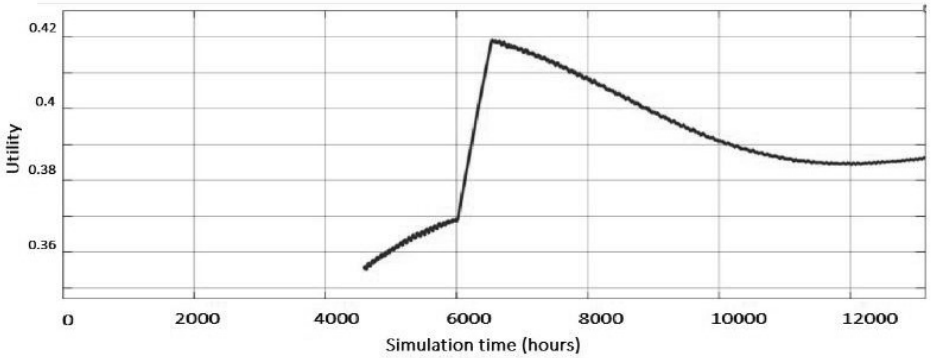


Figure 9.42: Utility at the processing facility during emergency slaughter scenario.

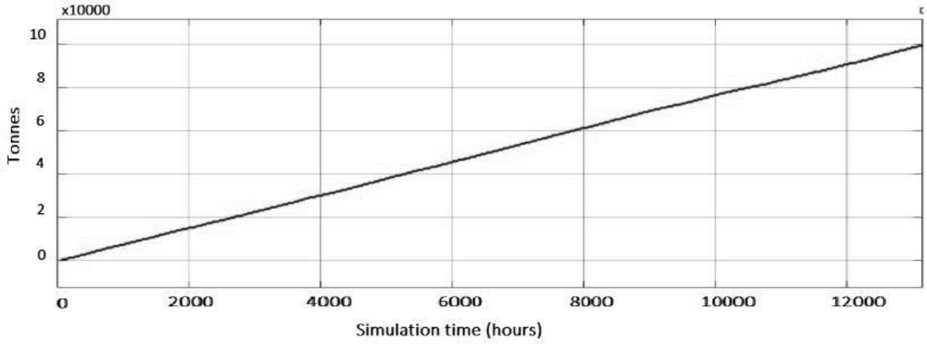


Figure 9.43: Entities loaded to cargo vessel during emergency slaughter scenario.

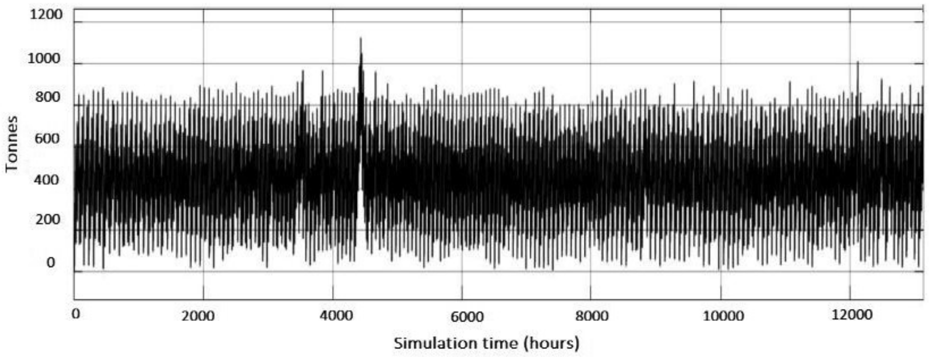


Figure 9.44: Amount of fish waiting to be picked up from exposed farm for simulation testing waiting cage prohibition on benchmark fleet.

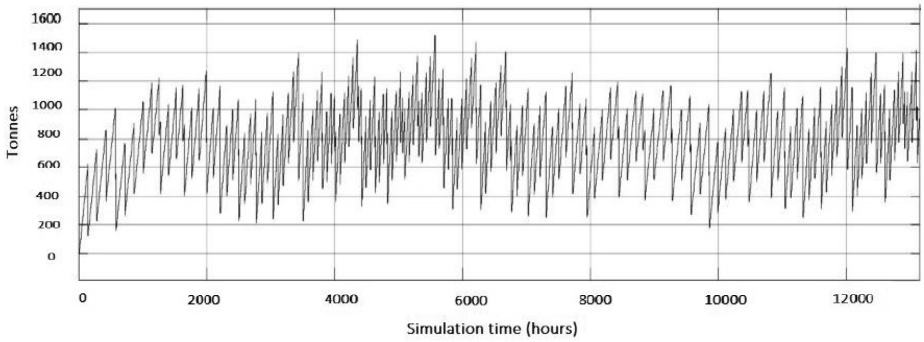


Figure 9.45: Amount of fish waiting to be picked up from traditional farms for simulation testing waiting cage prohibition on benchmark fleet.

