Marte Fosen
Elina Nygaard

Solving the Multi-Objective Stochastic Cattle Feeding Planning Problem using the L-shaped Method

Master's thesis in Industrial Economics and Technology Management
Supervisor: Henrik Andersson
Co-supervisor: Atle Riise
June 2023

Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Economics and Management
Dept. of Industrial Economics and Technology Management

NTNU
Norwegian University of
Science and Technology

Marte Fosen
Elina Nygaard

Solving the Multi-Objective Stochastic Cattle Feeding Planning Problem using the L-shaped Method

Master's thesis in Industrial Economics and Technology Management
Supervisor: Henrik Andersson
Co-supervisor: Atle Riise
June 2023

Norwegian University of Science and Technology
Faculty of Economics and Management
Dept. of Industrial Economics and Technology Management

NTNU
Norwegian University of Science and Technology

Solving the Multi-Objective Stochastic Cattle Feeding Planning Problem using the L-shaped Method

Master's thesis in Industrial Economics and Technology Management

Marte Fosen
Elina Nygaard

June, 2023

Norwegian University of Science and Technology
Department of Industrial Economics and Technology Management
Faculty of Economics and Management

Supervisors:
Henrik Andersson, Professor - Department of Industrial Economics and Technology Management
Atle Riise, Senior Research Scientist - SINTEF Digital

Preface

This master's thesis concludes our Master of Science in Industrial Economics and Technology Management at the Norwegian University of Science and Technology, Department of Industrial Economics and Technology Management. The work is a continuation of our preparatory project completed in the fall of 2022.

We would like to thank our supervisor, Professor Henrik Andersson, and our co-supervisor, Atle Riise, for valuable guidance and feedback throughout the project. We would also like to extend our gratitude to Alexandre Teixeira from TKS, Anders Tørring from TINE and farmer Morten Lynum for valuable domain insights and inspiring discussions.

Marte Fosen and Elina Nygaard

Trondheim, June 7, 2023

Abstract

Developing a cost-efficient feeding plan that maximizes milk production is an important aspect of dairy farming. However, the feeding planning process is complicated by factors such as limited ingredient availability, uncertain nutritional content in ingredients, changing nutritional requirements of the dairy herd, and the trade-off between high feed quality and low costs. To address these challenges, this thesis presents the Stochastic Cattle Feeding Planning Problem (SCFPP). The aim is to investigate the value of using mathematical optimization to support farmers when determining a feeding plan. The SCFPP is formulated as a two-stage stochastic problem, taking uncertainty related to the content of the silage bales into account. The first-stage decisions decide which silage bales to use every day during the planning horizon, and are made when the actual content of each silage bale is unknown. The second-stage decisions are taken on the day of usage when the actual content of the silage bale is known. These decisions decide the combination of silage and feed concentrate for the daily feed compositions and aim to create feed compositions that satisfy the daily requirements of the dairy herd. The problem aims to both minimize costs and ensure correct and stable quality in the feed compositions. These two objectives are combined using a weighted-sum.

A literature review is conducted to get a broader understanding of the existing literature on diet- and blending problems with similar problem structures, as well as existing literature within the agricultural industry. Through this literature review, we identify three gaps in the literature that we address in this thesis. Firstly, while previous studies primarily focus on minimizing costs, our thesis addresses the trade-off between cost and feed quality. Secondly, this thesis incorporates factors such as limited feed availability, uncertain quality of ingredients, changing feeding requirements, and inventory management to the feeding problem. By doing so, we combine operational and tactical aspects to a greater extent than what has been done in previous literature. Lastly, by formulating the feed planning problem as a two-stage stochastic problem and solving it using the L-shaped method, we provide a novel approach to addressing the uncertainty in the problem.

The preparatory project to this thesis (Fosen & Nygaard, 2022) revealed that the Cattle Feeding Planning Problem (CFPP) is computationally heavy to solve. As the SCFPP is a continuation of this model where uncertainty is included, the L-shaped method is proposed as a solution method to improve the efficiency of the model. The L-shaped method is further accelerated by using the multi-cut version, adding a Two-Phase approach, generating Pareto-optimal cuts, and approximating the master solution. The computational results reveal that this accelerated L-shaped method outperforms the standard Gurobi solver in most cases and is able to achieve a gap smaller than 10% for instances where the Gurobi solver achieves 100% optimality gap.

The computational results of the SCFPP also demonstrate how the model may be used as a valuable tool for decision-making. Investigating the value of planning with uncertainty reveals that the stochastic model finds solutions with lower costs and improved quality of the meals, compared to

a deterministic approach. Furthermore, when resources are limited, planning for a longer horizon results in a more even distribution of the available ingredients. The computational results also illustrate that the prioritization of cost and quality has a considerable impact on the optimal feed compositions. This highlights the challenges associated with balancing the two objectives and demonstrates the value of a tool that allows the decision-maker to adjust the prioritization between them.

The SCFPP successfully demonstrates how mathematical optimization can be useful for farmers when determining feeding plans for dairy cattle. Although further research and development of the model is required to utilize it in a real-world scenario, it makes a valuable contribution to the existing literature within the field of farming optimization. We believe this work serves as an important starting point for future research in this area.

Sammendrag

Valg av fôringsstrategi har en stor påvirkning på både melkeproduksjonen og det økonomiske resultatet til en melkegård. Det er imidlertid flere faktorer som kan gjøre det utfordrende å utvikle en god fôringsplan. Noen av disse utfordringene er knyttet til en stor variasjon i næringsbehovene til dyrene, begrenset tilgang på ingredienser, usikker kvalitet på ingrediensene, og en balanse mellom fôrkvalitet og kostnader. For å adressere disse utfordringene, presenterer vi i denne masteroppgaven et planleggingsproblem for fôring av melkekyr (Stochastic Cattle Feeding Planning Problem (SCFPP)). Formålet er å undersøke verdien av å bruke optimering til å utvikle en god fôringsplan. Problemet er formulert som et to-stegs stokastisk problem som inkluderer usikkerhet knyttet til innholdet i rundballer. Beslutningene i første steg bestemmer hvilke rundballer som skal brukes på hvilken dag i planleggingshorisonten. Disse beslutningene tas når det faktiske innholdet i rundballene er usikkert. Beslutningene i andre steg tas når usikkerheten er realisert og det faktiske innholdet i rundballene er kjent. Disse beslutningene bestemmer hvordan man kan kombinere surfor og kraftfor til fôrsammensetninger som tilfredsstiller de daglige kravene til melkekyrene. Problemet har som mål å minimere kostnadene og sørge for at kvaliteten på fôret holdes på et riktig og stabil nivå. Disse to objektivene kombineres ved hjelp av en vektet sum.

Et litteraturstudie er gjennomført for å få innsikt i eksisterende forskning på diettproblemer og relevant litteratur innen jordbruksindustrien. Gjennom dette litteraturstudiet identifiserer vi tre områder som tidligere forskning ikke dekker, og som vi tar sikte på å addressere i denne masteroppgaven. For det første, mens tidligere studier hovedsakelig har fokusert på kostnadsminimering, tar vår modell hensyn til både kostnad og fôrkvalitet, og utforsker balansen mellom disse to faktorene. For det andre inkluderer vår modell faktorer som begrenset fôrtilgang, usikkerhet knyttet til innholdet i rundballer, variasjoner i ernæringsmessige krav hos dyrene og aspekter knyttet til lagerstyring. Så langt vi vet, har disse faktorene ikke blitt kombinert tidligere. Til slutt, ved å formulere fôrplanleggingsproblemet som et to-stegs stokastisk problem, presenterer vi en ny tilnærming for å håndtere usikkerheten i problemet.

SCFPP er et komplekst problem og vi benytter derfor L-shaped metoden for å løse modellen mer effektivt. Denne metoden er utforsket i kombinasjon med ulike aksellerasjonsmetoder som for eksempel å legge til flere kutt for hver iterasjon, å dele løsningsmetoden inn i to steg, å generere Pareto-optimale kutt og å approksimere løsningen på masterproblemet. Resultater fra beregningsstudiet viser at den aksellererte L-shaped metoden lykkes i å forbedre effektiviteten til modellen, og klarer å oppnå et optimalitetsgap på 10% i tilfeller der Gurobi oppnår et gap på 100%.

Analyser for å undersøke verdien av å bruke SCFPP som et beslutningsverktøy innen fôring er også gjennomført. Resultater fra analysene viser at den stokastiske modellen, SCFPP, finner løsninger som både gir lavere kostnader og bedre kvalitet på fôret sammenliknet med en deterministisk modell. I tillegg viser resultatene at man oppnår en jevnere fordeling av begrensede ressurser ved å forlenge planleggingshorisonten. Videre viser analysene at prioriteringen av kostnad og kvalitet

påvirker de optimale fôrsammensetningene, noe som demonstrerer verdien av et verktøy som lar beslutningstakeren prioritere mellom disse objektivene.

Denne masteroppgaven lykkes i å vise hvordan optimering kan være nyttig for melkebønder når en fôringsplan skal lages. Selv om videre utvikling er nødvendig for å kunne ta modellen i bruk i virkeligheten, mener vi at arbeidet er et godt utgangspunkt for fremtidig forskning.

Table of Contents

List of Figures

List of Tables

Acronyms

BC	Base Case
CFPP	Cattle Feeding Planning Problem
DP	Dynamic Programming
EEV	Expected result of using the Expected Value solution
EV	Expected Value
IW	Initializing Warm start
LP	Linear Programming
MW	Magnanti-Wong
NDF	Neutral Detergent Fiber
NTNU	Norwegian University of Science and Technology
RP	Recourse Problem
RW	Robust Warm start
SCFPP	Stochastic Cattle Feeding Planning Problem
TKS	T. Kverneland & Sønner AS
TP	Two-Phase
VSS	Value of the Stochastic Solution

Glossary

LB-stop Acceleration method where the solution process in the master problem is stopped when a solution value equal to the current best lower bound is found

SolNum Acceleration method where the solution process in the master problem is stopped when a given number of feasible solutions are found

Chapter 1

Introduction

Agriculture is an important industry in Norway, providing the country with milk, meat, cereals, and vegetables. With increased geopolitical uncertainty and supply disruptions due to climate issues, domestic food production has gained further importance (Regjeringen, 2021). According to SSB (2022), there are approximately 38 000 agricultural enterprises in the country today, and around 6 700 of these are farms that engage in milk production (Opplysningskontoret for Meieriprodukter, 2022). The dairy sector has achieved high self-sufficiency, surpassing 98% in 2022 (Landbruksdirektoratet, 2022), making it an important contribution to Norway's food production and security.

Planning and decision-making within the agricultural industry have traditionally been based on experience. However, with technological advancements, there has been an increasing adoption of innovative production and automation systems, which has stimulated the need for decision support systems based on mathematical optimization. This has resulted in a significant increase in research within this field (Glen, 1987). Some researchers focus on strategic decisions related to investments in technology, and decisions related to optimal harvesting strategies (Crabtree, 1981; Moghaddam & DePuy, 2011). Others address tactical problems related to the allocation of feeding resources (Bellingeri et al., 2020; Meyer & Newett, 1970; Notte et al., 2020; Polimeno et al., 1999), or operational decisions related to developing optimal feed compositions for animals (Dean et al., 1969; Goswami et al., 2013; Gupta et al., 2013; Rahman et al., 2015; Zhang et al., 2021).

This thesis focuses on both tactical and operational decisions related to the feeding planning process within dairy farming. According to TINE Rådgivning (2012), animal feeding is associated with a large part of the total costs within milk production, and the choice of feeding strategy consequently has a large impact on the financial result for a dairy farm. In addition, a well-designed feeding strategy that meets the specific nutritional requirements in each stage of a cow's life cycle can improve milk production, reproductive performance, and overall animal health. However, the feeding planning process is complicated by several factors. These include changing nutritional requirements of the dairy herd, limited availability of ingredients such as silage bales and uncertainties related to their nutritional content, and the balance between cost-efficiency and high feed quality. Determining and planning a good feeding strategy is therefore an important and challenging task for the dairy farmer, and advanced decision support systems can be important for streamlining the process, optimizing resource allocation, and improving the cost efficiency and quality of the animal feed.

In this thesis we formulate and study the Stochastic Cattle Feeding Planning Problem (SCFPP).

The purpose is to investigate the value of using mathematical optimization to support dairy farmers in developing feeding plans. The primary focus of the problem is to determine how to distribute a limited number of silage bales over a planning period, and how to use them in combination with feed concentrate to create high-quality and cost-effective meals for dairy cattle. Building upon the deterministic Cattle Feeding Planning Problem (CFPP) studied in the preparatory project by Fosen and Nygaard (2022), this thesis extends the problem formulation by incorporating uncertainty related to the dry matter and nutritional content in silage bales. The hypothesis driving this research is that considering this uncertainty when making feeding plans for dairy cows can lead to cost savings and an overall improvement in feed quality.

The purpose of this thesis is to be achieved through three goals. In the preparatory project for this thesis (Fosen & Nygaard, 2022), the CFPP proved to be computationally heavy to solve. Therefore, the first goal is to develop a solution method capable of effectively addressing the high computational complexity of the SCFPP, which becomes more challenging when uncertainty is introduced. The second goal is to demonstrate the value of planning for a longer time horizon while considering uncertainty when developing a feeding plan for dairy cattle. Lastly, the third goal is to investigate the trade-off between cost and quality and develop a model that enables farmers to prioritize between cost and quality when developing a feeding plan.

Through this research, three significant contributions are made to the existing literature. Firstly, while prior studies within feeding planning primarily focus on cost minimization, this thesis introduces a multi-objective model that addresses the trade-off between quality and cost. Secondly, it improves the integration of tactical and operational decisions by incorporating aspects such as limited feed availability, uncertain ingredient quality, changing feeding requirements, and inventory management to the traditional feeding problem. To the best of our knowledge, simultaneously addressing these aspects is not explored in the existing literature. Lastly, by formulating the model as a two-stage stochastic model and solving it using the L-shaped method, we provide a novel approach to addressing uncertainty within feeding planning.

The thesis is written in collaboration with SINTEF, one of Europe's largest independent research organizations, and T. Kverneland & Sønner AS (TKS), a Norwegian producer of advanced farming equipment. TKS is currently developing a platform for automated feeding logistics for animal farms, where SINTEF is engaged to develop optimization models needed for the automated solution. As an established player in the agricultural industry, TKS has contributed with deep industry knowledge and an understanding of the current challenges faced by farmers. Meanwhile, SINTEF's expertise in optimization, including their experience within the farming domain, has provided valuable insights to this thesis. The collaboration with SINTEF and TKS has been an important part of defining the problem, and regular discussions and knowledge exchange has served as important foundations for this thesis.

The thesis is structured as follows. Chapter 2 introduces the cattle feeding process, and provides necessary background information about dairy cattle feeding requirements, ingredients, and feeding supply chain logistics and challenges. Chapter 3 presents the reviewed literature and positions the SCFPP in this context. In Chapter 4, we provide an in-depth description of the problem, and the mathematical model used to solve the problem is presented in Chapter 5. Subsequently, Chapter 6 introduces the solution- and evaluation methods used. In Chapter 7, the data sets and parameters generated to test the model are introduced, before Chapter 8 presents and discusses the computational results of our study. Lastly, concluding remarks and future research are presented in Chapter 9.

Chapter 2

Background

This chapter introduces the necessary background information for this thesis. The content in this section builds upon the findings from the preparatory project of this thesis (Fosen & Nygaard, 2022), supplemented with additional relevant information. A major part of the insights presented in this chapter is derived from discussions with our industry partners, SINTEF and TKS, who possess valuable knowledge within the field. Furthermore, an increased understanding of the dairy industry's perspective and challenges is gained through interviews with TINE. To ensure a thorough understanding of the topic, we also contacted a farmer, who shared practical knowledge and experiences from an on-the-ground point of view. Additional insights are gathered from online research and research articles to obtain additional insights and ensure a comprehensive understanding of the problem.

First, Section 2.1 presents important characteristics within the dairy cattle feeding process. This includes an introduction to the general feeding requirements for dairy cattle, an introduction to the animal dynamics, and lastly a presentation of relevant ingredients. Thereafter, the feeding supply chain for dairy cattle is described in Section 2.2. Lastly, some of the main challenges when determining a feeding strategy are presented in Section 2.3.

2.1 Animal Feeding Characteristics

The following section presents characteristics that are important to understand within the dairy cattle feeding process. This includes an introduction to the general feeding requirements of dairy cattle, an overview of the dairy cattle's lifecycle and how this impacts the feeding requirements, as well as a description of ingredients used to construct daily feed compositions.

2.1.1 Feeding Requirements

Cattle are rumen animals and consequently process food differently than other animals. An important distinction in a cattle's digestive system is that the stomach has four separate compartments, allowing cattle to digest grass or vegetation without completely chewing it first. This makes it possible for cattle to convert vegetation into usable energy more efficiently than other animals. The conversion is done by rumen microbes, which consist of fungi, bacteria, and protozoa that are produced in the rumen part of the animal's stomach. These microbes work together to break down the feed and turn it into usable energy and protein. Providing a diet that ensures a high production of rumen microbes is therefore an important aspect of cattle feeding to ensure a successful milk production. This is done by providing a stable and consistent diet with feed compositions that satisfies the daily requirements of the animals.

When determining feed compositions for the animals, it is important to provide the correct intake of dry matter and nutrients. Ensuring a correct dry matter intake is important to obtain high milk production, growth, reproduction, and body condition for the animals. Therefore, dairy cattle typically have a target value, as well as an upper and lower limit of dry matter intake, usually expressed in kg per day. The main determinant of dry matter in dairy cows is body weight, and typically a dry cow, i.e a cow that is not lactating, has to consume dry matter corresponding to 2-3% of its body weight per day. For higher-producing lactating cows, a minimum amount of 4% is often required. (Department of Agriculture, Fisheries and Forestry and Dairy Australia, 2013a).

In addition to dry matter intake, other nutritional requirements are essential to ensure optimal production. Some important nutrients to consider are Neutral Detergent Fiber (NDF), energy, sugar, and protein, and it is important to ensure a correct intake of these nutrients. A too high intake may both affect cow health, and reduce the feed conversion efficiency, i.e the kg of milk produced per kg of feed. At the same time, a too low intake may have a significant impact on milk production, fertility and cow health (West et al., 1991). Therefore, dairy cattle often have a target, upper and lower limit of the different nutrients. The intake of nutrients is dependent on the total dry matter intake, and nutritional requirements are therefore often expressed per kg dry matter. The feed composition requirements of dairy cattle strongly depend on the animal's purpose and phase, which is further discussed in Section 2.1.2.

For optimal rumen microbe growth and activity, it is important to provide dairy cattle with a consistent diet with minimal changes. This is because rumen microbes take time to recover after sudden feed changes, and may use up to six weeks to fully adjust to a change of feeds. (Department of Agriculture, Fisheries and Forestry and Dairy Australia, 2013b) Thus, if the feed composition is changed frequently, the rumen microbes will not be able to convert the feed into useable energy in an optimal manner which affect the milk production and performance of the animal. It is therefore crucial to keep the diet as stable as possible and make adjustments gradually if needed.

2.1.2 Animal Dynamics

Dairy cattle go through several phases during their life cycle, depending on their growth, lactation, and gestation. The feeding requirements change in line with the phase the animal is in, and providing a balanced diet that meets these changing requirements is challenging.

Table 2.1 summarizes some of the main phases that dairy cattle go through. Calf, Youngstock, and Heifer are cows in the growth stage, that have not yet had their first calf. For these animals, it is important to provide them with a diet that supports healthy growth. For instance, it is important to feed calves with feed consisting of a high amount of protein, to support high muscle and skeletal development. After the first calving, the cattle can go through phases five to eight several times. This is referred to as the lactation cycle, which is the period between one calving and the next. During the lactation period, i.e phase five to seven, the cows produce milk, and proper feeding management is important to ensure high milk production and quality. Phase eight is the dry period, where the cattle gather strength to begin a new lactation process. Managing the different phases in the lactation cycle is important, as variations in milk production, feed intake, and weight of the animal have a significant impact on the feeding requirements. Figure 2.1 illustrates the four cycles and the changes that occur. As the figure shows, the milk production is highest during the first months of lactation, leading to higher feed requirements to support the production. Phase nine represents cattle that do not fit into any of the other phases, i.e animals who are sick or not lactating. The dynamics between the phases complicate the feeding logistics and make it difficult for the farmer to predict future feeding needs.

Table 2.1: Overview of animal groups in a dairy farm including a description of when cattle typically are in each phase.

Phase number	**Animal group**	**Description**
1	Calf	< 6 months
2	Youngstock	6 months - first pregnancy (typically 17 months)
3	Heifer	During first pregnancy (until calving)
4	In-calf cow	During pregnancy (2nd time or more)
5	Lactation - early	0 - 75 days after calving
6	Lactation - mid	75 - 150 days after calving
7	Lactation - late	> 150 days after calving
8	Dry Cow	0 - 65 days after finished lactating
9	Other	Not specified

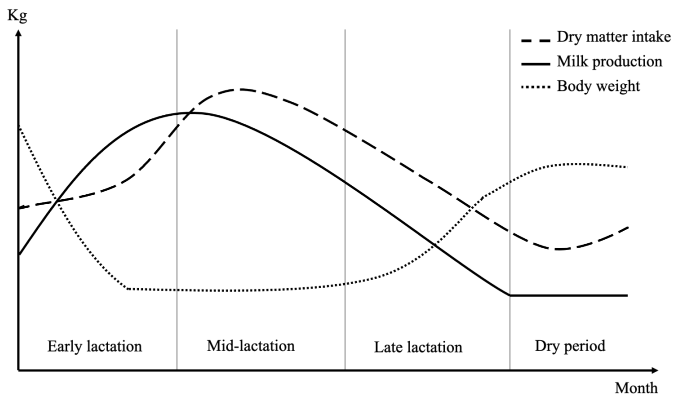


Figure 2.1: Variations in dry matter intake, milk production and body weight during lactation cycle.

2.1.3 Ingredients

The feeding of dairy cattle is done by mixing ingredients into a meal that aim at satisfying the daily feeding requirements of the animals. A typical feed composition consists of forage such as hay, silage, and grains, supplemented with vitamins and minerals from feed concentrate. In this thesis we focus on silage stored as bales and different types of feed concentrate.