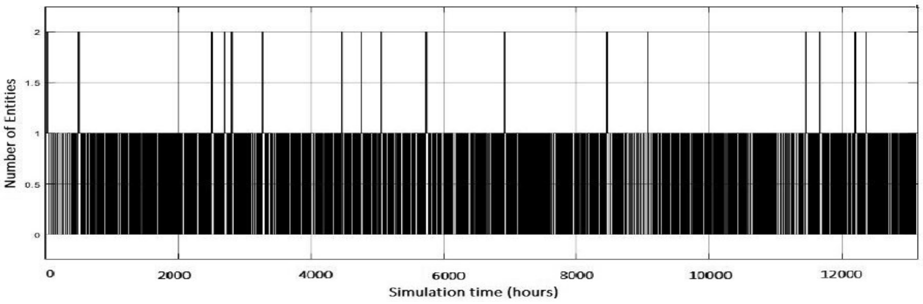


Figure 9.46: Processing vessels waiting for cargo from exposed farm during waiting cage prohibition.

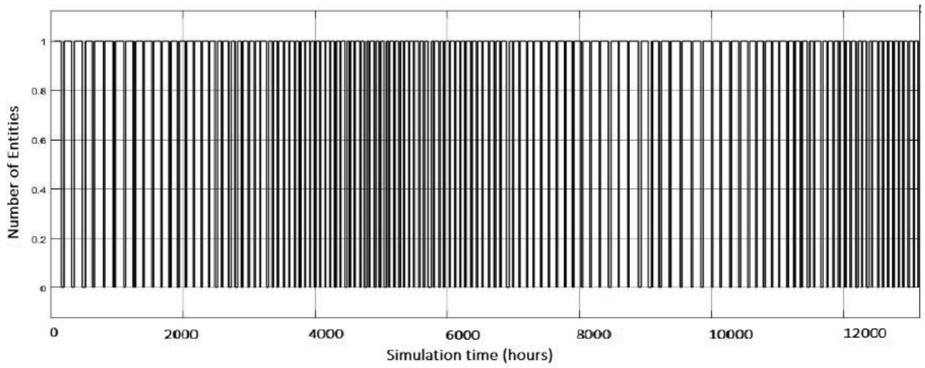


Figure 9.47: Processing vessels waiting for cargo from traditional farms during waiting cage prohibition.

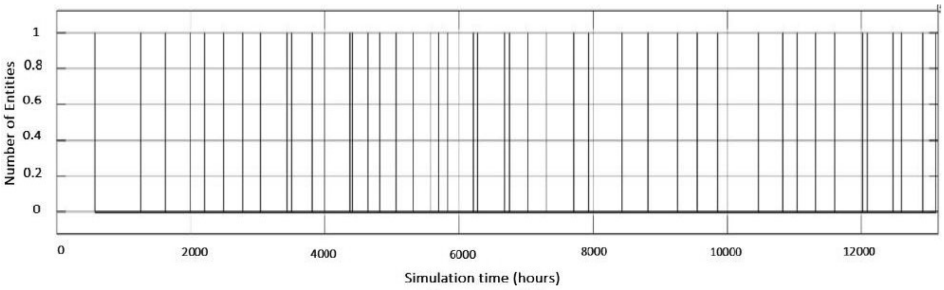


Figure 9.48: Live fish carriers waiting for cargo from traditional farms during waiting cage prohibition simulation.

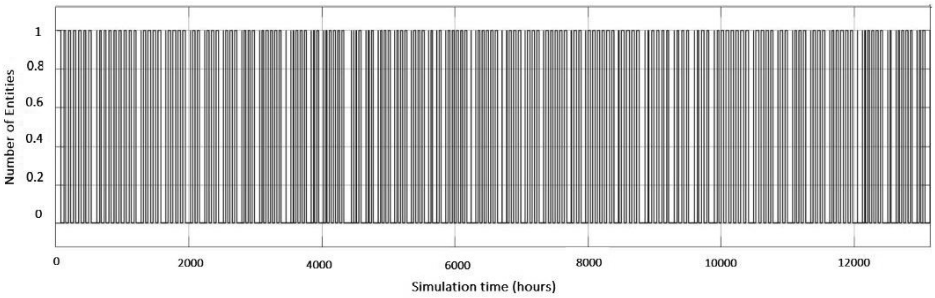


Figure 9.49: Live fish carriers waiting for cargo from exposed farm during waiting cage prohibition simulation.