Silage Bales

Baled silage is forage that is baled at higher moisture than forage stored as dry hay. The hay is fermented, which means that it is harvested green and preserved in a more-or-less airtight container until being used. The goal of this preservation process is to maintain the nutritional quality of the forage, allowing farmers to have a reliable source of feed for their livestock throughout the year. If handled correctly, silage bales can offer high-quality feed that is both tasty and nutritionally balanced for dairy cattle.

However, the quality of silage, in terms of both dry matter and nutritional content, may vary greatly depending on several factors such as cultivation strategy, soil quality, weather conditions and harvesting policy. Within the same field, there may be large variations in the quality and content of the silage. This is shown in Figure 2.2, where the dry matter content of the silage within a field is illustrated by a satellite image, showing the variation. In addition, handling, wrapping, and storage conditions have a great impact on the fermentation process, affecting the quality of the silage bales. Silage bales are sensitive to air exposure, and damages to the bale wrap due to carelessness during transportation or storing may lead to mold and yeast in the silage, significantly reducing its quality. This may even lead to the silage bale being useless to the farmer, and TKS informs that around 20% of the silage bales that are considered useless are due to bad wrapping.

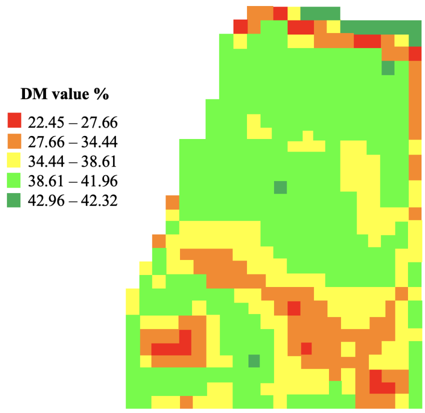


Figure 2.2: Satellite picture of the field showing the variance in dry matter content. Red color indicates a low dry matter content while green indicate high dry matter content. The picture is received from TKS.

Furthermore, the quality of silage is perishable, and both the nutritional- and dry matter content is reduced over time with a varying degree of deterioration. TKS estimates that a high-quality silage bale typically has a loss of nutritional content of 0.5-1% every month, while a silage bale with a less successful fermentation process may experience a loss of up to 10% every month. These variations may make it difficult to accurately predict the nutritional content of a silage bale and design an optimal feeding strategy.

To get an indication of the silage bales, farmers often perform a silage analysis 5-6 weeks after ensiling, based on samples from the silage bale batch. This gives an expected value of the dry matter- and nutritional content. However, the actual quality of the silage bale is often not discovered until the day of usage, when the silage bale is opened.

Feed Concentrates

Feed concentrates are feed additives that are rich in nutrients, such as protein, fiber, vitamins, and minerals. They are often added to the feed composition as a supplement to forage and help provide a balanced and complete diet for the animals. Feed concentrates play a crucial role in cattle feeding, especially in cases where the forage is of lower quality with an insufficient concentration of certain nutrients. The nutritional and dry matter content of feed concentrate is known and stable, and it can be stored over a long period of time without deteriorating.

Feed concentrates are bought from the market, and are associated with a significantly high cost compared to forage such as silage bales. Farmers can choose from a range of feed concentrates, depending on the specific nutritional needs of the animals. For instance, some types of feed concentrate are high in protein and suitable for growing or high-producing cattle. Others are higher in fat and may be a good option if the focus is high content of fat in the milk.

The amount of feed concentrate used in each feed composition depends on various factors, including the animals' life cycle phase, level of production, and their specific nutritional requirements. For instance, lactating cattle have higher nutritional requirements to support their milk production, and therefore, a larger amount of feed concentrate is often needed to meet their needs and ensure high-quality milk. On the other hand, when dealing with animals that are unwell or not producing milk, the primary focus shifts to maintenance rather than production, resulting in a smaller amount of feed concentrate needed to fulfill their nutritional needs. While feed concentrate provides concentrated nutrients, it lacks certain essential components such as roughage, fiber, and minerals that are necessary for proper digestion and overall well-being. Therefore, feed concentrate alone cannot constitute the complete diet of a dairy cattle.

2.2 The Feeding Supply Chain

We have identified the main steps in the feeding supply chain of dairy cattle, illustrated in Figure 2.3. The goal of this section is to give a general description of the different parts of the supply chain and the activities that occur in each step.

2.2.1 Production and Procurement of Ingredients

The first-stage in the animal feeding supply chain is the production and procurement of ingredients. Silage is often self-grown in one or several fields. To produce high-quality silage bales, farmers need to make strategic decisions regarding crop production. This involves choosing which crops to grow, deciding when to plant them, determining the area to use for each crop, and whether or not they should use fertilizer. However, uncertainties related to weather conditions, pests, and diseases can complicate this planning process and greatly affect crop quality.

In addition, farmers must make tactical decisions related to harvesting schedules. Limited harvesting resources on the farm can make it difficult to harvest all crops at an appropriate time, and farmers must decide how to best perform the harvesting operations to ensure the best quality silage. Therefore, self-produced silage bales become available in batches at different times throughout the year due to both harvesting seasons and schedules.

Feed concentrates are often bought in large quantities from the market. Different types of feed

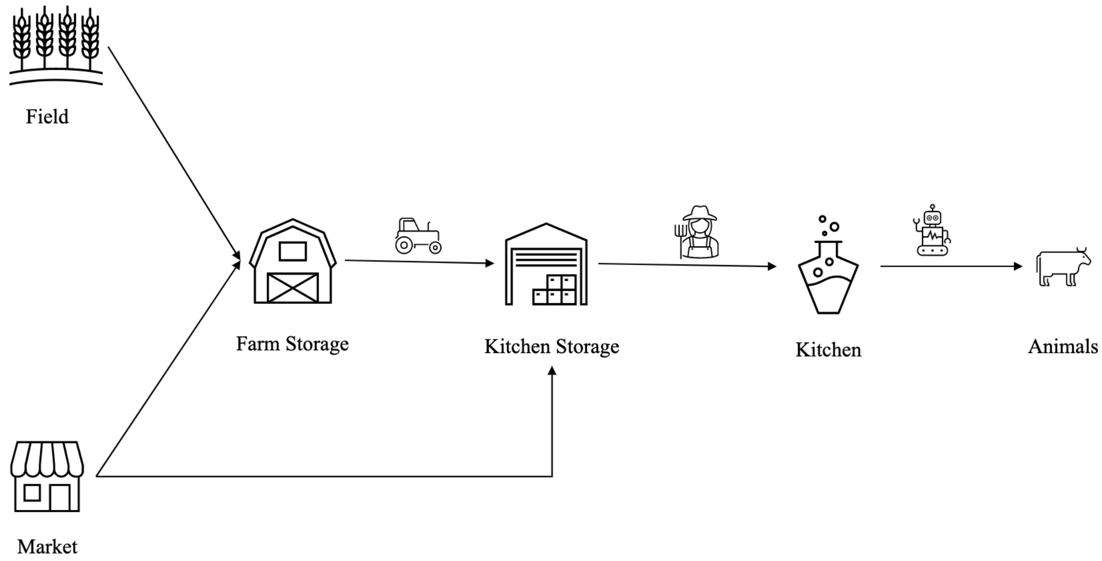


Figure 2.3: An overview of the feeding supply chain, illustrating the identified steps from production and procurement to the feeding of cattle.

concentrate are available with varying nutritional content. When procuring feed concentrate, farmers need to ensure that the purchased ingredients meet the specific requirements of the dairy herd while keeping costs within budget.

2.2.2 Long-Time Storage of Ingredients

The second step in the supply chain is the storage of the ingredients. Silage bales are often long-time stored on the field, inside a barn, or in another big storage unit. After the silage has been rolled into bales, the bales should be transported to the long-time storage immediately to reduce the risk of damaging the bales. The longer time it takes before the bales are moved, the less firm they will be, and moving them may result in collapsed silage bales with significantly reduced quality. The silage bales have to be stored in long-time storage for at least six weeks before they are used. This is done for the conservation of the ingredients to be complete.

2.2.3 Kitchen Storage and Mixing

The area where the feed compositions are made is referred to as the kitchen. The number of kitchens in a farm depends on the farm size, where larger, industrial, farms may have multiple kitchens, while smaller to medium-sized farms usually only has one. Each kitchen typically has an associated kitchen storage, as well as a mixing area. On the day of usage, the relevant silage bales are moved to the kitchen storage, which is a storage close to the mixing area. The kitchen storage has limited capacity and is intended for storing the silage bales that are planned used on a given day. However, the feed concentrates are stored in the kitchens storage at all times.

In the mixing area, the ingredients for the feed compositions are mixed together by a mixing machine. The silage bales are opened one by one and sequentially loaded into the mixing machine. Feed concentrate is added in the end, before the feed composition is mixed together. After the mixing, the meals are distributed to the relevant animals. This can be done manually or by an automated feeding robot.

2.3 Challenges in Determining a Feeding Strategy

Determining a good feeding strategy can be challenging, and the farmer must manage a range of complicating factors. In this section, we present some of the challenges a farmer experiences when determining a feeding strategy for dairy cattle. We also discuss some of the strategies they may use to overcome these challenges and develop a successful feeding plan.

2.3.1 Uncertain Nutritional Content of Ingredients

As described in Section 2.1.3, the actual dry matter and nutritional content of silage bales is uncertain. This complicates the feeding planning process as it may be difficult to ensure that the animals are receiving a correct and consistent diet. Today, farmers often use sample quality analysis as a basis for planning feed compositions for the animals. However, due to both changes in the nutritional content over time, and uncertainties in the quality of the individual silage bales, these sample analyses are not reliable for determining the actual content of the silage bale. This may lead to suboptimal feed compositions, which can lead to reduced production, milk quality, appetite, and animal welfare. Therefore, farmers often have to readjust their feeding strategy after they observe how the animals respond to the feed.

To limit the challenges related to uncertainty, farmers may take the potential variability in the quality of silage bales into account when developing a feeding strategy. Moreover, individual silage bales could be tested on a regular basis to get more accurate information about the nutritional content. This may allow for supplementing the silage with other feed concentrates or adding nutritional supplements to ensure that the animals are receiving all the necessary nutrients.

2.3.2 Resource Utilization Planning

Due to harvesting seasons and schedules, silage bales of different quality become available at different times during the year. It is necessary to distribute these ingredients in such a way that the daily feed intake satisfies the requirements for dry matter and nutritional content. For instance, it might be suboptimal to use all silage bales with the highest quality at the beginning of the planning period as this might lead to future feed compositions not satisfying the requirements. Furthermore, this may lead to higher variations in the feed compositions over the planning horizon.

Tactical planning related to the distribution of ingredients over a given time horizon must therefore be made. Several factors complicate these decisions. As described in Section 2.1.2, the animal dynamics complicate the feeding planning process as feeding requirements vary between the different phases. As the animals evolve, the requirements change, making it difficult to plan the usage and distribution of ingredients. In addition, uncertainties regarding the actual quality of silage bales further complicate the planning process.

2.3.3 Trade-off Between Cost and Quality

Another challenge in determining an optimal feeding strategy is the trade-off between serving the animals with high-quality feed compositions and keeping the cost low for the farmer. Ingredients with high nutritional content, such as feed concentrate or high-quality silage, are generally associated with better milk production and animal health. However, these ingredients often come at a higher cost than lower-quality feed. On the other hand, using lower-quality feed is more cost-effective, but may lead to reduced milk production and potential health issues. Therefore, the farmer must find a balanced feeding strategy, aiming to maintain animal health and productivity, while keeping costs within budget. This requires careful planning and decision-making based on factors such as the availability and price of different ingredients, as well as the nutritional requirements of the animals at different stages of their life cycle.

Chapter 3

Literature review

This section presents relevant literature reviewed for this thesis. First, Section 3.1 presents the strategy used for finding the literature. Second, Section 3.2 presents literature on diet and blending problems, as these problems have a similar problem structure as the SCFPP. Thereafter, Section 3.3 presents existing literature within the farming industry. Section 3.2 and Section 3.3 are reproduced from the preparatory project by Fosen and Nygaard (2022) and mainly address deterministic problems. Section 3.4 present relevant literature that addresses uncertainty, and the solution methods used. Finally, Section 3.5 positions this thesis in the existing literature and provides the motivation for this thesis.

3.1 Search Strategy

This section presents the search strategy for the literature review. The search is conducted using Scopus as the primary search portal due to its advanced search capabilities. However, additional relevant literature is found through Google Scholar by examining references, as they have a larger selection of articles.

The initial search is divided into two main searches: mathematical model search and industry search. The goal of the mathematical model search is to find literature related to similar problem structures, while the industry search aims to provide an overview of related literature in the field of agriculture. In both searches, common keywords such as ”optimization” and ”mathematical model” are used to find literature within the field of optimization. Additionally, search words such as ”stochastic”, ”multi-objective” and ”uncertainty” are used to find problems addressing similar aspects as the SCFPP. Table 3.1 provides an overview of the search words used for the two searches. Note that some search words have an asterisk at the end. This is done to ensure that all variants of a word are captured.

Table 3.1: Overview of search keywords for literature review.

Search	**Search Within**	**Keywords**
Mathematical Model Search	Title	(Diet, menu, blending) In combination with: (Planning, Problem or Model)
Industry Search	Title	(Farm*, Feed*, Animal, Cattle or Dairy) In combination with: (Planning, Problem or Model)
Common for both searches	Title, Abstract, Keywords	(Optimization, Mathematical Model, Operations research or Decision Support System) In combination with: (Stochastic, Multi-objective, Uncertain*, Limited resources or Allocation)

The procedure for finding relevant articles is illustrated in Figure 3.1. After performing the initial search for both models, the results are combined, duplicates are removed, and only English articles are considered. The results are then filtered by removing irrelevant subject areas such as "Energy" and "Arts & Humanities", and irrelevant keywords such as "Wind power". Titles and abstracts of the remaining articles are carefully reviewed and additional relevant articles are found by examining citations. By following this procedure, we end up with 30 articles that thoroughly read.

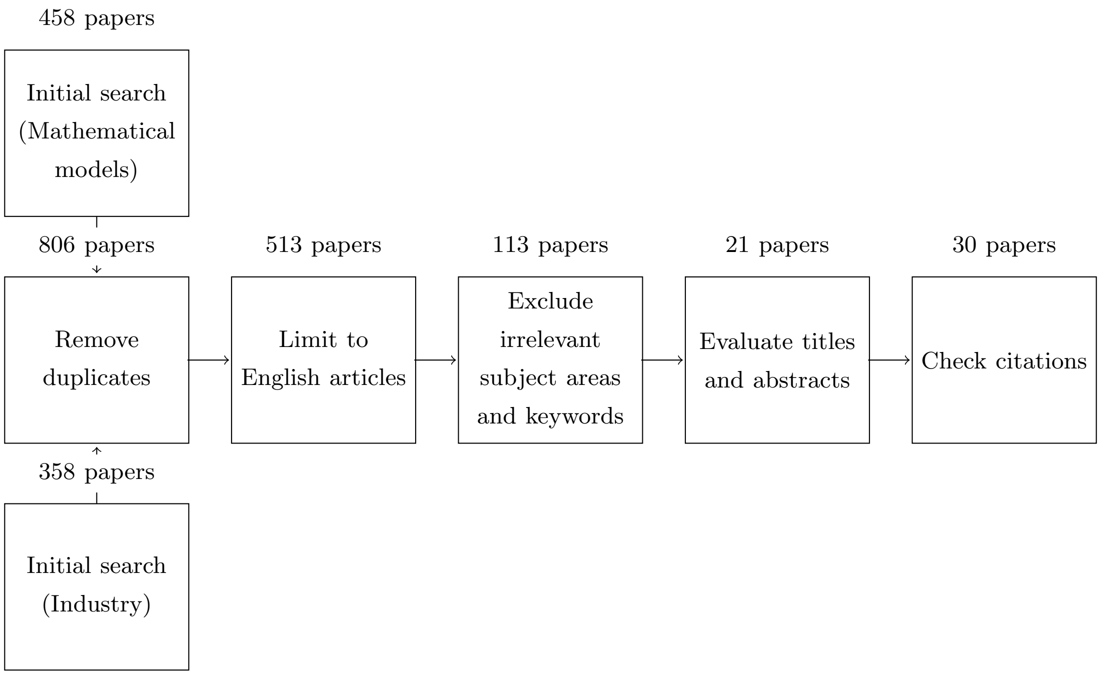


Figure 3.1: An overview of the material collection for literature review.

3.2 Blending and Diet Problems

The blending problem is one of the oldest optimization models solved using the simplex method. The problem involves determining the optimal blend of ingredients to create one or more products while respecting composition-related specifications. The objective of the problem is to minimize costs. The first variant of the blending problem is the problem proposed by George Stigler (Stigler, 1945), referred to as Stigler's diet problem. This problem determines the optimal combination of different ingredients and aims to minimize costs while satisfying nutritional requirements. The proportions of the nutrients in each food are given, in addition to the cost of the food. Furthermore, the recommended intake of each nutrient is known.

Since Stigler's diet problem was introduced, the problem has appeared in literature for several different application areas. Osman and Sufahani (2022) design a problem to ensure that sinusitis patients achieve a healthy diet, while Sufahani and Ismail (2015) develop a mathematical model to optimize the weekly diet plan for Malaysian school children, ensuring that the necessary nutrient intake is achieved. As for the traditional diet problem, both models aim at minimizing cost.

Some researchers extend diet models to include other objectives than minimizing costs. The Thrifty Food Plan (TFP), originally developed by the Agricultural Research Service in 1974-1975, aims to develop nutritious diets focusing on tastiness and variety. The objective of the model is to minimize the deviation from previous food consumption patterns while meeting nutritional goals, and the model includes the cost perspective as budget constraints (Garille & Gass, 2001). Other objectives may include minimizing the total glycemic load (Bas, 2014) or minimizing the environmental footprint (Gephart et al., 2016). Researchers also include multiple objectives in their problem formulations. Amin et al. (2019) present a multi-objective model with four objectives: minimize total cost, minimize saturated and trans fats in the foods, minimize sugar in the foods and maximize the amount of vitamins. The problem is solved using both the weighted-sum method and the ϵ-constraint technique. Cakrak and Cimen (2017) develop a model for optimal meal planning in the military. They consider two objectives: minimizing the total costs of the food supply chain and maximizing the variety of the dishes. This is solved by adding an objective function composed of penalties and goals.

Diet problems can be solved for multiple periods. When planning an optimal diet for the military, Cakrak and Cimen (2017) consider a time horizon of 365 days. The problem is solved using a genetic algorithm, and food-supply chain aspects such as ordering frequency, ordering costs, inventory costs, and inventory capacity are taken into account. Amin et al. (2019) also consider a multi-period model, over a time period of one month. In the model, they consider how both the costs of the ingredients and the nutritional requirements may vary from day to day. Furthermore, Osman and Sufahani (2022) and Sufahani and Ismail (2015) develop a model for a multi-period horizon, respectively five days and a week. Both models are constrained by each food only being allowed to use once during the planning horizon, and they both solve this using a delete reshuffle algorithm. This algorithm runs in a loop and solves the model for one day at a time. As food is selected for a given day, this food is removed as a possible choice for the rest of the days in the planning horizon (Osman & Sufahani, 2022; Sufahani & Ismail, 2015).

Some problems take limitations regarding raw material availability into account. Fomeni (2018) presents a tea blending problem, where the aim is to make tea blends where the content is as close as possible to the target recipe. The original tea blend problem aims to make tea blends that correspond exactly to their target recipe. However, due to limited resource availability, this is not always feasible. Fomeni (2018) proposes two methods to account for this. First, he develops a parametric relaxation of the model. The model allows the requirements to be within a specified bound, and Monte Carlo simulation is used to simulate different values of the bounds. Second, he introduces a multi-objective optimization model, aiming to minimize the total cost of the tea blend as well as the violation of the targeted blend characteristics.

Limited resources may also lead to tactical decisions involving production planning aspects such as resource allocation, inventory planning, or purchasing decisions, as done by Cakrak and Cimen (2017) in the military problem introduced earlier. Gómez-Pantoja et al. (2021) introduce a food bank resource allocation problem, aiming to help food banks decide how to allocate limited resources among different beneficiaries. The problem focuses on deciding which beneficiaries to serve and what products to deliver, taking aspects such as inventory management, purchasing policy, and nutritional content into account. The mathematical model is formulated as a Mixed Integer Problem (MIP), and an adaptive heuristic is used to solve larger instances of the problem.

3.3 Planning Problems Within the Farming Industry

The farming industry has undergone significant changes over the past century, primarily due to advances in technology. This advancement has led to increased complexity and importance in farm planning (Glen, 1987), and multiple decision-support models have been proposed to address this increasing complexity. In this section, we provide an overview of the existing literature on planning problems in the farming industry within three planning levels: strategic-, tactical-, and operational planning. The main focus is on literature related to animal feeding at the tactical and operational planning levels as this is most relevant for the SCFPP. However, we give a brief overview of the strategic planning level as well.

3.3.1 Strategic Planning Level

The strategic planning level includes decisions with a long-term planning horizon, typically one year or more. The scope of these problems is often broad and the decisions are made on a fairly high managerial level. Within the farming industry, strategic decisions involve investments in technology, storage space or land, and decisions regarding what and how many acres to harvest. All of these decisions aim at making the farm more profitable in the long run. For an example regarding investment in equipment, we refer to Crabtree (1981). With regards to the harvesting strategy, Moghaddam and DePuy (2011) create a model aiming to find the optimal harvesting strategy for maximizing profit.

3.3.2 Tactical Planning Level

The tactical planning level involves decisions with a planning horizon between a few months and five years. Within farming, these decisions often include the allocation of resources, as well as decisions related to harvesting and storage. There exist several models considering the allocation of feeding resources within farming. Some of these models aim at distributing the limited ingredients in an optimal manner over a given time period (Bellingeri et al., 2020; Notte et al., 2020). Bellingeri et al. (2020) solve the resource allocation problem with a linear programming approach, while Notte et al. (2020) use the Differential Evolution (DE) algorithm as a solution method. Other models, such as Glen (1980), Meyer and Newett (1970) and Polimeno et al. (1999), include the animals' evolution in their models, aiming at making feed compositions that result in a certain weight for the animals. This is done by taking a dynamic programming approach.

An important aspect of farming is the relationship between the dairy products the farm produces and the resources it uses, such as crops or food for the animals. Effective management of these relationships can help a farm to be sustainable and profitable. Bellingeri et al. (2020) approach this by creating a model that selects the optimal cropping plan while simultaneously creating an allocation plan for the different ingredients. The model is solved for one year at a time and aims at finding the most cost efficient plan for harvesting and feeding while taking limited storage capacity and field limitations into account. A similar model is created by Kikuhara et al. (2009), where the feeding costs over one year are minimized while ensuring high production levels and satisfied feeding requirements.

The objective function in tactical farm feeding problems is often subject to cost or income optimization (Bellingeri et al., 2020; Glen, 1980; Kikuhara et al., 2009; Meyer & Newett, 1970; Polimeno et al., 1999; Sirisatien et al., 2009). However, Notte et al. (2020) present a model aiming at maximizing milk production, margin over feed costs and herbage intake, while also minimizing the diet cost and the supplement intake.

3.3.3 Operational Planning Level

The time horizon for the operational planning level is short, and the planning is typically performed on a daily basis. Decisions involve determining detailed plans regarding the transportation of bales from the field to the storage unit, feeding of animals, and allocation of production to machines. This section focuses on decisions related to feed composition, since this is considered to be most relevant for this thesis.

Most of the existing problems regarding animal feeding focus on the monetary aspects of the creation of feed compositions. For instance, Dean et al. (1969) approach the least-cost feed formulation for dairy cattle by solving a linear programming model with the objective to maximize profit while still meeting the animal's feeding requirements. The model aim at selecting the concentrate and roughage components for the feed composition, the amount of feed each animal should be fed, and the quantity of milk production in an optimal manner. Goswami et al. (2013) solves a similar problem, creating the least cost balanced diet for small dairy farmers in India by solving a linear programming program. Furthermore, also Gupta et al. (2013) and Rahman et al. (2015) solve the diet problem for farm animals with objective to minimize the associated costs.

Although many researchers solve the feed composition problem using a linear programming approach, Gupta et al. (2013) states the limitations of these approaches and solves the problem using the Controlled Random Search Technique and Genetic Algorithm. The result is a faster model with small deviations from the optimal solution. Furthermore, Rahman et al. (2015) solve the problem using the Evolutionary Algorithm. They find that this solution approach provides feasible solutions in all runs and prove that it is robust to parameter changes. Lastly, Zhang et al. (2021) use the Improved Tabu Search heuristic to create optimal feed compositions for pigs.

3.4 Methods for Handling Uncertainty

In the literature reviewed in Section 3.2 and Section 3.3, all input data is assumed to be known and deterministic. However, in reality, many parameters are subject to uncertainty, making the mathematical models more complex. In this section, we provide an overview of the relevant literature on diet problems and feeding problems that incorporates uncertainty. We study the various types of uncertainties that are included in problems and review the solution methods used to handle this.

A common way of addressing uncertainty in diet and feeding problems is by using a stochastic programming framework, where either the objective function or constraints are treated with probabilistic terms. Bas (2014) proposes a diet problem with the objective of minimizing the total glycemic load in a human diet. A robust optimization approach is used to account for uncertainty in the glycemic load values of different ingredients. Within animal feeding, Van de Panne and Popp (1963) are amongst the first to consider the uncertainty of the nutritional content in feed ingredients. They formulate a feed mixing problem for cattle with probabilistic, non-linear protein constraints. Similarly, Patil et al. (2022) discusses the optimization problem of formulating the lowest-cost ration for Indian goats, taking into account variations in the nutritional content of various ingredients. They formulate a stochastic model to solve this problem, where they replace the nutritional content criteria with a probabilistic one. Other researchers, such as Rahman and Bender (1971), and Chen (1973), propose a linear approximation of the probabilistic constraint. While Rahman and Bender (1971) aim to minimize the cost of the feed mix while meeting nutrition requirements, Chen (1973) aim to maximize the probability of success while minimizing costs and uses an iterative quadratic programming technique to solve the problem. Patil et al. (2022) suggests both the use of a linear and non-linear stochastic model for solving the feeding problem for goats. Peña et al. (2009) presents a multi-objective stochastic model for designing feed compositions for pigs. Instead of fixing the desired level of probability, they suggest a multi-objective model, aiming to minimize the cost of the ration and to maximize the probability of meeting the nutrient requirements of the animal.

Other authors use simulation to address the effects of uncertainty. Shalloo et al. (2004) develop a stochastic simulation model of a dairy farm, including uncertainties related to milk price, feed concentrate cost, and silage quality. They use Monte Carlo simulation to determine the influences of stochastic parameters and estimate the resulting farm profitability. Trebeck and Hardaker (1972) investigates a stochastic farm planning problem through the integrated use of simulation and linear programming. The article discusses an optimization problem faced by a beef producer and includes strategic decisions related to the stocking of feed, as well as tactical decisions related to cattle management over a time period of a year. The problem includes uncertainty in terms of weather variability, affecting the quality of the crops and the amount of feed available during the year. Simulation is used to evaluate possible outcomes of strategic decisions, and use this as input for making tactical decisions.

Amin et al. (2019) use a scenario-based approach to develop an optimization problem for solving a multi-objective diet problem, considering uncertainty in both the nutritional content and the cost of ingredients. Other researchers have applied a two-stage stochastic formulation to account for uncertainty in their models. Flaten and Lien (2007) formulate a two-stage stochastic model for a dairy farm that aims to maximize the expected net income. Their model accounts for uncertainties regarding crop yield, where the actual quality is known when the crops are harvested. The decisions in this model are related to production activities, labor activities, and the number of heifers to obtain. Udomsungworagul and Charnsethikul (2018) formulate a two-stage linear programming problem that is used to optimize the feed mix production planning in the animal feed industry. Its purpose is to find the optimal feed mix with the minimum cost fulfilling nutrient requirements, taking into account the uncertainty of the nutrient content in the different ingredients and the demand for different product mixes. The first-stage decisions are related to the quantity of each ingredient used in the final mix, while the second-stage variables determine the results of these decisions in each scenario. To improve the effectiveness of the model, they solve the problem using a combination of Dantzig-Wolfe decomposition and Benders decomposition.

3.5 Our Contribution

As presented in this chapter, many researchers have studied problems related to diet and meal planning, as well as planning problems within farming. In this section, we describe our contribution to the existing literature and motivate this thesis.

While the majority of diet problems within animal feeding focus on reducing costs, we introduce a multi-objective optimization problem considering the trade-off between cost and quality. Both cost and quality provide a high contribution to the overall profitability of the farm, and by taking a multi-objective approach, we aim to better reflect the trade-offs that dairy farmers face when making feeding decisions. Furthermore, we introduce the importance of having a stable diet. As far as we know, this aspect of cattle feeding has not been explored in literature within optimization but is an important consideration for dairy farmers as described in Section 2.1.

We also integrate tactical decisions related to resource utilization and operational decisions related to daily feeding to a larger extent than what has been done in previous literature. Within the farming industry, there has been research on both tactical and operational decisions, as presented in Section 3.3. Some of these studies take the impact of limited resources or the animal dynamics into account. Meanwhile, other researchers have their main focus on the creation of feed compositions, aiming to satisfy the feeding requirements of the animals, both considering uncertainty and not. However, as far as we know, these aspects have not been combined. We fill this gap by proposing a model that incorporates operational decisions related to constructing optimal feed compositions, while also taking into account tactical aspects such as limitations in feed availability, uncertain quality, changing feeding requirements, and inventory aspects. By doing so, we provide a more comprehensive understanding of the problem that can lead to more effective feeding strategies.

Our study addresses the challenge of uncertainty in the nutritional content of ingredients, which is typically handled using methods such as chance constraints, quadratic programming, fuzzy programming, or simulation in existing research. We address a gap in the literature by formulating and solving the problem as a two-stage stochastic program and using the L-shaped method to improve the effectiveness of the model. Similar to our problem, Udomsungworagul and Charnsethikul (2018) address uncertainty in the nutritional content of products and formulates the problem as a two-stage stochastic problem. However, in their model, first-stage decisions focus on feed mixing, while second-stage decisions are linked to the outcome and quantity of products sold. Consequently, Udomsungworagul and Charnsethikul (2018) lack flexibility in feed mixing after the nutritional content is known. In contrast, our approach assumes that the quality of ingredients is known on the day of usage, allowing for the adjustment of the feed mix in the second-stage, once the actual quality of ingredients is known. By using this two-stage approach, we aim to better reflect the daily uncertainty faced by dairy farmers and provide increased flexibility in decision-making processes related to feed planning.

In summary, our study combines a multi-objective perspective, integrates tactical and operational decisions, incorporates uncertainty in ingredient quality, and suggests a solution method that is rarely used in previous animal feeding problems. This approach offers valuable insights and contributions to the field of dairy cattle feeding. Table 3.2 provides an overview of a selection of the reviewed literature and compares important elements to our report. The table illustrates categories that are relevant to this thesis, with the cells highlighted in blue indicating similarities with the SCFPP. From analyzing the table, it becomes clear that none of the reviewed literature is entirely comparable to the SCFPP, emphasizing how our study differentiates itself from the existing literature.

Table 3.2: Comparison of the relevant literature and the SCFPP.

Reference	Planning level	Objective	Limited Resources	Uncertainty	Solution Method
Van de Panne and Popp (1963)	Operational	Minimize cost	No	Yes	Chance constrained programming
Dean et al. (1969)	Operational	Maximize profit	Yes	No	LP
Meyer and Newett (1970)	Tactical	Minimize cost	No	No	LP and DP
Rahman and Bender (1971)	Operational	Minimize cost	No	Yes	Stochastic LP
Trebeck and Hardaker (1972)	Strategic / Tactical	Maximize Payoff	Yes	Yes	Integrated simulation and LP
Chen (1973)	Operational	Minimize cost	No	Yes	Iterative quadratic programming
Glen (1980)	Tactical	Minimize cost	No	No	LP and DP
Polimeno et al. (1999)	Tactical	Maximize income	No	No	LP and DP
Tozer (2000)	Operational	Minimize cost	No	Yes	LP with RHS-adjustment, safety margin and stochastic programming
Shalloo et al. (2004)	Strategic	Maximize profit	Yes	Yes	Stochastic simulation
Flaten and Lien (2007)	Strategic / Tactical	Maximize net income	Yes	Yes	Two-stage stochastic
Kikuhara et al. (2009)	Tactical	Minimize cost	No	No	LP
Peña et al. (2009)	Operational	Minimize cost and maximize probability of meeting requirements	No	Yes	Stochastic programming
Goswami et al. (2013)	Operational	Minimize cost	No	No	LP
Gupta et al. (2013)	Operational	Maximize milk yield	No	No	Heuristic and Genetic Algorithm
Udomsungworagul and Charnsethikul (2018)	Tactical	Minimize cost	No	Yes	Two-stage stochastic, Dantzig-Wolfe and Benders decomposition
Amin et al. (2019)	Tactical / Operational	Minimize cost and maximize health	No	Yes	Scenario-based approach
Notte et al. (2020)	Tactical	Maximize profit and milk production	Yes	No	Heuristics: Pareto-frontier Differential Evolution (DE)
Bellingeri et al. (2020)	Tactical	Minimize cost	Yes	No	LP
Zhang et al. (2021)	Operational	Minimize cost	No	No	Heuristic
Patil et al. (2022)	Operational	Minimize cost	Yes	Yes	Stochastic programming
Our model	Tactical / Operational	Cost and Quality	Yes	Yes	Two-Stage Stochastic/L-shaped

Chapter 4

Problem Description

This chapter provides a detailed description of the Stochastic Cattle Feeding Planning Problem (SCFPP). The SCFPP aims to determine how to distribute silage bales over a given time period and how to use them in combination with feed concentrates to construct daily feed compositions that satisfy the animals' feeding requirements while minimizing the costs. The problem takes the uncertainty of the actual quality of the silage bales into account. First, Section 4.1 provide a description of the characteristics defining the problem. Afterward, Section 4.2 gives an overview of the objectives and decisions of the SCFPP.

4.1 Problem Characteristics

The problem consists of a set of animal groups, i.e. dairy cattle in different phases. The animal groups have daily feeding requirements in terms of both dry matter and nutritional content. We assume that the daily amount of feed for each animal group is mixed together at the beginning of the day. In this mixture, each animal group has a target for daily dry matter content, as well as a minimum and maximum limit that must not be exceeded. Furthermore, the content of each nutrient should be as close to a given target value as possible, preferably within a specified interval. The feeding requirements for each animal group can change from day to day, as the animals continuously evolve.

A set of ingredients are available for constructing the feed compositions, consisting of different types of silage bales and feed concentrates. There is a cost related to the usage of the silage bales (per bale) and different types of feed concentrates (per kg). Silage bales become available in batches at different times during the time horizon. As silage bales become available, they are stored at the barn for long-time storage in a specific order. Feed concentrates are stored in the kitchen storage, and we assume that the supply of feed concentrates is unlimited. Every ingredient has a dry matter content (kg) and nutritional density (per kg dry matter) of each nutrient. While feed concentrates have stable and known content, the content of silage bales is both perishable and uncertain. The dry matter content and nutritional density of a silage bale depend on which batch it belongs to, what day it is used, as well as individual differences between the bales. Until a silage bale is opened, the actual nutritional content of the bale is unknown. On the day of usage, the true values are learned, and it is possible to adapt to this new knowledge by using feed concentrate. However, every animal group has a maximum allowable amount of feed concentrate that can be used in the feed composition. This amount is expressed as a fraction of the target dry matter content in the composition. Reserve silage can also be added to the feed composition, but this is associated with a significantly high cost. During the mixing of ingredients, we assume that you do not need to use an entire silage bale before opening another.

A set of kitchens determines where the feed compositions may be produced, and the composition for one animal group can only be made in one kitchen for a given day. Furthermore, all kitchens have an associated storage capacity, specified in the number of silage bales that can be stored in one day. Silage bales that are planned to be used on a given day must be transported from the barn to the associated kitchen storage at the beginning of the day of usage. Therefore, only silage bales that have been transported to the kitchen can be used for constructing feed compositions. As air exposure ruins the conservation of hay, we assume that all leftover silage in a kitchen is thrown away at the end of the day.

4.2 Objectives and Decisions

The SCFPP is a multi-objective problem aiming to both minimize costs and ensure a correct and stable content in the feed compositions. The cost objective includes minimizing the costs related to the usage of ingredients. The quality objective is split into two parts. The first part aims to minimize deviations from the target in terms of dry matter and nutritional content. The second part aims at ensuring a stable diet by minimizing fluctuations between the nutritional and dry matter content in a feed composition from one day to the next for every animal group.

The objectives of the SCFPP are to be achieved through several decisions. These decisions can be divided into two main groups:

- *Tactical decisions related to resource allocation planning.* This includes determining which silage bales to transport to which kitchen every day during the planning horizon. Furthermore, it includes determining in which kitchen the daily feed compositions for the different animal groups should be produced. The tactical decisions related to resource allocation planning are performed once at the beginning of the planning period.
- *Operational decisions related to daily feed compositions.* These decisions are made daily and aim to decide how much silage and feed concentrate to use in the final feed compositions for each animal group. The final feed compositions are made after the silage bales are opened and the actual quality is known.

An overview of the model is given in Figure 4.1. The figure summarizes the decisions and the objectives of the model and illustrates what part of the value chain the specific decisions are related to.

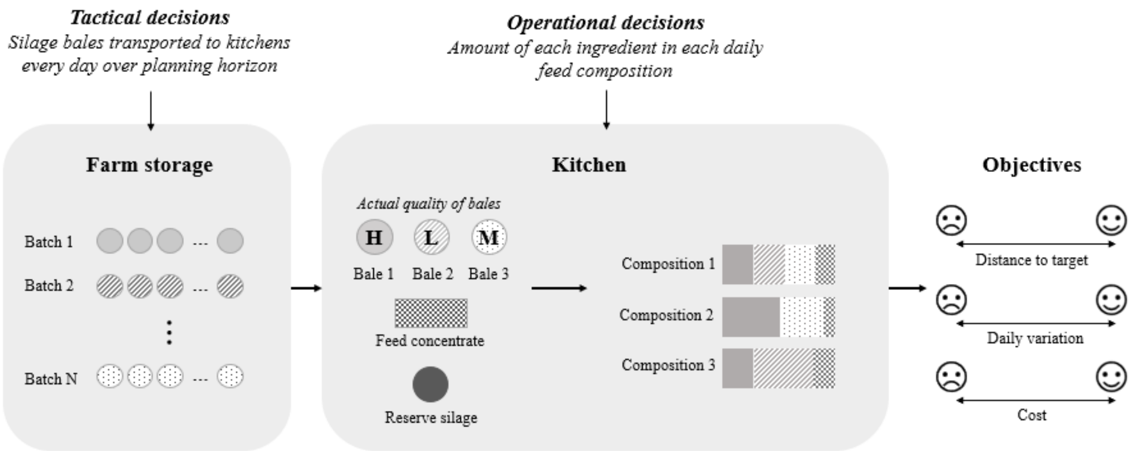


Figure 4.1: An overview of the decisions and objectives in the SCFPP, illustrating what part of the cattle feeding value chain the specific decisions are related to.

Chapter 5

Mathematical Model

This section presents our mathematical formulation for the Stochastic Cattle Feeding Planning Problem (SCFPP), described in Chapter 4. The model is a continuation of the preparatory project by Fosen and Nygaard (2022), but is modified and extended by including uncertainty. Section 5.1 introduces the modeling approach used, before the mathematical model is presented in Section 5.2.

5.1 Modelling Approach

A two-stage approach is used to formulate and solve the SCFPP. The first-stage decisions are related to resource allocation planning, determining which silage bales to transport to which kitchen on which day during the planning horizon. The second-stage decisions are related to the daily feed compositions and are made after the actual content of the silage bales is known. Figure 5.1 illustrate the value chain of the problem, as well as the input and output of the two stages.

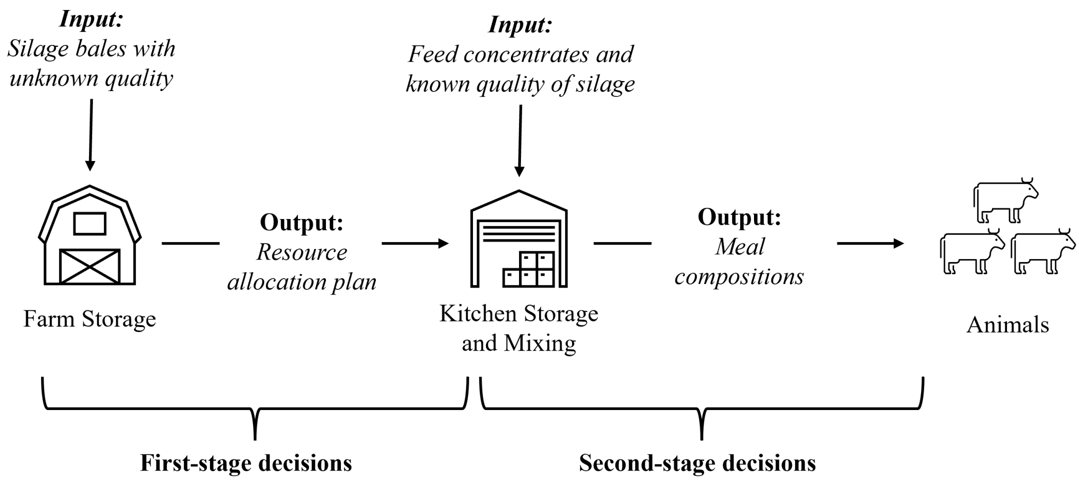


Figure 5.1: Illustration of the stochastic modelling approach, dividing the problem into two stages with separate decisions.

A set of scenarios describes the uncertainty in the problem, where each scenario expresses a possible realization of quality for all silage bales available during the planning period. The scenarios are generated using a sample-based approach, where the quality of a silage bale in a scenario is drawn from a known distribution. This distribution is described in Chapter 7.

5.2 Mathematical Formulation

The following section presents the mathematical formulation for the SCFPP. First, the notation is introduced, followed by the objective functions. Thereafter, the constraints are presented together with a short description. The compressed version of the model can be found in Appendix A.

5.2.1 Notation

Sets and Indices

$\mathcal{A}$	Set of animal groups, $a \in \mathcal{A}$
$\mathcal{D}$	Set of days in the planning horizon, $d \in \mathcal{D}$
$\mathcal{I}$	Set of silage bale batches, $i \in \mathcal{I}$
$\mathcal{F}$	Set of feed concentrate types, $f \in \mathcal{F}$
$\mathcal{K}$	Set of kitchens, $k \in \mathcal{K}$
$\mathcal{N}$	Set of nutrients, $n \in \mathcal{N}$
$\mathcal{S}$	Set of possible scenarios, $s \in \mathcal{S}$
$\mathcal{B}_i$	Set of silage bales from batch i
$\mathcal{D}_i$	Set of days when silage bale batch i is available, $\mathcal{D}_i \subseteq \mathcal{D}$
$\mathcal{I}_d$	Set of silage bale batches available on day d, $\mathcal{I}_d \subseteq \mathcal{I}$

Parameters

C^B_{di}	Cost per silage bale from batch i on day d
C^R	Cost of using reserve silage
C^F_f	Cost per kg used of feed concentrate type f
D^A_i	The day silage bale batch i becomes available
F^B_{bdis}	Amount (kg) dry matter in silage bale b from batch i in scenario s on day d
F^F_f	Fraction of dry matter in feed concentrate type f
F^{MAX}_a	Maximum fraction of dry matter from feed concentrate in the feed composition for animal group a
K_k	Daily storage capacity in kitchen k
N^B_{bdins}	Nutritional density of nutrient n (per kg dry matter) in silage bale b from batch i in scenario s on day d
N^F_{fn}	Nutritional density of nutrient n (per kg dry matter) in feed concentrate type f
N^R_n	Nutritional density of of nutrient n (per kg dry matter) in reserve silage bale
L^M_{ad}	Minimum dry matter (kg) for animal group a on day d
U^M_{ad}	Maximum dry matter (kg) for animal group a on day d
T^M_{ad}	Target dry matter (kg) for animal group a on day d
L^N_{adn}	Minimum amount (per kg dry matter) of nutrient n for animal group a on day d
U^N_{adn}	Maximum amount (per kg dry matter) of nutrient n for animal group a on day d
T^N_{adn}	Target amount (per kg dry matter) of nutrient n for animal group a on day d
P_s	Probability of scenario s
P^M	Penalty for deviating from dry matter target
P^{IN}	Penalty for deviating from nutrient target when the content is between target and minimum or maximum value
P^{ON}	Penalty for deviating from nutrient target when the content is outside the minimum or maximum limit
R^M_a	Relative deviation from dry matter target for animal a on the last day before planning horizon start
R^N_{an}	Relative deviation from target for nutrient n for animal a on the last day before planning horizon start

Decision Variables

x_{abdkis}	Amount (kg) dry matter from silage bale b from silage bale batch i used in the feed composition for animal group a on day d in kitchen k in scenario s
r_{ads}	Dry matter (kg) from reserve bale used in feed composition for animal group a on day d in scenario s
y_{adfs}	Amount (kg) of feed concentrate type f used in feed composition for animal group a on day d in scenario s
d^M_{ads}	Deviation in dry matter content in feed composition for animal group a on day d in scenario s
d^N_{adns}	Deviation in nutritional content of nutrient n for animal group a on day d in scenario s
p^M_{ads}	Penalty related to deviation from dry matter target in feed composition for animal a on day d in scenario s
p^N_{adns}	Penalty related to deviation from target for nutrient n in feed composition for animal a on day d in scenario s
$v^M_{a(d-1)ds}$	Variation in deviation from target for dry matter content in two consecutive feed compositions, from day $(d-1)$ to day d for animal group a
$v^N_{a(d-1)dns}$	Variation in deviation from target for nutrient n in two consecutive feed compositions, from day $(d-1)$ to day d for animal group a in scenario s
m_{bdik}	1 if silage bale b from batch i is transported to kitchen k on day d, 0 otherwise
w_{adk}	1 if the feed composition for animal group a on day d is produced in kitchen k, 0 otherwise

Weight Parameters

α	Weight of quality objective

5.2.2 Modeling

Quality Objective

$$f^Q = \min \sum_{s\in\mathcal{S}} P_s(\sum_{a\in\mathcal{A}}\sum_{d\in\mathcal{D}}((p^M_{ads} + \sum_{n\in\mathcal{N}} p^N_{adns}) + (v^M_{a(d-1)ds} + \sum_{n\in\mathcal{N}} v^N_{a(d-1)dns})) \tag{5.1}$$

The quality objective function consists of two parts. The first part aims to ensure that the dry matter and nutritional content in each feed composition is as close as possible to the target value by penalizing deviations from the target. The second part minimizes the variation in deviation from day to day.

Cost Objective

$$f^C = \min \sum_{d\in\mathcal{D}} \sum_{i\in\mathcal{I}_d} \sum_{b\in\mathcal{B}_i} \sum_{k\in\mathcal{K}} C^B_{di} m_{bdik} + \sum_{s\in S} P_s (\sum_{a\in\mathcal{A}} \sum_{d\in\mathcal{D}} (\sum_{f\in\mathcal{F}} C^F_f y_{adfs} + C^R r_{ads})) \tag{5.2}$$

The objective function that aims at minimizing costs includes the costs related to the usage of ingredients. This includes the cost of transporting silage bales in the first-stage, as well as the cost of using feed concentrate and reserve silage in the second-stage.

Weighted Sum Objective

$$Z = \min \ \alpha f^Q + (1-\alpha) f^C \tag{5.3}$$

We have chosen to formulate the multi-objective problem as a weighted sum as shown in Equation (5.3). The weights, α and $(1-\alpha)$, represent the importance of each objective.

Resource Allocation Constraints

$$\sum_{k\in\mathcal{K}} w_{adk} = 1 \qquad a\in\mathcal{A}, d\in\mathcal{D} \tag{5.4}$$

$$\sum_{d\in\mathcal{D}_i} \sum_{k\in\mathcal{K}} m_{bdik} \leq 1 \qquad i\in\mathcal{I}, b\in\mathcal{B}_i \tag{5.5}$$

$$\sum_{d'=1}^{d} \sum_{k\in\mathcal{K}} (m_{(b-1)d'ik} - m_{bd'ik}) \geq 0 \qquad d\in\mathcal{D}, i\in\mathcal{I}_d, b\in\mathcal{B}_i \setminus \{1\} \tag{5.6}$$

$$\sum_{i\in\mathcal{I}_d} \sum_{b\in\mathcal{B}_i} m_{bdik} \leq K_k \qquad d\in\mathcal{D}, k\in\mathcal{K} \tag{5.7}$$

Constraints (5.4) ensure that each feed composition is mixed in exactly one kitchen. Constraints (5.5) make sure that each silage bale can only be used once. Constraints (5.6) ensure that bales that are stacked in front are used first. Constraints (5.7) limit the number of transported silage bales to a kitchen on a given day to the kitchen's daily storage capacity.

Production Constraints

$$x_{abdiks} \leq M_1 w_{adk} \qquad a\in\mathcal{A}, d\in\mathcal{D}, i\in\mathcal{I}_d, b\in\mathcal{B}_i, k\in\mathcal{K}, s\in\mathcal{S} \tag{5.8}$$

$$\sum_{a\in\mathcal{A}} x_{abdiks} \leq F^B_{bdis} m_{bdik} \qquad d\in\mathcal{D}, i\in\mathcal{I}_d, b\in\mathcal{B}_i, k\in\mathcal{K}, s\in\mathcal{S} \tag{5.9}$$

Constraints (5.8) state that feed compositions only use silage bales from the kitchen they are mixed. Constraints (5.9) ensure that the amount of dry matter used from a silage bale does not exceed the actual dry matter content of the bale. The big-M values for Constraints (5.8) are set to the maximum amount of dry matter that can be used from a given bale in a given feed composition in a given kitchen. This is either limited by the maximum amount of dry matter in the feed composition or by the dry matter content in the specific bale. This is formulated as:

$$M_1 : M_{adibs} = \min\{U^M_{ad}, F^B_{bdis}\}$$

Feed Construction Constraints

$$\sum_{k\in\mathcal{K}}\sum_{i\in\mathcal{I}_d}\sum_{b\in\mathcal{B}_i} x_{abdiks} + \sum_{f\in\mathcal{F}} F^F_f y_{adfs} + r_{ads} = T^M_{ad} + d^M_{ads} \qquad a\in\mathcal{A}, d\in\mathcal{D}, s\in\mathcal{S} \quad (5.10)$$

$$L^M_{ad} \leq T^M_{ad} + d^M_{ads} \leq U^M_{ad} \qquad a\in\mathcal{A}, d\in\mathcal{D}, s\in\mathcal{S} \quad (5.11)$$

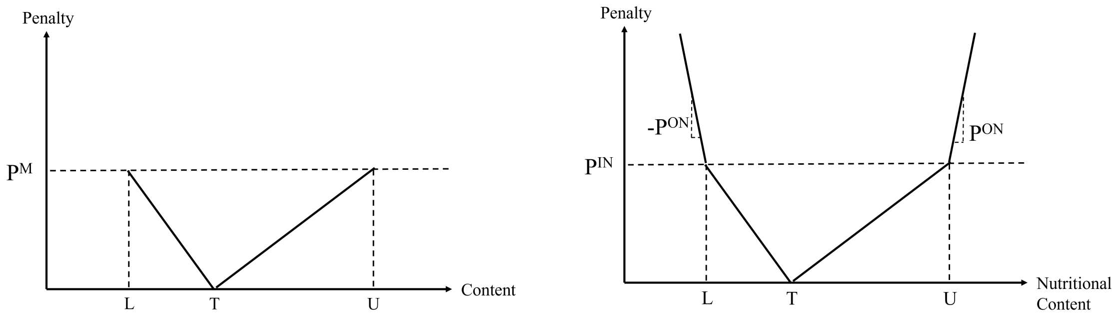
$$\sum_{k\in\mathcal{K}}\sum_{i\in\mathcal{I}_d}\sum_{b\in\mathcal{B}_i} N^B_{bdns} x_{abdiks} + \sum_{f\in\mathcal{F}} N^F_{fn} F^F_f y_{adfs} + N^R_n r_{ads} = T^N_{adn} + d^N_{adns} \qquad a\in\mathcal{A}, d\in\mathcal{D}, n\in\mathcal{N}, s\in\mathcal{S} \quad (5.12)$$

$$\sum_{f\in\mathcal{F}} F^F_f y_{adfs} \leq F^{MAX}_a T^M_{ad} \qquad a\in\mathcal{A}, d\in\mathcal{D}, s\in\mathcal{S} \quad (5.13)$$

Constraints (5.10) set the deviation from the target for dry matter content, and Constraints (5.11) ensure that the dry matter content in a feed composition is within its lower and upper limit in all scenarios. Constraints (5.12) set the deviation from target in nutritional content. Lastly, Constraints (5.13) ensure that the amount of feed concentrate used in a feed composition is lower than its maximum limit.

Penalty Constraints

As described in Chapter 4, all animals have requirements regarding nutritional and dry matter content, and the further the content is from the target value, the higher the penalty is. Figure 5.2 illustrate the penalty functions for deviating from the target for dry matter and nutrients. As the figures show, the penalty increases linearly as the content deviates from the target, until it reaches P^M and P^{IN} at the limits for dry matter and nutrients, respectively. The slope of the penalty curve depends on the difference between the target and the upper or lower limit. This is based on the assumption that it is worse to deviate from the limit if the allowed interval is narrow. For nutrients, the upper and lower limits are not strict. However, as shown in Figure 5.2b, when the content reaches its upper or lower bound, the growth rate of the penalty increases to a rate of P^{ON}.



(a) Penalty function for deviating from target in dry matter content.

(b) Penalty function for deviating from target in nutritional content.

Figure 5.2: Illustration of the penalty functions used for penalizing deviation from target in dry matter and nutritional content in a feed composition.

As the penalty functions are convex and piecewise linear, they are described using linear inequalities:

$$p^M_{ads} \geq \frac{P^M d^M_{ads}}{U^M_{ad} - T^M_{ad}} \qquad a \in \mathcal{A}, d \in \mathcal{D}, s \in \mathcal{S} \quad (5.14)$$

$$p^M_{ads} \geq \frac{P^M d^M_{ads}}{L^M_{ad} - T^M_{ad}} \qquad a \in \mathcal{A}, d \in \mathcal{D}, s \in \mathcal{S} \quad (5.15)$$

$$p^N_{adns} \geq \frac{P^{IN} d^N_{adns}}{U^N_{adn} - T^N_{adn}} \qquad a \in \mathcal{A}, d \in \mathcal{D}, n \in \mathcal{N}\ s \in \mathcal{S} \quad (5.16)$$

$$p^N_{adns} \geq \frac{P^{IN} d^N_{adns}}{L^N_{adn} - T^N_{adn}} \qquad a \in \mathcal{A}, d \in \mathcal{D}, n \in \mathcal{N}, s \in \mathcal{S} \quad (5.17)$$

$$p^N_{adns} \geq P^{IN} + P^{ON}(T^N_{adn} + d^N_{adns} - U^N_{adn}) \qquad a \in \mathcal{A}, d \in \mathcal{D}, n \in \mathcal{N}, s \in \mathcal{S} \quad (5.18)$$

$$p^N_{adns} \geq P^{IN} - P^{ON}(T^N_{adn} + d^N_{adns} - L^N_{adn}) \qquad a \in \mathcal{A}, d \in \mathcal{D}, n \in \mathcal{N}, s \in \mathcal{S} \quad (5.19)$$

The penalties for deviating from the dry matter target are set in Constraints (5.14) and (5.15). Constraints (5.16) - (5.19) set the penalties for deviating from target for the different nutrients. Constraints (5.16) and (5.17) set the penalties when the content is between the target and upper or lower bound, while Constraints (5.18) and (5.19) set the penalties for being outside the upper or lower limits.

Variability Constraints

$$v^M_{a(d-1)ds} \geq \frac{d^M_{ads}}{T^M_{ad}} - \frac{d^M_{a(d-1)s}}{T^M_{a(d-1)}} \qquad a \in \mathcal{A}, d \in \mathcal{D} \setminus \{1\}, s \in \mathcal{S} \quad (5.20)$$

$$v^M_{a(d-1)ds} \geq \frac{d^M_{a(d-1)s}}{T^M_{a(d-1)}} - \frac{d^M_{ads}}{T^M_{ad}} \qquad a \in \mathcal{A}, d \in \mathcal{D} \setminus \{1\}, s \in \mathcal{S} \quad (5.21)$$

$$v^N_{a(d-1)dns} \geq \frac{d^N_{adns}}{T^N_{adn}} - \frac{d^N_{a(d-1)ns}}{T^N_{a(d-1)n}} \qquad a \in \mathcal{A}, d \in \mathcal{D} \setminus \{1\}, n \in \mathcal{N}, s \in \mathcal{S} \quad (5.22)$$

$$v^N_{a(d-1)dns} \geq \frac{d^N_{a(d-1)ns}}{T^N_{a(d-1)n}} - \frac{d^N_{adns}}{T^N_{adn}} \qquad a \in \mathcal{A}, d \in \mathcal{D} \setminus \{1\}, n \in \mathcal{N}, s \in \mathcal{S} \quad (5.23)$$

$$v^M_{a01s} \geq \frac{d^M_{a1s}}{T^M_{a1}} - R^M_a \qquad a \in \mathcal{A}, s \in \mathcal{S} \quad (5.24)$$

$$v^M_{a01s} \geq R^M_a - \frac{d^M_{a1s}}{T^M_{a1}} \qquad a \in \mathcal{A}, s \in \mathcal{S} \quad (5.25)$$

$$v^N_{a01ns} \geq \frac{d^N_{a1ns}}{T^N_{a1n}} - R^N_{an} \qquad a \in \mathcal{A}, n \in \mathcal{N}, s \in \mathcal{S} \quad (5.26)$$

$$v^N_{a01ns} \geq R^N_{an} - \frac{d^N_{a1ns}}{T^N_{a1n}} \qquad a \in \mathcal{A}, n \in \mathcal{N}, s \in \mathcal{S} \quad (5.27)$$

Constraints (5.20)-(5.23) set the relative variation from one feed composition to another in terms of dry matter and nutritional content and ensure that this variation is set to the absolute value. Constraints (5.24)-(5.27) set the deviation for dry matter and nutritional content at the first day of the planning period.

Variable Constraints

$$x_{abdiks} \geq 0 \qquad a \in \mathcal{A}, d \in \mathcal{D}_i, i \in \mathcal{I}_d, b \in \mathcal{B}_i, k \in \mathcal{K}, s \in \mathcal{S} \quad (5.28)$$

$$y_{adfs} \geq 0 \qquad a \in \mathcal{A}, d \in \mathcal{D}, f \in \mathcal{F}, s \in \mathcal{S} \quad (5.29)$$

$$r_{ads},\ p^M_{ads},\ v^M_{a(d-1)ds} \geq 0 \qquad a \in \mathcal{A}, d \in \mathcal{D}, s \in \mathcal{S} \quad (5.30)$$

$$p^N_{adns},\ v^N_{a(d-1)dns} \geq 0 \qquad a \in \mathcal{A}, d \in \mathcal{D}, n \in \mathcal{N}, s \in \mathcal{S} \quad (5.31)$$

$$m_{bdik} \in \{0,1\} \qquad i \in \mathcal{I}, b \in \mathcal{B}_i, d \in \mathcal{D}_\rangle, k \in \mathcal{K} \quad (5.32)$$

$$w_{adk} \in \{0,1\} \qquad a \in \mathcal{A}, d \in \mathcal{D}, k \in \mathcal{K} \quad (5.33)$$

$$d^M_{ads} \ \ free \qquad a \in \mathcal{A}, d \in \mathcal{D}, s \in \mathcal{S} \quad (5.34)$$

$$d^N_{adns} \ \ free \qquad a \in \mathcal{A}, d \in \mathcal{D}, n \in \mathcal{N}, s \in \mathcal{S} \quad (5.35)$$

Constraints (5.28)-(5.35) are variable constraints. Constraint (5.28) - (5.31) are non-negativity constraints, while Constraints (5.32)-(5.33) ensure binary variables. Lastly, Constraints (5.34)-(5.35) allows the variables to take any value.

Chapter 6

Solution- and Evaluation Methods

This chapter introduces the methods used to solve and evaluate the mathematical model presented in Chapter 5. First, Section 6.1 provides an overview of the L-shaped method. Next, Section 6.2 proposes an L-shaped decomposition algorithm for the SCFPP based on the Benders reformulation of the problem, and introduces the algorithm used to implement the model. In Section 6.3, several acceleration techniques that aim at improving the efficiency of the L-shaped method are presented. A method for reducing the complexity of the problem is presented in Section 6.4. Lastly, a simulation-based evaluation approach is presented in Section 6.5. By the end of this chapter, readers should have a comprehensive understanding of the method used to solve the SCFPP problem, and the techniques implemented to accelerate the solution algorithm. Furthermore, they should have insight into the method used to evaluate the model.

6.1 L-Shaped Method

The SCFPP, as presented in Chapter 5, has a characteristic structure known as the dual block-angular structure, illustrated in Figure 6.1. This structure enables each scenario to be solved independently of the others, given the initial decisions made. Therefore, it is well-suited for applying Benders decomposition, a reformulation and decomposition technique utilized to solve large linear programs. In the case of stochastic programs, the Benders decomposition method is called the L-shaped method (Van Slyke & Wets, 1969).

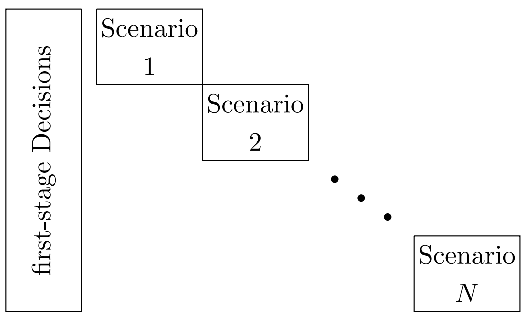


Figure 6.1: Illustration of dual block angular structure.

The L-shaped method is a cutting plane technique. The main idea is to approximate the recourse function in the objective. By doing this, the model avoids multiple function evaluations for the recourse function and consequently reduces the computational solution time (Birge & Louveaux, 2011). The decomposition is done by establishing a master problem, where the recourse function only is evaluated exactly as a subproblem. The L-shaped algorithm iterates between the master- and the subproblem until a satisfactory solution is found. For each time the second-stage problem, i.e. the subproblem, is solved, a feasibility cut and/or an optimality cut is added to the master problem. This method enables the model to solve smaller problems in each iteration, where only the necessary restrictions are included. If there exists a solution in each subproblem for all possible solutions in the master problem, then the problem has *relatively complete recourse*. This means that there is no need for feasibility cuts, as a feasible solution always will be found.

To utilize the L-shaped method, the problem is required to have a finite number of scenarios. Given that real-life probabilities often follow continuous distributions, it is common practice to sample a finite set of scenarios from these distributions, aiming to ensure that the selected scenarios are representative of the underlying distribution. As stated in Chapter 5, this approach is also used when solving the SCFPP, where the distribution is presented in Chapter 7.

6.2 Applying the L-shaped Method to the SCFPP

This section presents the application of the L-shaped method to the SCFPP. First, the Benders reformulation of the mathematical model is presented. Section 6.2.1 and Section 6.2.2 present the sub- and master-problem of the formulation. The algorithm used for implementation is presented in Section 6.2.3 and lastly, Section 6.2.4 introduces the termination criteria used in the implementation.

In the decomposition of the SCFPP, the master problem consists of the first-stage decisions presented in Chapter 5. The subproblems deal with the second-stage decisions and are solved after the actual content of the silage bales is known. The decisions in the different subproblems are independent of each other. Figure 6.2 gives an overview of the decisions that are made in the master- and subproblems.

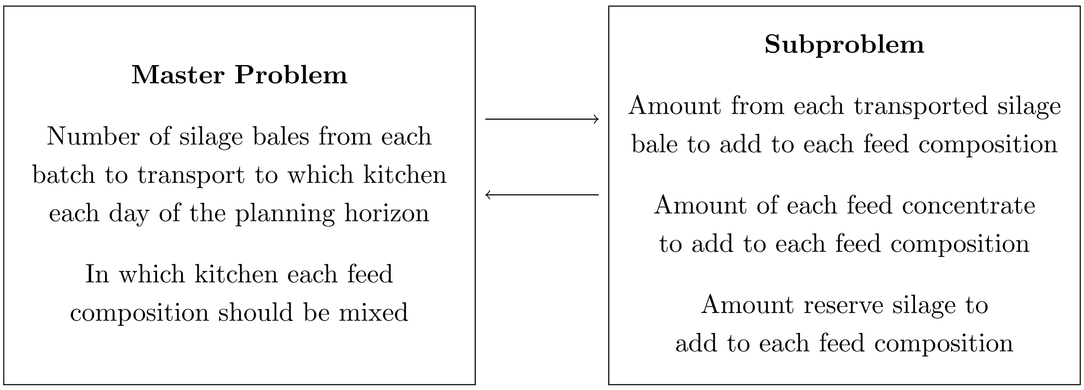


Figure 6.2: Overview of the decisions taken in the master- and subproblem when solving the problem using the L-shaped method.

6.2.1 Subproblem

The subproblem corresponds to the second-stage problem in the mathematical model presented in Chapter 5. Each subproblem is independent of all other subproblems and represents a scenario

$s \in \mathcal{S}$. The presented formulation represents a general formulation for any scenario. The optimal first-stage decisions in the current iteration is used as input in the subproblem as $\hat{m}_{bdik}$ and $\hat{w}_{adk}$, where $\hat{m}_{bdik}$ represent whether or not a silage bale is transported to kitchen k on day d and $\hat{w}_{adk}$ represents the choices related to where each feed composition should be made. The objective function for the quality and cost objectives is given by Equation (6.1) and Equation (6.2), respectively. Equation (6.3) presents the weighted-sum objective function that is used in the subproblem.

Quality Objective for Subproblem

$$f_s^Q = \min \sum_{a \in \mathcal{A}} \sum_{d \in \mathcal{D}} ((p_{ads}^M + \sum_{n \in \mathcal{N}} p_{adns}^N) + (v_{a(d-1)ds}^M + \sum_{n \in \mathcal{N}} v_{a(d-1)dns}^N)) \tag{6.1}$$

Cost Objective for Subproblem

$$f_s^C = \min \sum_{a \in \mathcal{A}} \sum_{d \in \mathcal{D}} (\sum_{f \in \mathcal{F}} C_f^F y_{adfs} + C^R r_{ads}) \tag{6.2}$$

Weighted-Sum Objective

$$Z_s^{SP} = \min \ \alpha f_s^Q + (1-\alpha) f_s^C \tag{6.3}$$

Constraints

$$x_{abdiks} \leq M_1 \hat{w}_{adk} \qquad a \in \mathcal{A}, d \in \mathcal{D}, i \in \mathcal{I}_d, b \in \mathcal{B}_i, k \in \mathcal{K} \tag{6.4}$$

$$\sum_{a \in \mathcal{A}} x_{abdiks} \leq F_{bdis}^B \hat{m}_{bdik} \qquad d \in \mathcal{D}, i \in \mathcal{I}_d, b \in \mathcal{B}_i, k \in \mathcal{K} \tag{6.5}$$

Constraints (6.4)-(6.5) correspond to Constraints (5.8)-(5.9) in the original formulation. In the subproblem, these constraints are formulated with the first-stage solutions held fixed. We introduce the dual variables of Constraints (6.4) in L-shaped iteration j as α_{abdiks}^j and the dual variables of Constraints (6.5) in L-shaped iteration j as β_{bdiks}^j. These dual variables are used in the formulation of the master problem, specifically in the formulation of the optimality cut.

$$\sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{I}_d} \sum_{b \in \mathcal{B}_i} x_{abdiks} + \sum_{f \in \mathcal{F}} F_f^F y_{adfs} + r_{ads} - d_{ads}^M = T_{ad}^M \qquad a \in \mathcal{A}, d \in \mathcal{D} \tag{6.6}$$

$$L_{ad}^M \leq d_{ads}^M + T_{ad}^M \leq U_{ad}^M \qquad a \in \mathcal{A}, d \in \mathcal{D} \tag{6.7}$$

$$\sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{I}_d} \sum_{b \in \mathcal{B}_i} N_{bdns}^B x_{abdiks} + \sum_{f \in \mathcal{F}} N_{fn}^F y_{adfs} + N_n^R r_{ads}^N - d_{adns}^N = T_{adn}^N \qquad a \in \mathcal{A}, d \in \mathcal{D}, n \in \mathcal{N} \tag{6.8}$$

$$\sum_{f \in \mathcal{F}} F_f^F y_{adfs} \leq F_a^{MAX} T_{ad}^M \qquad a \in \mathcal{A}, d \in \mathcal{D} \tag{6.9}$$

$$p^M_{ads} - \frac{P^M d^M_{ads}}{U^M_{ad} - T^M_{ad}} \geq 0 \qquad a \in \mathcal{A}, d \in \mathcal{D} \quad (6.10)$$

$$p^M_{ads} - \frac{P^M d^M_{ads}}{L^M_{ad} - T^M_{ad}} \geq 0 \qquad a \in \mathcal{A}, d \in \mathcal{D} \quad (6.11)$$

$$p^N_{adns} - \frac{P^{IN} d^N_{adns}}{U^N_{adn} - T^N_{adn}} \geq 0 \qquad a \in \mathcal{A}, d \in \mathcal{D}, n \in \mathcal{N} \quad (6.12)$$

$$p^N_{adns} - \frac{P^{IN} d^N_{adns}}{L^N_{adn} - T^N_{adn}} \geq 0 \qquad a \in \mathcal{A}, d \in \mathcal{D}, n \in \mathcal{N} \quad (6.13)$$

$$p^N_{adns} - P^{ON} d^N_{adns} \geq P^{IN} + P^{ON}(T^N_{adn} - U^N_{adn}) \qquad a \in \mathcal{A}, d \in \mathcal{D}, n \in \mathcal{N} \quad (6.14)$$

$$p^N_{adns} + P^{ON} d^N_{adns} \geq P^{IN} - P^{ON}(T^N_{adn} + L^N_{adn}) \qquad a \in \mathcal{A}, d \in \mathcal{D}, n \in \mathcal{N} \quad (6.15)$$

$$v^M_{a(d-1)ds} - (\frac{d^M_{ads}}{T^M_{ad}} - \frac{d^M_{a(d-1)s}}{T^M_{a(d-1)}}) \geq 0 \qquad a \in \mathcal{A}, d \in \mathcal{D} \setminus \{1\} \quad (6.16)$$

$$v^M_{a(d-1)ds} - (\frac{d^M_{a(d-1)s}}{T^M_{a(d-1)}} - \frac{d^M_{ads}}{T^M_{ad}}) \geq 0 \qquad a \in \mathcal{A}, d \in \mathcal{D} \setminus \{1\} \quad (6.17)$$

$$v^N_{a(d-1)dns} - (\frac{d^N_{adns}}{T^N_{adn}} - \frac{d^N_{a(d-1)ns}}{T^N_{a(d-1)n}}) \geq 0 \qquad a \in \mathcal{A}, d \in \mathcal{D} \setminus \{1\}, n \in \mathcal{N} \quad (6.18)$$

$$v^N_{a(d-1)dns} - (\frac{d^N_{a(d-1)ns}}{T^N_{a(d-1)n}} - \frac{d^N_{adns}}{T^N_{adn}}) \geq 0 \qquad a \in \mathcal{A}, d \in \mathcal{D} \setminus \{1\}, n \in \mathcal{N} \quad (6.19)$$

$$v^M_{a01s} - \frac{d^M_{a1s}}{T^M_{a1}} \geq -R^M_a \qquad a \in \mathcal{A} \quad (6.20)$$

$$v^M_{a01s} + \frac{d^M_{a1s}}{T^M_{a1}} \geq R^M_a \qquad a \in \mathcal{A} \quad (6.21)$$

$$v^N_{a01ns} - \frac{d^N_{a1ns}}{T^N_{a1n}} \geq -R^N_{an} \qquad a \in \mathcal{A}, n \in \mathcal{N} \quad (6.22)$$

$$v^N_{a01ns} + \frac{d^N_{a1ns}}{T^N_{a1n}} \geq R^N_{an} \qquad a \in \mathcal{A}, n \in \mathcal{N} \quad (6.23)$$

Constraints (6.6)-(6.23) correspond to Constraints (5.10)-(5.27) in the original problem and does not contain any first-stage decision variables. The difference from the original model formulation is that the scenario s is set in each of the subproblems, and the constraints are consequently only applicable to the current scenario.

$$x_{abdiks} \geq 0 \qquad a \in \mathcal{A}, d \in \mathcal{D}, i \in \mathcal{I}_d, b \in \mathcal{B}_i, k \in \mathcal{K} \quad (6.24)$$

$$r_{ads},\ p^M_{ads},\ v^M_{a(d-1)ds} \geq 0 \qquad a \in \mathcal{A}, d \in \mathcal{D} \quad (6.25)$$

$$y_{adfs} \geq 0 \qquad a \in \mathcal{A}, d \in \mathcal{D}, f \in \mathcal{F} \quad (6.26)$$

$$p^N_{adns},\ v^N_{a(d-1)dns} \geq 0 \qquad a \in \mathcal{A}, d \in \mathcal{D}, n \in \mathcal{N} \quad (6.27)$$

$$d^M_{ads} \ \ free \qquad a \in \mathcal{A}, d \in \mathcal{D} \quad (6.28)$$

$$d^N_{adns} \ \ free \qquad a \in \mathcal{A}, d \in \mathcal{D}, n \in \mathcal{N} \quad (6.29)$$

Constraints (6.24)-(6.29) are variable constraints.

6.2.2 Master Problem

The objective of the master problem consists of two parts. Firstly, it includes the costs associated with the first-stage decisions. Secondly, it incorporates an estimated second-stage cost, represented by the variable θ. This allows the master problem to consider the impact of the initial decisions on the second-stage solutions, leading to more informed decision-making. The objective of our master problem is formulated in Equation (6.30).

$$Z^{MP} = \min \sum_{i \in \mathcal{I}^M} \sum_{d \in \mathcal{D}_i} \sum_{b \in \mathcal{B}_i} \sum_{k \in \mathcal{K}} C^B_i m_{bdik} + \theta \qquad (6.30)$$

Constraints

$$\sum_{k \in \mathcal{K}} w_{adk} = 1 \qquad a \in \mathcal{A}, d \in \mathcal{D} \quad (6.31)$$

$$\sum_{d \in \mathcal{D}_i} \sum_{k \in \mathcal{K}} m_{bdik} \leq 1 \qquad i \in \mathcal{I}, b \in \mathcal{B}_i \quad (6.32)$$

$$\sum_{d'=1}^{d} \sum_{k \in \mathcal{K}} (m_{(b-1)d'ik} - m_{bd'ik}) \geq 0 \qquad d \in \mathcal{D}, i \in \mathcal{I}_d, b \in \mathcal{B}_i \setminus \{1\} \quad (6.33)$$

$$\sum_{i \in \mathcal{I}_d} \sum_{b \in \mathcal{B}_i} m_{bdik} \leq K_k \qquad d \in \mathcal{D}, k \in \mathcal{K} \quad (6.34)$$

Constraints (6.31)-(6.34) of the master problem correspond to Constraints (5.4)-(5.7) in the original formulation.

$$\begin{aligned} \theta \geq \sum_{s \in \mathcal{S}} P_s (& \sum_{a \in \mathcal{A}} \sum_{d \in \mathcal{D}} \sum_{i \in \mathcal{I}_d} \sum_{b \in \mathcal{B}_i} \sum_{k \in \mathcal{K}} M_1 w_{adk} \alpha^j_{abdiks} \\ & + \sum_{d \in \mathcal{D}} \sum_{i \in \mathcal{I}_d} \sum_{b \in \mathcal{B}_i} \sum_{k \in \mathcal{K}} F^B_{bdis} m_{bdik} \beta^j_{bdiks}) \\ & + W^j \end{aligned} \qquad (6.35)$$

Each iteration of the subproblems adds a new optimality cut, denoted by (6.35), which restricts the value of θ. This constraint approximates the second-stage objective value given the first-stage decisions. The right-hand side of the constraint is computed as a sum of the dual values to the constraints associated with the first-stage variables, i.e. Constraints (6.4)-(6.5), multiplied by their corresponding first-stage variables. The constant W^j includes all the constant values involved in the calculation, and is given by the following expression:

$$W^j = \sum_{s \in \mathcal{S}} P_s (Z_s^{SP} + (\pi_s^j)^T T_s x^j) \qquad (6.36)$$

where Z_s^{SP} is the objective function value in scenario s, π_s^j is the dual values of the solution in scenario s and T_s is the constant matrix for the first-stage values in scenario s. In Equation (6.36), x^j is the first-stage decisions, which in the SCFPP is represented by the variables w_{adk} and m_{bdik}.

$$m_{bdik} \in \{0, 1\} \qquad i \in \mathcal{I}, b \in \mathcal{B}_i, d \in \mathcal{D}, k \in \mathcal{K} \quad (6.37)$$

$$w_{adk} \in \{0, 1\} \qquad a \in \mathcal{A}, d \in \mathcal{D}, k \in \mathcal{K} \quad (6.38)$$

$$\theta \geq 0 \qquad (6.39)$$

Constraints (6.37)-(6.39) are variable constraints.

6.2.3 Algorithmic Implementation

Algorithm 1 presents the L-shaped algorithmic implementation of the reformulation. The algorithm starts by building the master and subproblems and initializing the lower and upper bounds. In each iteration, the master problem is solved, before the subproblem is solved with fixed first-stage decisions. Then, the upper and lower bound is updated. The algorithm terminates if a termination criterion is reached, discussed further in Section 6.2.4. Otherwise, an optimality cut is added to the master problem, and the algorithm continues.

On the first iteration of the master problem, there are no optimality cuts. Each time the subproblem is solved, one optimality cut is added to the master problem, as long as a termination criterion is not reached. This formulation is a single-cut formulation of the problem since only one optimality cut is added each time the subproblem is solved.

6.2.4 Termination Criteria

In the L-shaped algorithm proposed by Birge and Louveaux (2009), the termination criterion is when the lower bound equals the upper bound, i.e. when the optimal solution is found. However, with large problems such as the SCFPP, many iterations may be needed before this is the case. Therefore, to reduce the number of iterations, a common termination criterion is to stop the algorithm when the gap between the lower- and upper bound is below a given value. Equations (6.40) and (6.41) present the calculation of the upper and lower bound, respectively. The lower bound is the current best objective value for the master problem. The upper bound is the minimum

Algorithm 1 Proposed algorithm

```
procedure L-SHAPED METHOD FOR SCFPP
    Build master problem
    Build a subproblem for each scenario s ∈ S
    LB ← −∞, UB ← +∞, Continue ← True, i ← 0
    while Continue do
        i ← i + 1
        Solve master problem to optimality
        for all s ∈ S do
            Solve subproblem s with given first-stage decisions
        end for
        Update LB and UB
        if Termination criteria is reached then
            Continue ← False
        else
            Add optimality cut to master problem
        end if
    end while
    output Solved master problem
end procedure
```

of the existing upper bound and the first-stage costs plus the actual total second-stage costs from the subproblems.

$$LB = Z^{MP} \tag{6.40}$$

$$UB = \min\{UB,\ Z^{MP} - \theta + \sum_{s \in \mathcal{S}} P_s Z_s^{SP}\} \tag{6.41}$$

Using this, the bounds are evaluated in every iteration. When the gap between upper and lower bound is smaller than a predetermined value, or the maximum time limit or iteration limit is reached, the algorithm is terminated.

6.3 Acceleration Techniques to the L-Shaped Method

The L-shaped method can be computationally heavy, particularly for larger instances. Furthermore, when the number of scenarios is large, the number of iterations required to obtain a good solution can be very high. This can lead to a long computational time, especially for realistic problem sizes. Therefore, several acceleration methods are proposed to improve the efficiency and convergence of the L-shaped method, such as a multi-cut version, approximate master solve, and two warm start approaches. These methods aim to speed up the overall solution process. In the context of solving the SCFPP, these techniques are discussed and implemented to improve the computational efficiency of the L-shaped method. It should be noted that all the following acceleration techniques are presented using the single-cut version of the L-shaped method unless stated otherwise.

6.3.1 Multi-Cut Version of the L-Shaped Approach

The generalized Benders optimality cut for the mathematical model in Section 6.2, presented in Equation (6.35) is a single-cut version. In this cut, the optimal simplex multipliers are aggregated together. However, the structure of stochastic programs allows for adding several cuts instead of one. This results in multiple rows being added to the master problem for each iteration of the subproblem, increasing the amount of information in the master problem. Birge and Louveaux (1988) introduce a Benders decomposition multi-cut version where they use outer approximations of each subproblem. The main goal of this approach is that multiple optimality cuts give more information than a single cut and result in fewer iterations needed for convergence. However, neither the single-cut version nor the multi-cut version of Benders optimality cuts is superior to the other in all circumstances. The balance between the increased size of the master problem and the reduced number of iterations of the subproblem is problem-dependent. Birge and Louveaux (1988) prove that the multi-cut approach can be expected to be especially efficient for problems where the subproblem is large, multiple optimality cuts are needed and the number of scenarios is not larger than the number of variables in the master problem.

The multi-cut constraints for SCFPPs mathematical model are presented in Restrictions (6.42). In this case, there is one θ_s for each scenario. The algorithm for the multi-stage approach is equal to 1, but the optimality cut in Equation (6.35) is replaced with Equations (6.42). Furthermore, the objective function in the master problem has to accumulate the θ_s for all scenarios to get an approximation of the second-stage cost. The θ in the objective function is therefore replaced with $\sum_{s\in\mathcal{S}} P_s\theta_s$.

$$\begin{aligned}\theta_s \geq & \sum_{a\in\mathcal{A}}\sum_{d\in\mathcal{D}}\sum_{i\in\mathcal{I}_d}\sum_{b\in\mathcal{B}_i}\sum_{k\in\mathcal{K}} M_1 w_{adk}\alpha^j_{abdiks} \\ & + \sum_{d\in\mathcal{D}}\sum_{i\in\mathcal{I}_d}\sum_{b\in\mathcal{B}_i}\sum_{k\in\mathcal{K}} F^B_{bdis} m_{bdik}\beta^j_{bdiks} \\ & + W^j_s \end{aligned} \qquad s\in\mathcal{S} \quad (6.42)$$

6.3.2 Warm Start Techniques

Warm start can be a useful technique to reduce the computational effort required for a problem solved with the L-shaped method (Kuudela & Popela, 2017). There exists multiple different warm start strategies in the literature (Rahmaniani et al., 2017), but in the context of the SCFPP we present and implement two different approaches, namely a Robust Warm start (RW) approach and an Initializing Warm start (IW) approach. These two approaches differ in the way that the first one aims at warm starting the first iteration of the algorithm, while the second approach aims at warm starting the master problem in each iteration. The main goal for both methods is to get a good description of the master problem in a short amount of time.

The RW method solves a deterministic problem before solving the problem with the L-shaped method. The deterministic problem is solved with the worst-case scenario. In the context of the SCFPP, this means a scenario where all the silage bales are assigned the lowest quality possible from the scenarios in the stochastic problem. The solution obtained is used as an initial first-stage solution in the subproblems. The algorithm for the RW method is presented in 2, where the change from 1 is that the first iteration in the stochastic problem uses the first-stage solutions from the robust optimization problem.

Algorithm 2 Proposed algorithm

```
procedure L-shaped method for SCFPP with Warm Start
    Lines (2)-(4) of Algorithm 1
    Solve the robust optimization problem
    for all s ∈ S do
        Fix first-stage variables in subproblem s from the deterministic problem
        Solve subproblem s
    end for
    Add optimality cut to master problem
    Lines (5)-(17) of Algorithm 1
end procedure
```

The second commonly employed technique for warm start in the L-shaped method involves initializing the master problem with the solution obtained from the previous iteration. This approach enables the master problem to leverage the information already gathered and potentially speed up the discovery of an optimal solution, thereby accelerating the overall problem-solving process, as discussed in Rodríguez et al. (2021).

6.3.3 A Two-Phase Approach

McDaniel and Devine (1977) introduces a modified algorithm for solving mixed integer problems with the L-shaped method. They show that valid optimality cuts can be obtained from the solution to the Linear Programming (LP) relaxation of the master problem. To take advantage of this, they apply the decomposition algorithm in two phases. In the first phase, they solve the linear relaxation of the master problem, which enables them to generate initial solutions rapidly and tighten the relaxation. Then, in the second phase, they reintroduce the integrality requirements to the master problem, and the solution process continues. By using this Two-Phase (TP) approach, McDaniel and Devine (1977) demonstrate that they are able to solve mixed integer programming problems more efficiently and effectively than prior methods. As the master problem in the SCFPP is a mixed integer problem, the TP approach can be applied to faster get a good description of the master problem and consequently converge more rapidly toward the optimal solution.

The effectiveness of the TP approach introduced by McDaniel and Devine (1977) depends on when the transition from the first to the second phase is done. In the original paper, the authors propose three different methodologies for determining this point. The first method involves continuing the LP iterations until no further iterations were possible. The second method is to solve the master problem as an LP for the first k iterations. Finally, the third method is to switch to the original master problem when the upper bound is less than or equal to the lower bound plus a small epsilon value.

For the purpose of this report, we adopt the third method proposed by McDaniel and Devine (1977). By using this method we avoid using too much time solving the linear relaxation, as method one might. Furthermore, the third method guarantees that a ϵ-optimal solution is found, which is not the case for method two, where the LP solution might have a very high gap. We consequently expect to achieve a balance between solution time and solution quality.

Note that when the integrality property is added to the master problem, the upper bound is set to ∞ while the lower bound is unchanged. This is because the LP-relaxation of the master problem

Algorithm 3 Proposed algorithm

procedure L-Shaped Method for SCFPP with a Two-Phase Approach
    Build **master problem** as a linear problem
    Line (3)-(17) in Algorithm 1
    if Termination criteria is reached **then**
        Introduce integrality property to the **master problem**
        $UB \leftarrow \infty$
        Line (5)-(17) in Algorithm 1
    end if
end procedure

generates a valid lower bound, but an invalid upper bound.

6.3.4 The Magnanti-Wong Method

The effectiveness of the L-shaped method relies on the quality of the optimality cuts it generates. In their work, Magnanti and Wong (1981) highlight that the degenerate primal subproblem yields optimality cuts with varying degrees of strength. To address this issue, they introduce a method for identifying a Pareto-optimal cut, which refers to a cut that is not dominated by any other cut. This technique involves utilizing core points which are points residing within the relative interior of the convex hull of the feasible area. The method solves the master problem and the subproblem in the same manner as the original L-shaped method. However, after solving a subproblem, the method distinguishes from the original method by solving the Magnanti-Wong subproblem for the same scenario in a chosen core point and by using the dual values from the Magnanti-Wong subproblem to generate an optimality cut. The Magnanti-Wong subproblem is also possible to formulate on dual form. This formulation is similar to the original dual subproblem, but includes a Magnanti-Wong constraint that ensures that the objective value is equal to the optimal objective value found in the original subproblem. The dual variable of this constraint is used in the formulation of the Magnanti-Wong primal problem.

Figure 6.3 is an illustration of the Magnanti-Wong (MW) method. In this figure, the original solution to the master problem is denoted as y, while y^0 and y^1 represent two computed core points. The dotted red lines signify the potential optimality cuts identified at point y. As depicted in the figure, multiple optimality cuts exist, and the original subproblem may not necessarily select the strongest one. By employing the Magnanti-Wong subproblem with a core point, the Pareto optimal cut, that is specific to that core point, is determined. However, Figure 6.3 demonstrates that the strongest cut varies for different core points. For core point y^0, the Pareto-optimal cut is represented by (a), while it is (b) for core point y^1.

The main goal of the MW method is to increase the information gained in each optimality cut and consequently converge faster. Even though the MW method finds the Pareto-optimal cut at each iteration, there are some drawbacks. One of the drawbacks is the need to solve two Linear Programming problems for each subproblem in each iteration. This increases the computational time used on each iteration, and might make the total computational time higher than what is needed when generating normal optimality cuts. Furthermore, the MW acceleration might suffer from numerical instability. Specifically, this is a common problem when the objective function value and the core points are of small magnitude (Perrykkad et al., 2022). Lastly, the generation of core points is not necessarily an easy task, and the generated cuts depend greatly on which core

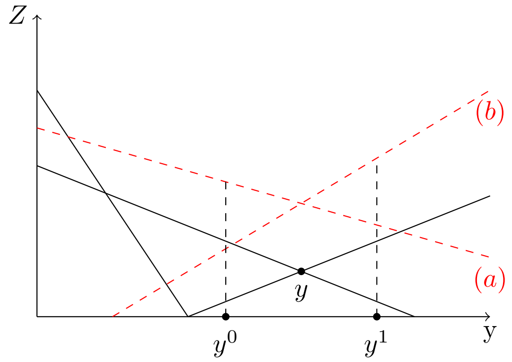


Figure 6.3: Illustration of the Magnanti-Wong method. y represent the found optimal solution for the master problem, while y^0 and y^1 represent two calculated core points. The dotted red lines, (a) and (b) represent the Pareto-optimal cuts for core point y^0 and y^1, respectively.

points are being used, as illustrated in Figure 6.3 (Magnanti & Wong, 1981).

Implementing Magnanti-Wong for the SCFPP

The objective function in the primal Magnanti-Wong subproblem is given in Equation (6.43). The difference from the original subproblem objective function is the additional term $(Z^{SP} - \epsilon)\beta$. This term comes from the Magnanti-Wong formulation, where Z^{SP} is the optimal objective value for the corresponding sub problem, ϵ is a tolerance parameter added due to numerical instability and β is the dual variable to the Magnanti-Wong constraint in the dual Magnanti-Wong subproblem.

$$Z^{MWSP} = \min \ \alpha f_s^Q + (1-\alpha) f_s^C + (Z^{SP} - \epsilon)\beta \tag{6.43}$$

The constraints with both first- and second-stage variables, i.e. Constraints (6.4)-(6.5) are reformulated in Constraints (6.44)-(6.45). The variables w^0_{adk} and m^0_{bdik} represent the core points. In the SCFPP, these are calculated as the average of the optimal solution and another feasible solution found in the current iteration of the master problem. Since both points are feasible, the average is a relative interior point unless both points are on the same facet of the integer hull.

$$x_{abdiks} \leq M_1(w^0_{adk} - \hat{w}_{adk}) \qquad a \in \mathcal{A}, d \in \mathcal{D}, i \in \mathcal{I}_d, b \in \mathcal{B}_i, k \in \mathcal{K} \tag{6.44}$$

$$\sum_{a \in \mathcal{A}} x_{abdiks} \leq F^B_{bdis}(m^0_{bdik} - \hat{m}_{bdik}) \qquad d \in \mathcal{D}, i \in \mathcal{I}_d, b \in \mathcal{B}_i, k \in \mathcal{K} \tag{6.45}$$

For all constraints without a first-stage variable, i.e. Constraints (6.6)-(6.23), the Magnanti-Wong acceleration only affects the right hand side. The change can be written as shown in Equation (6.46).

$$RHS = RHS(1-\beta) \tag{6.46}$$

Due to the numerical instability, where ϵ is added to the right hand side of the dual Magnanti-Wong constraint, β is set to be ≤ 0 in the Magnanti-Wong subproblem. This makes sure that the objective value in the original subproblem is $\geq Z_s^{SP} - \epsilon$.

Algorithm 4 Proposed algorithm

procedure L-SHAPED METHOD FOR SCFPP WITH PARETO-OPTIMAL CUTS
  Lines (2)-(8) of Algorithm 1
  Build a **Magnanti-Wong subproblem** for each scenario $s \in \mathcal{S}$
  for all $s \in \mathcal{S}$ **do**
    Solve **subproblem** s with given first-stage decisions
    Solve **Magnanti-Wong subproblem** with given first-stage decisions, core points and optimal **subproblem**-solution
  end for
  Lines (11)-(17) of Algorithm 1
end procedure

6.3.5 Approximate Master Solution

The SCFPP has integer first-stage variables, and solving the master problem to optimality can therefore become a time-consuming process. According to research by Magnanti and Wong (1981) and Zarandi (2010), the master problem can account for over 90% of the total computational time. To address this issue, modifications can be made to the general L-shaped algorithm, such as terminating the master problem before it reaches optimality. By doing so, the algorithm can send first-stage decisions to the subproblems earlier, which allows for the creation of optimality cuts and may accelerate the overall process. The main goal of the following proposed methods is to faster get a good description of the master problem, and consequently reduce the computational time.

One way to approximate the master solution is by stopping the solution process of master problem when a solution equal to the best lower bound so far is found, hereby referred to as the LB-stop method. This technique significantly reduces the computational effort required to solve the master problem, since finding a better solution that is valid is impossible. According to Rahmaniani et al. (2017), this approach can be particularly effective when used in conjunction with the L-shaped approach. They note that by exploiting the structure of the problem, the approach can reduce the computational effort required to solve the problem by up to 60%.

Another approximation method suggested by Rahmaniani et al. (2017) is the ϵ-approach. This involves solving the master problem to ϵ-optimality, allowing for a small tolerance in the objective function value each time the master problem is solved. The value of epsilon can be gradually decreased over successive iterations until the optimal solution is obtained. According to Zhao et al. (2017), the ϵ-technique can significantly reduce the number of iterations required to solve the problem, consequently reducing the total computation time.

Lastly, it is possible to accelerate the master problem for a given iteration by stopping the solution process after a certain number of solutions have been found, hereby referred to as the SolNum approach. This number can be gradually increased for each successive iteration, which can further accelerate the algorithm. The goal of this approach is to get more information from the subproblems in the form of optimality cuts, and consequently find the optimal solution in a shorter amount of time. However, this approach may require additional iterations of the algorithm to improve the solution quality.

6.4 Reducing Complexity by Limiting Uncertain Days

When dealing with uncertainty over a long planning horizon, the problem becomes increasingly complex and computationally heavy to solve. To reduce the computational complexity, while still being able to solve for a longer planning horizon, we propose a method that involves incorporating uncertainty for a subset of days in the planning horizon. While reducing the uncertainty in the problem may have a considerable impact on the solution of the model, the hypothesis behind the proposed method is that the reduced complexity may outweigh these effects. The effect of the method is studied in Chapter 8.

The proposed method is performed by splitting the planning period into two periods; with and without uncertainty. During the period including uncertainty we consider uncertainty regarding nutritional and dry matter content. For the period without uncertainty, we assume that the dry matter and nutritional content of all silage bales is equal to the expected value of the corresponding batch. This means that during the period when uncertainty is not considered, $F^B_{bdis} = \overline{F}^B_{di}$ and $N^B_{bdins} = \overline{N}^B_{bdins}$ for all bales b in batch i. The only additional input parameter that is necessary when using this approach is the chosen number of days that should take uncertainty into account.

6.5 Simulation-Based Approach for Model Evaluation

To evaluate the performance of the model, we propose a simulation-based approach that incorporates a rolling horizon method. This allows us to observe the outcome of the models' decisions in a real-life situation by simulating the quality of each silage bale over the planning horizon and allowing new information to become available as time progresses.

An illustration of the method is illustrated in Figure 6.4. In the SCFPP, the first-stage decisions are tactical decisions that are made once at the beginning of the planning horizon. In contrast, the second-stage decisions are made daily, when the actual quality is known. Therefore, the first-stage decisions are fixed based on the initial decision made at the start of the planning horizon, while the second-stage decisions are solved one day at a time. The actual quality of the silage bales transported on each day is simulated by sampling from its corresponding distribution. Using this sampled quality, the second-stage decisions are solved, taking the quality of the silage bales as well as decisions made on the previous days into account. By continuing this process for each day of the planning horizon until the entire planning horizon is solved, we can evaluate the performance of the model under a simulated reality. The objective value of the simulated model may then be compared to the objective value of the original model with scenarios.

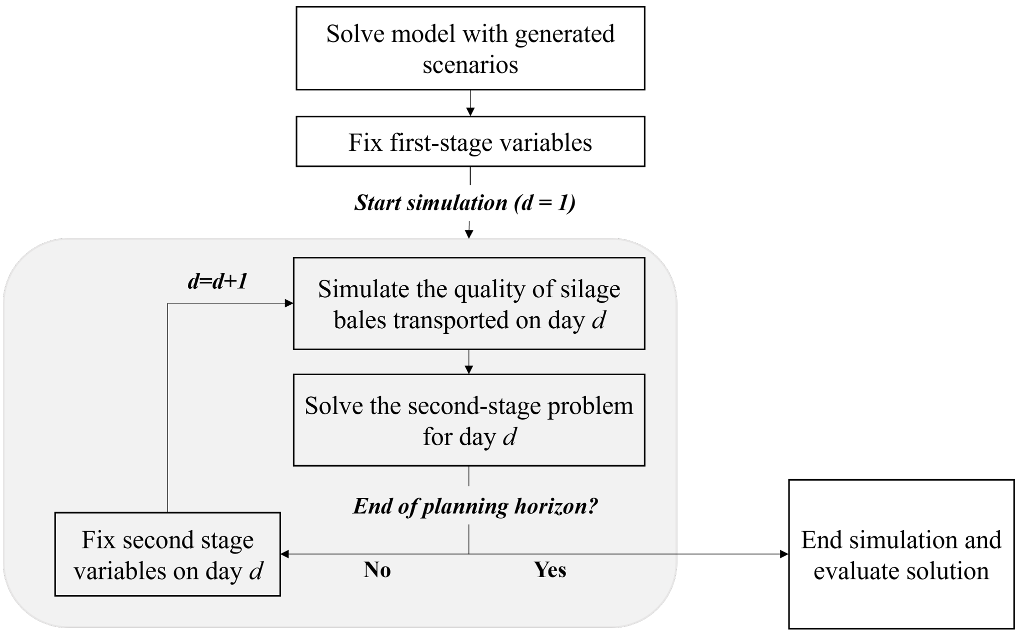


Figure 6.4: Illustration of the simulation-based approach for model evaluation.

Chapter 7

Data Sets and Parameters

This section describes the data sets and parameters used to generate instances for the computational study of the model. First, Section 7.1 describes the farm size and the animal groups considered in our test instances. Thereafter, input data related to the animal groups' feeding requirements are presented in Section 7.2. Section 7.3 presents the ingredients considered and data regarding their availability and nutritional content. Since a major part of the data used in our model is generated for the purpose of this thesis, the data may be prone to uncertainties and inaccuracies. Therefore, Section 7.4 presents some limitations to the generated data.

7.1 Farm Size and Animal Groups

In this case study, we consider a small to medium-sized farm with 30 dairy cattle, which is the average size of a Norwegian dairy farm (Statistisk sentralbyrå, 2023). The farm has one kitchen with a daily capacity of four silage bales. The animals on the farm are divided into three main categories: growing, lactating, and maintained, where each of the animal groups consists of a combination of the groups introduced in Section 2.1.2. The animal groups considered in this case study are described in Table 2.1.

Table 7.1: An overview of the animal groups considered in this case study.

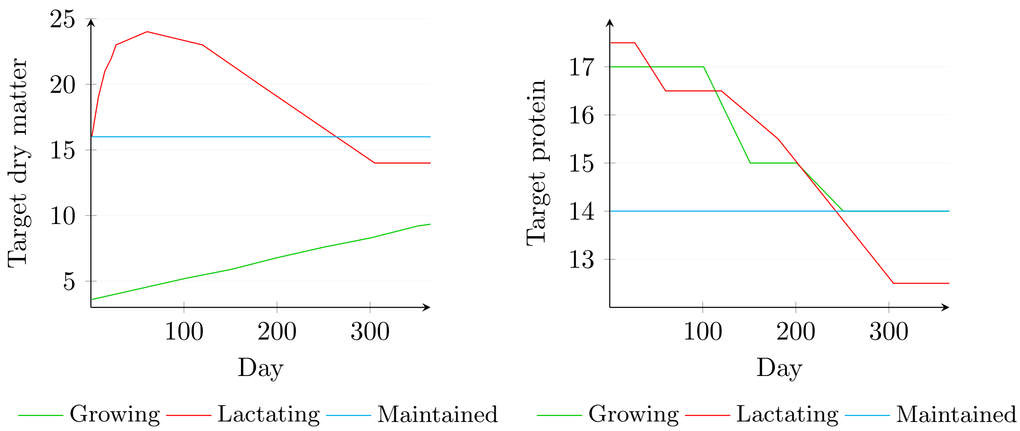
Group	**Description**
Growing	10 growing animals such as calves and heifers. The feeding requirements vary over the time horizon based on the animals' weight and growth rate.
Lactating	15 lactating animals of varying production. This group covers all stages of lactation: early lactation, mid-lactation, late lactation, and dry cows. The feeding requirements change depending on the stage of lactation.
Maintained	5 animals that are not lactating or in a growth stage, such as mature cows that have completed their lactation cycle and heifers that have not yet started their lactation cycle. The main goal when feeding these animals is to maintain their current body weight and health.

7.2 Feeding Requirements

For each animal group presented in Section 7.1, a data set is generated regarding the daily feed composition requirements. Each data set includes the daily minimum, maximum, and target values for dry matter, protein, and Neutral Detergent Fiber (NDF) over a time period of 365 days. Triangulation is used to combine data from multiple sources to improve the credibility and overall quality of the data. The main sources used for data collection are research articles, example data, and knowledge provided by TINE and TKS.

The daily requirements for each animal group are based on typical requirements for one animal in each animal group and multiplied by the number of animals. For the growing group, the single animal requirements are based on cattle with a starting weight of 100 kg and a daily growth rate of 1 kg per day. The lactating group's requirements are generated based on a typical lactation cycle, with specific requirements for each stage of lactation. The nutritional requirements for the maintenance group correspond to an average adult animal in a non-growth and non-lactating stage.

Figure 7.1 illustrates the target values for dry matter and protein for each of the three animal groups over a time period of 365 days. The NDF target is not illustrated, as this is constant at 30% of the target dry matter intake for all animal groups. The requirements for dry matter are illustrated in Figure 7.1a. For the growing animal group, the requirements for dry matter increase over time as the animal grows up and gains weight. For the lactating group, the dry matter requirements peak during the first months of lactation, following the development as described in Section 2.1.2. The dry matter requirements for the maintained group remain stable throughout the time period. Figure 7.1b illustrates the protein requirements for the animal groups. As the figure shows, the protein requirements for the growing group are higher at the beginning of the time horizon. As described in Section 2.1.2, this is because the animal requires a higher amount of protein to support high muscle and skeletal development during the early stages of growth. Likewise, for the lactating group, the protein requirements are highest during the early stage of lactation when milk production is at its peak. The protein requirements for the maintaining group remain consistent throughout the entire time period as illustrated in Figure 7.1b. The upper and lower bounds of all feeding requirements for the three animal groups are illustrated in Appendix B.



(a) Dry matter target content

(b) Protein target content

Figure 7.1: Target content of dry matter and protein for the different animal groups over a time horizon of 365 days.

The maximum amount of feed concentrate, F_a^{MAX}, is set based on the animals' stage and production level. For the lactating group, the maximum amount of feed concentrate in each feed composition is set to 70% of the target dry matter intake, as they require a higher level of nutrition to support milk production. Growing animals require less feed concentrate than lactating ones, and the limit is set to 40% of the target dry matter intake. For the maintained group, the lowest level of feed concentrate is required, and the limit is set to 20% of the target dry matter intake.

The penalty function parameter values, P^{ON}, P^{IN} and P^{IM}, are described in Chapter 5, and are set to penalize deviations from the target content. If the nutritional or dry matter content is between the target and the upper or lower limit, the penalty increases linearly until it reaches P^{IM} or P^{IN} at the limits, where the penalty is set to 1. The slope of the penalty function when the amount of nutritional content exceeds the limits is determined by P^{ON}. This value is set to 400 as the upper and lower bound are considered to be quite strict limits.

7.3 Ingredients

The ingredients used in our case study are limited to silage bales and feed concentrates. Data regarding the content of the different ingredients is mainly provided by TINE, but also supplemented by information from TKS and online research. Specifically, the Nordic feed evaluation platform NorFor (Nordic Feed Evaluation System, 2023) has been a valuable tool for getting additional information on the nutritional content of ingredients and ensuring the accuracy of the data generated. The costs of silage bales and the different types of feed concentrate are set to reflect the cost ratio found in the data provided by TINE.

7.3.1 Silage Bales

Our case study considers six silage bale batches, each becoming available at different times during the planning horizon. The silage bale batches are harvested from different places on the field at different times, affecting their nutritional content. The total amount of silage bales available in a batch is dependent on the number of days in the planning horizon of the test instance. This is done to ensure that all instances have comparable resource availability regardless of the length of the planning horizon. For batch 1, which is in stock from last year, the number of bales is set to $1 * |D|$. For all batches i that become available during the planning horizon (batch 2 - batch 6) the number of bales in each batch is set to $2 * |D|$.

The expected nutritional content of the silage bale batches on the day of harvest is based on actual silage quality analyses provided by TINE, found in Appendix C. To account for the natural deterioration of silage bales presented in Chapter 2, we assume that the expected content of dry matter, protein, and NDF decreases by 1% per day during the planning horizon. The cost of using a silage bale is set to 120 NOK on the day it becomes available. The cost is linearly reduced as the quality decreases.

As described in Section 2.1.3, the actual quality of each single silage bale is subject to uncertainty due to various factors such as cultivation strategy, soil quality, weather conditions, harvesting policy, handling, and storage conditions. We model this uncertainty using a triangular distribution, a modeling strategy that is supported by several studies (Hardaker et al., 2015; Shalloo et al., 2004). The triangular distribution is illustrated in Figure 7.2. The mode of the triangular distribution is

Table 7.2: Information on the silage bale batches used in this case study.

Batch $[i]$	**Day available** $[D_i^A]$
1	1 (In stock from last year)
2	1
3	1
4	5
5	10
6	20

set to 1. This represents the expected value of the silage bale batch on a given day. Given the potential risks associated with the fermentation process, we assume that the likelihood of a silage bale having lower quality than indicated by the sample analysis is higher than that of it having higher quality. Therefore, the minimum and maximum values are set to 0.4 and 1.2, respectively.

As described in Chapter 5, the scenarios are generated by drawing random values from the triangular distribution. Additionally, we account for a 5% probability that the fermentation process of the silage bale results in a complete loss, and that the silage bale is useless. When this is the case, the content of the silage bale in terms of both dry matter and nutritional content is set to 0.

In situations where the content of a silage bale is insufficient or the bale is unusable, an alternative solution is to utilize reserve silage as outlined in Chapter 4. To discourage regular reliance on the reserve silage, the cost of using it has been set at 10 NOK per kilogram of dry matter, which approximates 2000 NOK for an entire bale. This cost has been set high to ensure that it is not preferable to depend on the reserve silage. However, it remains low enough to make it more favorable to use the reserve silage in the rare cases when a silage bale is useless, rather than consistently transporting an additional silage bale.

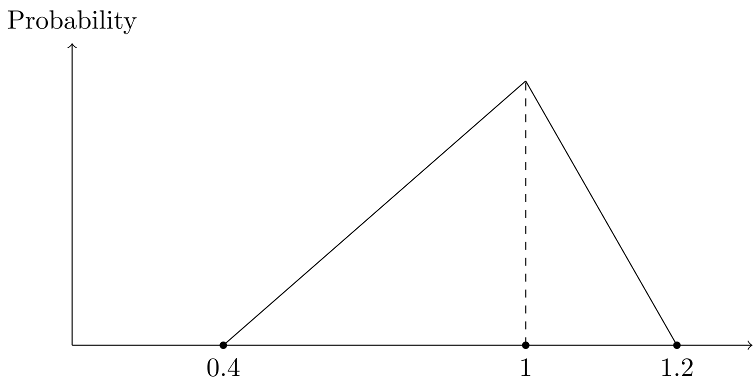


Figure 7.2: Triangular distribution used to model the uncertainty in silage bale content.

7.3.2 Feed Concentrates

TINE has shared the nutritional content of different types of feed concentrates. For this case study, four types with varying nutritional content and cost are included. An overview of these are presented in Table 7.3.

Table 7.3: Information on the feed concentrate types used in this case study.

Feed Concentrate	Description	Cost (kr/kg)
Formel Protein 32 FKA	High protein content, medium fiber quality	12
Formel Elite Normal FKA	Medium protein content, high fiber quality.	10
Formel Solid Normal FKA	Medium protein content, medium fiber quality.	9
Natura Protein FKA	Very high protein content, medium fiber quality	13

7.4 Limitations of Data Generated

It is important to note that the data generated has certain limitations, which may affect the accuracy of the results. These limitations are due to the lack of real-life data and assumptions made during the data generation process. Firstly, the generated feeding requirements only cover dry matter, protein, and NDF, which is not an accurate reflection of the complete nutritional requirements of dairy cattle. In addition, the data assumes that all animals in a particular group have the same feeding requirements, despite the fact that animals in a group may be at different stages of growth or lactation at different times. Lastly, the modeling of the uncertainty and natural deterioration of silage bales is a simplification of reality and does not accurately reflect reality. These limitations may affect the results of the model, and the choices it takes. However, we believe that the generated data is sufficient for testing the model and gaining valuable insights into the potential of the model as a useful tool for optimizing dairy cattle feeding.

Chapter 8

Computational Study

This chapter presents and discusses the computational results of this thesis. Throughout the chapter we aim to address the three goals outlined in Chapter 1; developing an efficient solution method to handle the computational complexity of the SCFPP, demonstrating the value of planning while considering uncertainty, and investigating the trade-off between cost and quality when developing feeding plans for dairy cattle.

The sections are organized as follows. To begin, we establish a base case in Section 8.1, determining the divisors and weights needed for meaningful comparisons between objectives in the model. Thereafter, Section 8.2 addresses the first goal of this thesis by analyzing and evaluating the technical performance of the proposed solution methods. In Section 8.3, a sample stability analysis is conducted to determine the appropriate number of scenarios for the remaining computational study. Sections 8.4, 8.5, and 8.6 adress the second goal of the thesis, and investigate the value of long-term planning while considering uncertainty. Section 8.4 studies the value of uncertainty, Section 8.5 explores the benefits of extended planning horizons, and Section 8.6 analyzes the impact of reducing the number of days with uncertainty to enable longer planning horizons. Finally, in Section 8.7, we address the last goal of the thesis by investigating the trade-off between cost and quality.

8.1 Base Case Divisors and Weights

The objective function of the model presented in Chapter 5 consists of two components: cost and quality, which are combined using a weighted sum objective. To compare these objectives effectively during the computational study and to make it easier to interpret α, it is necessary to establish an objective function where none of the terms is dominant when the two objectives are weighted the same, i.e. $\alpha = 0.5$. Since the magnitudes of the individual objective terms may differ significantly, each term is assigned a divisor that ensures they possess equal significance in the overall objective function.

To determine the balancing divisors, we solve the problem with different values of α, specifically 0.99 and 0.01. These values correspond to approximately solving a single-objective problem for quality and cost, respectively. The objective value for the prioritized objective is set as the divisor for that particular objective. As a result, when implementing these divisors, the best possible solution for each objective is represented by a value of 1. All subsequent problem instances discussed in this chapter utilize the divisors presented in Table 8.1. It is worth noting that the divisors are generated by solving a problem with 10 days and 20 scenarios. Even though we have not generated one specific divisor for each of the test instances, the presented divisors are evaluated to be adequate for the analysis done in this thesis. Furthermore, all models are run with $\alpha = 0.5$ unless stated otherwise.

Table 8.1: The objectives with their respective divisors.

Objective	**Divisor**
Quality	51.86
Cost	1597.60

8.2 Model Performance Analysis

This section addresses one of the three goals presented in Chapter 1, namely, finding a solution method that effectively solves the SCFPP. This is done by evaluating the performance result of the model when solved with the solution methods presented in Chapter 6, and comparing this to the performance of the standard commercial solver, Gurobi. First, the performance of the standard Gurobi solver is presented in Section 8.2.1. Thereafter, Section 8.2.2 evaluates the L-shaped method and its acceleration methods one by one, aiming to find a suitable combination of acceleration methods. Lastly, Section 8.2.3 compares the resulting combined acceleration method of the L-shaped method to the performance of the Gurobi solver. The primary focus when evaluating and comparing the methods is studying how the optimality gap and bounds evolve over time for the different modeling techniques.

All instances are run with a time limit equal to three hours (10 800 seconds), a maximum number of iterations of 1 000, and a 1% optimality gap as termination criteria. The model is implemented using the commercial software Gurobi Optimizer version 9.5.2 with the Gurobi Python Interface. All preprocessing and test instance generation is done using Python 3.8.8. The problem instances are run on a computational cluster provided by the Department of Industrial Economics and Technology Management at NTNU. The specifications of the computational nodes that are used are presented in Table 8.2.

Table 8.2: Hardware specifications.

Computer	Dell PowerEdge R640
Processor	2x2.4GHz Intel Xeon Gold 5115 CPU - 10 core
RAM	96Gb
Disk	55Gb SATA SSD

The model is solved for different numbers of days in the planning horizon, number of scenarios, and solution methods. Based on results from preliminary testing, the number of days are set to 5, 10, 15, and 20 days and the number of scenarios is set to 2, 10, 20, 60, and 100 scenarios. We apply the standard Gurobi solver as well as the solution methods presented in Chapter 6 to our test instances. We use notation to differentiate between the instances, where we name the test instance from the number of days (D) and the number of scenarios (S). For example, a test instance denoted as 10D20S represents an instance with 10 days and 20 scenarios.

8.2.1 Model Performance using Gurobi Solver

This section provides an analysis of the results obtained by applying the standard Gurobi Solver to solve the SCFPP. An overview of the results is presented in Table 8.3, while the complete result table can be found in Appendix D.

The findings reveal that for the smallest instances, the commercial solver successfully solves the problem to optimality well within the specified time limit of 10 800 seconds. However, as the period length increases, the optimality gap and runtime do the same. In the case of the largest instances, the solver obtains a 100% optimality gap when reaching the time limit. This highlights the need to explore alternative solution approaches to improve the overall performance.

Table 8.3: Results from solving the problem instances using the standard Gurobi solver.

	Number of days					
	5		**10 days**		**20 days**	
Scenarios	time	gap	time	gap	time	gap
2	2.72s	0.52%	1576.15s	1.00%	-*	8.43%
10	54.83s	0.00%	-*	3.95%	-*	32.03%
20	453.75s	0.71%	-*	16.18%	-*	32.14%
100	722.34s	0.00%	-*	99.99%	-*	100.00%

* timed out (runtime > 10 800s)

8.2.2 Performance of the L-Shaped Decomposition Method

This subsection compares the different versions of the L-shaped algorithm, aiming at finding a good version of the method to use further in the computational study. The original L-shaped method, i.e. the single-cut version, serves as our initial Base Case (BC). One by one, the acceleration techniques presented in Section 6.3 are added to the general L-shaped algorithm. Each are evaluated against the current BC and added to the BC if the results are better. This is a greedy approach where we are not guaranteed to find the best combination of acceleration techniques. However, we consider it to be an approach that will result in a sufficiently good acceleration approach for this thesis. The enhancement techniques are added and evaluated in the order presented in Table 8.4.

Comparing the Single- and Multi-Cut Version of the L-Shaped Approach

The first acceleration technique to be compared to the Base Case is the multi-cut version of the L-shaped method. The performance of the L-shaped method with single- and multi-cut is presented in Table 8.5 and 8.6, respectively. We observe that the multi-cut version outperforms the single-cut version for all instances where the number of days is five, and for most instances when the number

Table 8.4: The order the acceleration methods are added to the Base Case and evaluated.

Number	Evaluated Method(s)
1	Multi-cut version of L-shaped
2	Warm start techniques and the Two-Phase (TP) approach
3	The Magnanti-Wong (MW) method
4	LB-stop
5	The ϵ- and the SolNum approach

of days is 15. However, when the number of days reaches 20, both the single- and the multi-cut version struggles.

Table 8.5: Results from solving the problem instances using the single-cut version of the L-shaped method.

	Number of days					
	5 days		**15 days**		**20 days**	
Scenarios	time	gap	time	gap	time	gap
2	195.93s	0.99%	-*	91.87%	-*	99.43%
10	-*	15.73%	-*	98.02%	-*	96.71%
20	-*	75.49%	-*	96.99%	-*	98.85%
100	-*	69.17%	-*	96.28%	-*	95.67%

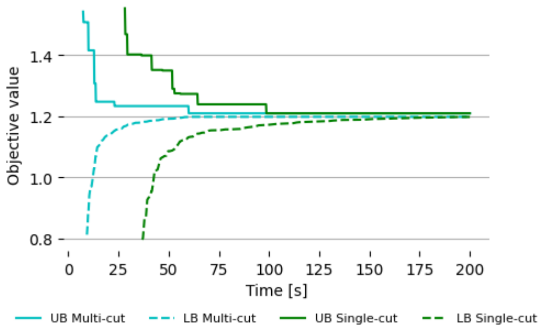
* timed out (runtime > 10 800s)

Table 8.6: Results from solving the problem instances using the multi-cut version of the L-shaped method.

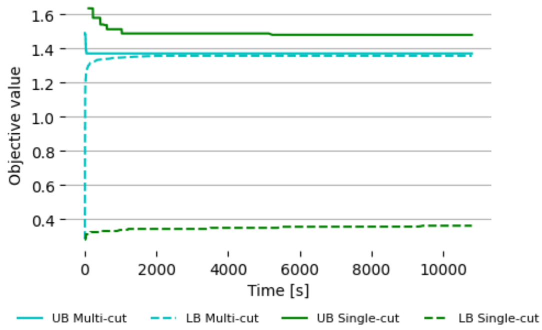
	Number of days					
	5		**15**		**20**	
Scenarios	time	gap	time	gap	time	gap
2	62.04s	0.96%	-*	92.52%	-*	95.91%
10	162.79s	0.94%	-*	29.67%	-*	96.91%
20	2225.56s	1.00%	-*	24.89%	-*	98.93%
100	-*	2.10%	-*	93.81%	-*	95.85%

* timed out (runtime > 10 800s)

Figure 8.1 illustrates the upper and lower bound development for the single- and multi-cut version of the L-shaped method on the instances 5D2S and 5D20S. In Figure 8.1a we observe that both instances find the optimal upper bound long before the time limit is reached. However, it is evident that the cuts generated from the multi-cut version are significantly more efficient than those in the single-cut version. This can be seen from the fact that the lower bound increases faster for the multi-cut version. This distinction becomes particularly apparent when looking at Figure 8.1b. In this graph, the lower bound of the single-cut version shows minimal growth, struggling to increase at all, whereas the lower bound of the multi-cut version approaches the upper bound, indicating a much more efficient optimization.



(a) Results for instance 5D2S.



(b) Results for instance 5D20S.

Figure 8.1: Lower- and upper bound development for the BC (single-cut) and multi-cut version of the L-shaped method.

As the multi-cut version dominates the single-cut version on smaller instances, and has a comparable performance on larger instances, we consider it beneficial to incorporate the multi-cut version as part of the BC. Consequently, the multi-cut version is employed for all the following evaluations.

Effect of Adding Warm Start Methods and the Two-Phase Approach

In the following evaluation, we investigate the effect of adding each of the two warm start techniques introduced in Chapter 6, namely the Robust Warm start (RW) algorithm and the Initializing Warm start (IW). The Two-Phase (TP) approach is also evaluated at this step, as this method has similarities to the warm start techniques. Each of these three methods is individually added to the Base Case (BC) and compared against one another. By evaluating their performance and effectiveness, we aim to determine whether any of these three approaches should be incorporated into the BC as an acceleration technique. Since all three approaches initialize the solution process in different ways, we only consider adding one of them to the BC.

As described in Chapter 6 the Two-Phase approach allows us to relax the integer property in the master problem until a chosen point in time. In this particular implementation, we transition to an integer master problem when the relative gap between the upper and lower bounds is below a predefined tolerance value. For this study, we have set this value to 0.2, considering it to be a suitably small value that can produce satisfactory linear programming solutions without significantly impacting the computational time. With this choice, we aim at achieving a balance between obtaining good integer solutions and maintaining computational efficiency.

Table 8.7, 8.8 and 8.9 approach presents the results for RW, IW and TP, respectively. All solution methods are able to solve all but one of the instances with five days planning horizon to 1%-optimality. Here, the RW method demonstrates the shortest computational solving times. However, as the instance size grows, the optimality gap for all solution methods increases. The TP approach performs best when this is the case, with the lowest optimality gap for several instances.

For most of the 15 and 20 day horizon instances, the TP approach is able to get lower gaps compared to both the RW and IW method. This indicates that the TP approach is more effective at improving the lower bound for instances with a longer time horizon.

Compared to the BC, all the added acceleration methods performs slightly worse on computational time for the smaller instances. However, as the number of days and scenarios increase, the TP approach performs slightly better when considering optimality gaps, as can be seen for all instances with 20 days.

Table 8.7: Results from solving the problem instances using the Robust Warm start (RW).

	Number of days					
	5		**15**		**20**	
Scenarios	time	gap	time	gap	time	gap
2	72.77s	0.99%	-*	91.87%	-*	92.72%
10	278.34s	0.94%	-*	50.92%	-*	95.86%
20	2783.91s	0.96%	-*	42.21%	-*	99.01%
100	-*	3.31%	-*	96.27%	-*	96.67%

* timed out (runtime > 10 800s)

Table 8.8: Results from solving the problem instances using the Initializing Warm start (IW) method.

	Number of days					
	5		**15**		**20**	
Scenarios	time	gap	time	gap	time	gap
2	131.02s	0.00%	-*	91.87%	-*	92.72%
10	572.28s	0.71%	-*	50.39%	-*	96.95%
20	9563.87s	0.58%	-*	46.16%	-*	98.85%
100	-*	75.08%	-*	96.27%	-*	96.35%

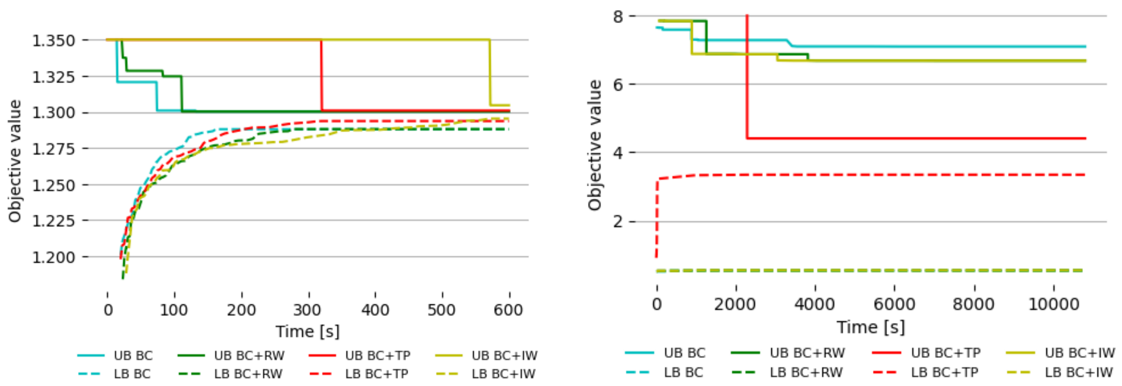
* timed out (runtime > 10 800s)

Figure 8.2 presents the progression of upper and lower bounds for the three warm start techniques (RW, IW, and TP) along with the BC on two selected test instances: 5D10S and 15D2S. When analyzing Figure 8.2a, it is evident that both the BC and RW method are able to find good feasible solutions relatively fast for test instance 5D10S, while the TP approach and IW struggle more. However, it is important to note that the lower bound on the TP approach increases at a faster rate than all the other methods. Furthermore, examining the instance presented in Figure 8.2b, it becomes apparent that the TP approach outperforms the other methods by finding both a better upper and lower bound. It is worth noting that the TP approach does not find a valid upper bound until it exceeds 2 000 seconds, unlike the other methods. However, once the TP approach finds a feasible solution, it significantly outperforms the quality of the solutions obtained by the other methods within the time limit of 10 800 seconds.

Table 8.9: Results from solving the problem instances using the Two-Phase (TP) approach.

	Number of days					
	5		**15**		**20**	
Scenarios	time	gap	time	gap	time	gap
2	103.14s	0.00%	-*	23.57%	-*	29.33%
10	350.50s	0.56%	-*	47.09%	-*	94.91%
20	7193.22s	0.76%	-*	76.50%	-*	87.45%
100	-*	5.15%	-*	31.71%	-*	67.60%

* timed out (runtime > 10 800s)



(a) Results for instance 5D10S.

(b) Results for instance 15D2S.

Figure 8.2: Lower- and upper bound development for the Base Case (BC), Robust Warm start (RW) method, Initializing Warm start (IW) method, and Two-Phase (TP) approach.

Based on the analysis presented, both the TP approach and the RW method demonstrate promising results. As several remaining acceleration techniques rely on iterations, we look at how many iterations the two methods are able to perform within the time limit. This is done to distinguish between the methods and decide which method to add to the BC. An overview of the number of iterations for the largest instances is presented in Table 8.10. The results clearly show that the TP approach is able to perform a higher number of iterations compared to the RW method.

When considering which method to proceed with, the TP approach is the preferred choice. As the approach performs good in terms of gap and solution time, as well as being able to perform a high number of iterations within the time limit, we consider it to have the greatest potential for further improvements. By leveraging the TP approach's capability for a larger number of iterations, we can take full advantage of the additional acceleration techniques. This strategic choice is expected to yield improved outcomes, accelerating the overall effectiveness of the approach.

Effect of Adding the Magnanti-Wong Method

The next acceleration technique is the Magnanti-Wong (MW) method, which creates Pareto-optimal optimality cuts. As stated in Chapter 6, the core point used in the Magnanti-Wong

Table 8.10: Number of iterations performed when using the Two-Phase approach and the Robust Warm start method.

	Number of days			
	15		**20**	
Scenarios	TP	RW	TP	RW
20	18	17	21	2
60	16	2	20	2
100	15	2	17	2

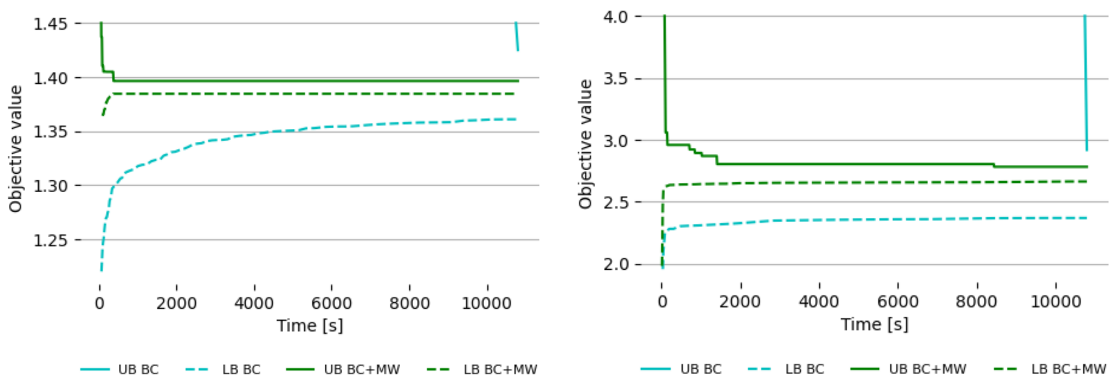
subproblem in this thesis is found by taking the average of the optimal solution found and another feasible solution found in the current iteration of the master problem. If only one feasible solution is found when solving the master problem, the MW method is not used in that iteration. An overview of the performance when adding this method to the BC is presented in Table 8.11. Here, we observe that the method finds a solution with a gap smaller than 1.00% for all the smallest instances. Furthermore, it manages to find a sufficiently small gap as the number of days increases. However, for the largest instances with the highest number of scenarios, the method struggles. The complete performance table can be found in Appendix D.

Table 8.11: Results from solving the problem instances with the Magnanti-Wong method added to the Base Case.

	Number of days					
	5		**10**		**20**	
Scenarios	time	gap	time	gap	time	gap
2	6.48s	0.26%	-*	16.52%	-*	11.72%
10	20.94s	0.81%	-*	10.25%	-*	61.32%
20	42.56s	0.30%	-*	15.01%	-*	81.93%
100	443.87s	0.96%	-*	78.52%	-*	61.54%

* timed out (runtime > 10 800s)

The upper and lower bound development for the BC and the BC with the MW method for instance 5D60S and 10D20S is illustrated in Figure 8.3. Figure 8.3 clearly illustrates the improvement of adding the Magnanti-Wong method to the BC. The implementation of Pareto-optimal cuts through the Magnanti-Wong subproblem results in a much quicker convergence. This is as expected, as the MW method aims at quicker cutting away the infeasible solutions and consequently needs less iterations to find the optimal solution. As the presented results show, the Magnanti-Wong method improves the solution method on almost all instances. The conclusion is therefore to include the MW method in the Base Case in the further computations.



(a) Results for instance 5D60S.

(b) Results for instance 10D20S.

Figure 8.3: Lower- and upper bound development for the Base Case (BC) and Magnanti-Wong (MW) method.

Effect of Terminating the Solution Process in the Master Problem when Current Lower Bound is Found

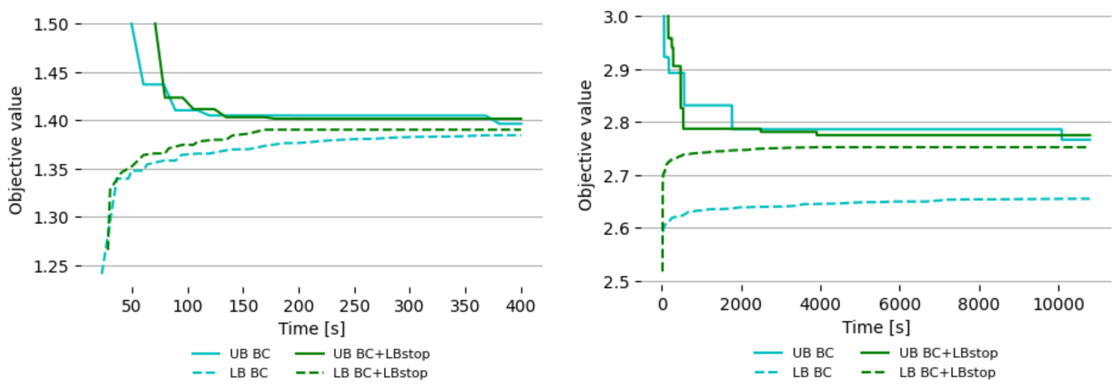
To try to reduce the time spent on solving the master problem in each iteration, we add the LB-stop method to the problem, where the current iteration of the master problem stops if an objective value equal to the current lower bound is found. A selection of the results from this approach is presented in Table 8.12.

Table 8.12: Results from solving the problem instances with the () method added to the Base Case (BC).

	Number of days					
	5		10		20	
Scenarios	time	gap	time	gap	time	gap
2	7.20	0.26%	6063.50	0.24%	-*	11.36%
10	25.44s	0.88%	3908.232s	0.82%	-*	14.67%
20	72.18s	0.99%	-*	3.91%	-*	69.43%
100	451.80s	0.96%	-*	8.20%	-*	38.38%

* timed out (runtime > 10 800s)

In comparison to the current Base Case, the runtime for the smallest instances is longer. However, the gap is smaller for the instances where the time limit is reached. By looking at Figure 8.4 we see that by adding the LB-stop method, the solver is able to increase the lower bound faster. This aligns with the expected behavior, as the LB-stop method aims at spending less time solving the master problem and consequently is able to iterate faster.



(a) Results for instance 5D60S.

(b) Results for instance 10D10S.

Figure 8.4: Lower- and upper bound development for the Base Case (BC) and the LB-stop method.

As the results show that adding the LB-stop method to the BC improves the result for the majority of the instances, the LB-stop method is added to the BC.

Effect of Adding the ϵ- and SolNum Approach

The next acceleration approaches that are evaluated are the ϵ approach and the SolNum approach. As stated in Chapter 6, these two approaches aim at giving a good description of the master problem in a faster manner. The performance of both the ϵ- and SolNum approach depend on their initialized settings. The ϵ approach depend on the value of ϵ and the SolNum approach depend on how many solutions it should find in each iteration. We initially set ϵ to 0.10 based on preliminary testing for different ϵ values. The value is set to decrease with 0.02 on each iteration. This means that ϵ is set to 0 after five iterations. For the SolNum approach, the number of solutions to find is set to 10 for the first five iterations and the maximum for all other iterations. A selection of the results from solving the test instances with the ϵ- and SolNum approach are presented in Table 8.13 and Table 8.14, respectively. The complete result table can be found in Appendix D.

Table 8.13: Results from solving the problem instances with the ϵ-approach added to the Base Case.

	Number of days					
	5		**10**		**20**	
Scenarios	time	gap	time	gap	time	gap
2	6.64s	0.94%	-*	1.22%	-*	17.89%
10	23.70s	0.83%	-*	6.04%	-*	40.65%
20	39.04s	0.96%	-*	5.12%	-*	69.33%
100	-*	0.25%	-*	25.48%	-*	65.68%

* timed out (runtime > 10 800s)

When comparing the results in Table 8.13 and Table 8.14 to the BC in Table 8.12, we find that both the BC and the SolNum approach demonstrates more consistent performance in reducing

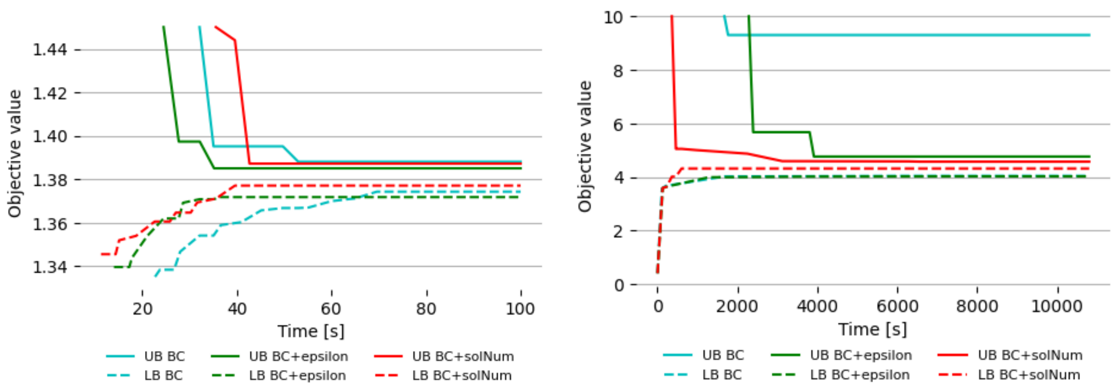
Table 8.14: Results from solving the problem instances with the SolNum approach added to the Base Case.

	Number of days					
	5		10		20	
Scenarios	time	gap	time	gap	time	gap
2	7.05s	0.26%	7301.762s	0.24%	-*	15.31%
10	25.35s	0.88%	-*	4.02%	-*	10.23%
20	39.58s	0.73%	-*	4.29%	-*	9.85%
100	133.09s	0.66%	-*	3.68%	-*	9.47%

* timed out (runtime > 10 800s)

optimality gaps compared to the ϵ approach. Furthermore, both the BC and SolNum approach are able to solve more instances to optimality than the ϵ approach. This result can be due to the initial value of ϵ, as the larger instances might benefit from having a larger initial ϵ value than what is appropriate to use in the smallest instances. When comparing the BC with the SolNum approach, the SolNum approach tends to either solve the instances faster or obtain a lower gap when the solution process reaches the time limit. The SolNum approach exhibits slightly worse performance when the number of scenarios is low. However, as the number of scenarios and the number of days increase, the SolNum approach significantly outperforms the BC algorithm.

Figure 8.5 illustrates the development of the upper- and lower bound when solving instance 5D20S and 15D100S using the BC-, ϵ- and SolNum approach. For instance 5D20S, Figure 8.5a shows that all three solution methods finds a similar upper bound with less than a 1% gap to the lower bound within a short amount of time. However, it is clear that the lower bound increase significantly faster for the ϵ- and SolNum approach. This is expected, as the two approaches aims at finding a good description of the master problem in a short amount of time by iterating rapidly in the beginning of the solution process. The same tendency can be seen in Figure 8.5b, where the BC struggles to find a good upper bound while both the ϵ- and the SolNum approach performs better. However, Figure 8.5b shows that the SolNum approach is more efficient in the convergence, which indicates that the method manages to gather more information than the ϵ approach.



(a) Results for instance 5D20S.

(b) Results for instance 15D100S.

Figure 8.5: Lower- and upper bound development for the Base Case (BC), ϵ approach and SolNum method.

As the presented results show, adding the SolNum approach results in better performance of the L-shaped approach compared to the current Base Case and the ϵ-approach. We therefore add the SolNum approach to the Base Case in the subsequent computational results.

The Resulting Best Combination of Acceleration Methods

An overview of the acceleration methods and whether or not they are added to the Base Case is presented in Table 8.15. The resulting accelerated L-shaped approach consists of adding multiple cuts in each iteration, solving the problem as a two-phase problem, adding Pareto-optimal cuts, stopping the solution process in the master problem when the current lower bound is found, and limiting the number of found solutions in the master problem for the first iterations. In the following section, this combination of acceleration techniques is compared to the performance of the standard Gurobi solver.

Table 8.15: The evaluated acceleration methods and whether or not they are added to the Base Case.

Acceleration Method	Added to Base Case
Multi-cut version of L-shaped	✓
Robust Warm start technique	✗
Initializing Warm start technique	✗
The Two-Phase approach	✓
The Magnanti-Wong method	✓
LB-stop	✓
The ϵ approach	✗
The SolNum approach	✓

8.2.3 Comparing the Accelerated L-Shaped Method and the Gurobi Solver

In this subsection, we compare the results from solving the instances using the standard Gurobi solver and the resulting accelerated L-shaped method. The results from solving the instances using Gurobi solver are presented in Table 8.3, and results from solving with the accelerated L-shaped method are presented in Table 8.14. Table 8.16 presents the best performance of the two approaches for some of the test instances. The cell color presents the solution method used to get the results presented in the cell. When the color of the cell is green, the solution approach used to obtain the results is the standard Gurobi solver. When the cell color of the cell is blue, the used solution approach is the accelerated L-shaped method.

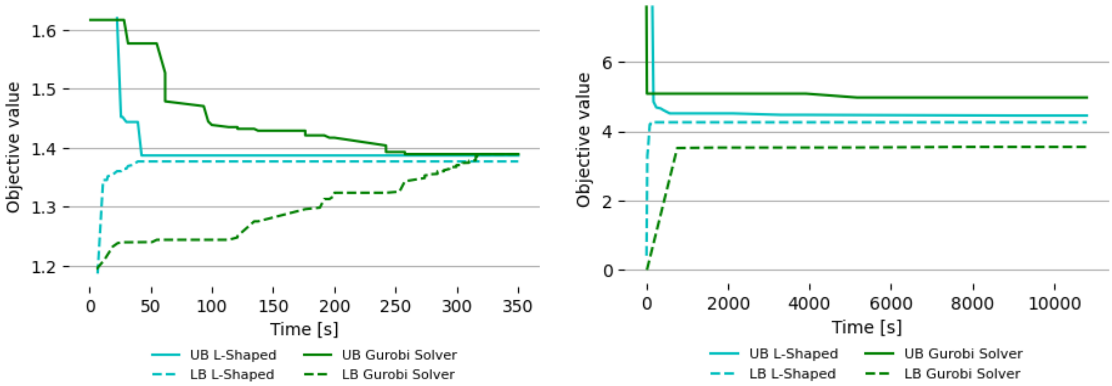
Table 8.16: Best performance results when solving the problem instances. Green = best performance when using the standard Gurobi solver. Blue = best performance when using the accelerated L-shaped method.

	Number of days					
	5		**10**		**20**	
Scenarios	time	gap	time	gap	time	gap
2	2.72s	0.52%	1576.15	1.00%	-*	15.31%
10	25.35s	0.88%	-*	3.95%	-*	10.23%
20	39.58s	0.73%	-*	4.29%	-*	9.85%
100	133.09s	0.66%	-*	3.68%	-*	9.47%

* timed out (runtime > 10 800s)

From Table 8.3 and 8.14 it is clear that the Gurobi solver obtains a greater gap than the accelerated L-shaped method for all instances that times out except from one. Furthermore, the accelerated L-shaped method finds a solution with a smaller gap than 1% faster than the Gurobi solver for three out of five of the instances presented in Table 8.16. This indicates that the accelerated L-shaped method is a well-suited decomposition technique for the problem.

Figure 8.6 illustrates the lower- and upper bound development for solving instance 5D20S and 15D20S by using Gurobi solver and the accelerated L-shaped method. As expected, the accelerated L-shaped method converges faster to the optimal solution than the Gurobi solver. The upper- and lower bound for the acceleration technique moves rapidly towards the optimal solution both in Figure 8.6a, where both solution methods finds the optimal solution within the time limit, and in Figure 8.6b, where neither of the models finds an optimal solution within the time limit.



(a) Results for instance 5D20S.

(b) Results for instance 15D20S.

Figure 8.6: Lower- and upper bound development for the accelerated L-shaped method and Gurobi solver.

As the presented results show, the accelerated L-shaped method outperforms the standard Gurobi solver in the majority of the test instances, either by having lower runtime or by having a lower gap at the time limit. Therefore, the computations in the following sections are solved using the accelerated L-shaped approach.

8.3 Sample Stability Analysis

In this section, we perform stability tests to evaluate the model and the scenario generation procedure. As described in Chapter 5, scenarios are generated based on sampling, and it is important to investigate whether this scenario generation approach results in reliable and accurate results. Our goal with the sample stability analysis is to determine a reasonable number of scenarios to use when running the model and ensure that the generated scenarios provide a valid representation of the underlying probability distribution.

There are two main types of stability tests: in-sample and out-of-sample. The in-sample stability test is a test on the model's internal consistency and investigates whether the scenario generation procedure generates scenarios that result in similar objective values. The out-of-sample stability test is a test on the model itself and studies the model's performance when the solution from the model is evaluated based on real-life data. The sample stability analysis presented in this section is performed on test instances of ten days with varying numbers of scenarios. All theory presented in this section is from King and Wallace, 2012.

8.3.1 In-Sample Stability

When testing in-sample stability, the objective values of instances that are solved with different scenarios are compared. If all instances get approximately the same objective value, regardless of the scenario tree used in the model, we have in-sample stability. This means that with perfect in-sample stability, the model will find the same objective function value, independent of the scenarios used in the instance.

To find the scenario size that gives in-sample stability for the SCFPP, the model is solved for an increasing number of scenarios. For each scenario size, we solve five instances where different scenarios are generated for each instance. We then compare the variations in the objective values for all instances of the same scenario size.

Figure 8.7 illustrates the standard deviation in the objective values for the different scenario sizes. The standard deviation is relatively low for all instances, ranging from 0.01 to 0.06. However, as seen in the figure, the standard deviation decreases as the number of scenarios increases. The improvement is especially evident as the number of scenarios reaches 30. After 30 scenarios, the figure shows that the standard deviations stabilize to a large extent, demonstrating a clear trend toward enhanced consistency in the results.

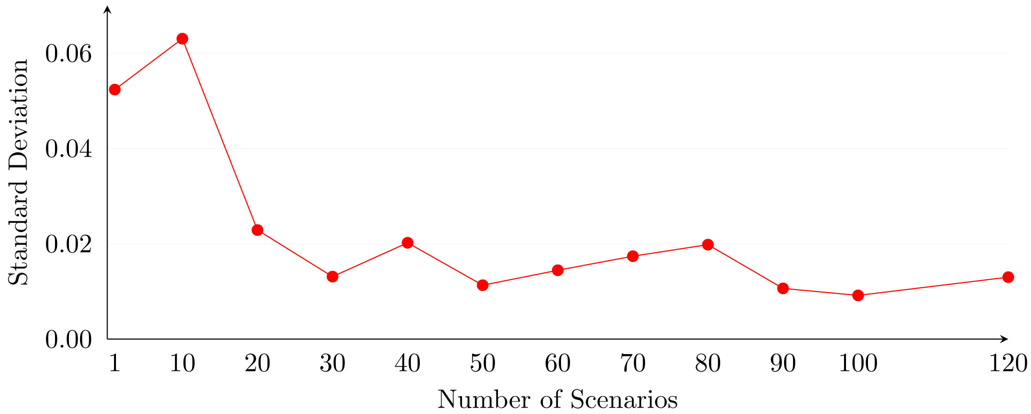


Figure 8.7: Results from in-stample stability test.

8.3.2 Out-of-Sample Stability

In an out-of-sample stability test, the goal is to test if you get approximately the same true objective value when a problem is solved with different scenario trees. The true objective value is the objective value of the true problem, where all possible scenarios are taken into account. The out-of-sample stability test is performed by solving the model for an increasing number of scenarios, where five instances are solved for each. For all instances with a given number of scenarios, the true objective value is approximated and compared. As it is nearly impossible to solve the true problem, we instead approximate the true objective value by fixing the first-stage variables from the original solution, and then solving the problem taking 1 000 scenarios into account. This approach provides an estimation of the true objective value, which we can use for comparison.

Figure 8.8 shows the standard deviation of the true objective value for the instances of different scenario sizes. We see a similar effect as for the in-sample test, where the variation significantly decreases as the number of scenarios increases. Specifically, the standard deviation is reduced from 0.39 when the models are solved with one scenario to 0.12 when the number of scenarios reaches 30. After this point, the standard deviation varies between 0.07 and 0.18. Although it does not stabilize completely, it remains at a significantly lower variation compared to the instances with 1 to 20 scenarios. The standard deviation is overall higher for the out-sample stability test than for the in-sample test. This means that the model requires a larger number of scenarios to give reliable results for the problem in real life.

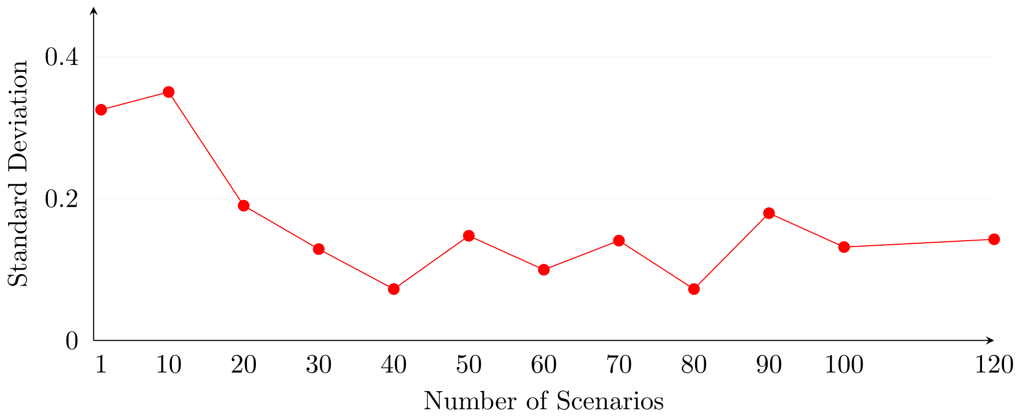


Figure 8.8: Results from out-of-stample stability test.

8.3.3 Conclusion from Sample Stability Tests

The aim of this section is to determine a reasonable number of scenarios to use when running future tests of the model, that ensures an acceptable stability of the results. The results from the stability tests indicate that the in-sample stability is overall higher than the out-of-sample stability. However, for both tests, the results demonstrate enhanced stability as the number of scenarios increases. As reviewed in Section 8.2, the computational performance of the model is significantly reduced as the number of scenarios increases. Consequently, there is a trade-off between ensuring high stability and acceptable computational performance. Taking the findings from the stability tests and the trade-off between stability and computational performance into consideration, we find that using 30 scenarios is a reasonable number of scenarios. This ensures that the model is able to solve the problems with a smaller gap, while still giving quite stable results.

8.4 The Value of Uncertainty

One goal of this thesis is to investigate the value of tactical planning with uncertainty. To contribute to this goal, this section studies the value of taking uncertainty into account when solving the SCFPP. We begin by quantifying the value by using the Value of the Stochastic Solution (VSS). Thereafter, we study the impact in a simulated reality. Lastly, we investigate the cost and quality of the animal feed from a deterministic versus a stochastic solution, as well as the different choices made by the model.

8.4.1 Value of the Stochastic Solution

The VSS is a concept that quantifies the value of uncertainty. It measures the benefit that can be obtained by using a stochastic solution rather than a deterministic solution when facing uncertain outcomes. The equation for calculating the VSS for a minimization problem is stated in Equation (8.1) (Birge & Louveaux, 2011).

$$VSS = EEV - RP \tag{8.1}$$

In Equation 8.1, the Expected result of using the Expected Value solution (EEV) is a measure of the quality of a solution when the expected value of the random variables is taken into account. The Recourse Problem (RP) represents the value of the stochastic problem. To calculate the EEV, we first solve the Expected Value (EV) problem. This is solved by only considering one scenario, where all silage bales have a quality equal to the expected value of their corresponding batch. Thereafter, the first-stage decisions of the EV problem are fixed, and the model is solved by using the same set of scenarios as for the RP.

Table 8.17: VSS for test instances with planning horizon of 5, 10 and 15 days.

Instance	**EEV**	**RP**	**VSS**	**VSS/EEV (%)**
5D30S	1.22	0.99	0.22	18%
10D30S	2.49	2.09	0.40	16%
15D30S	3.64	3.86	0.23	6%

Table 8.17 shows the results for the VSS calculations for the SCFPP for three different instances, 5D30S, 10D30S, and 15D30S. As the table shows, the objective value is improved for all instances when the model takes uncertainty into account. Note that the percentage of improvement tends to decrease as the planning horizon increases. This may be due to the fact that as the number of days increases, so does the optimality gap. Hence, the optimal solutions are not found for the larger instances. However, as the EEV is significantly positive for all instances, the results indicate that taking uncertainty into account when determining a feeding plan has a substantial value for the farmer.

8.4.2 Value of Uncertainty in a Simulated Reality

When determining the VSS, both models are solved for the same set of scenarios. This set of scenarios are the same scenarios as the RP consider when solving the problem. It is therefore natural that the RP in most cases achieves better results than the EEV, as supported by the results in Section 8.4.1. Therefore, it is interesting to also compare the stochastic and deterministic solution in a possible real-life situation. In this section, we explore the value of uncertainty in a simulated reality by comparing the stochastic and deterministic solutions when both solutions are evaluated based on a simulated scenario. We refer to this analysis as the VSS^{SIM}, where the comparison objectives with and without uncertainty are denoted as EEV^{SIM} and RP^{SIM}, respectively.

To calculate EEV^{SIM} and RP^{SIM}, we use the simulation framework introduced in Section 6.5. As for the traditional EEV, we first solve the EV problem, considering the scenario where all silage bales take their expected value. We then fix the first-stage variables and run the simulation model, denoting the solution as EEV^{SIM}. Thereafter, we solve the stochastic problem based on a set of scenarios and run the simulation model for the same simulated scenario as the EEV^{SIM}, denoting this solution as RP^{SIM}. Lastly, we compare the two solutions. Since the simulated scenarios are generated randomly, the outcomes may be affected by the specific scenarios generated. To account for this variability, we run the simulation five times and compare the average objective values. The objective values for all simulations are found in Appendix E.

Table 8.18: VSS^{SIM} for test instances with planning horizon of 5, 10 and 15 days. The values presented are average values based on five simulated scenarios.

Instance	$\mathbf{EEV^{SIM}}$	$\mathbf{RP^{SIM}}$	$\mathbf{VSS^{SIM}}$	$\mathbf{VSS^{SIM}/EEV^{SIM}}$(%)
5D30S	1.28	0.95	0.32	25%
10D30S	2.55	1.99	0.56	21%
15D30S	3.69	3.05	0.64	17%

Table 8.18 presents the results when evaluating the solutions in a simulation-based scenario. These findings align with those presented in Table 8.17, demonstrating a notable improvement in the objective function value when the first-stage decisions are made using a stochastic model (RP^{SIM}) compared to a deterministic model (EEV^{SIM}). Specifically, all three instances indicate 17-25% improvement when solved utilizing the stochastic model. Note that when comparing the results in Table 8.18 to the results in Table 8.17, the VSS^{SIM} is higher than the VSS for all instances. The reason for this may be that the simulated scenarios are randomly generated, introducing variability in the results for each simulated reality. These random variations can lead to more extreme individual cases, some resulting in a substantial effect on solving the stochastic program. These variations are seen from the results of each simulation, which are presented in Appendix E. Nevertheless, the results from this analysis support the findings from Table 8.17, and indicate a significant effect of considering uncertainty, also when considering a real-life scenario.

8.4.3 Impact on Objective and Decisions

Section 8.4.1 and Section 8.4.2 illustrate the value of considering uncertainty by comparing the objective value of stochastic and deterministic solution methods in a stochastic environment. However, it is also interesting to study the impact on the different parts of the objective and the decisions made by the model as this provides insights into the practical benefits of considering uncertainty in feed planning. In this section, we study the impact of considering uncertainty on the cost and quality and investigate the differences it leads to in decision-making. We use the test instances from Table 8.18, and compare the stochastic and deterministic solution when evaluated in a simulated reality.

Table 8.19 provides a breakdown of the cost and quality objectives for both the deterministic and stochastic solutions. The cost is divided into first-stage and second-stage costs, where the first-stage costs are costs associated with the transportation of silage bales, and the second-stage costs are related to the usage of feed concentrate and reserve silage bales. The quality is represented by the total penalty of deviating from the target for dry matter and nutritional content, denoted as Q1, as well as the penalty for not ensuring a stable diet for the animals, denoted as Q2.

The results from Table 8.19 show that for all instances, the first-stage costs are higher for the stochastic model compared to the deterministic model, indicating that the stochastic model transports more silage bales. Furthermore, the table shows that the stochastic solutions consistently have lower second-stage costs compared to their deterministic counterparts. This is expected, as the deterministic model does not account for variations in the actual quality of silage bales during the planning, and therefore has to compensate with feed concentrate or reserve silage when the silage bales have reduced quality. The quality objective is also consistently better for the stochastic model than for the deterministic model, although the difference is smaller than for the cost ob-

Table 8.19: Breakdown of the cost- and quality objectives for deterministic and stochastic solutions.

		Cost			Quality		
Instance	**Type**	**1st stage**	**2nd stage**	**Total**	**Q1***	**Q2****	**Total**
5D30S	Deterministic	0.07	0.74	0.82	0.43	0.03	0.46
	Stochastic	0.19	0.45	0.64	0.28	0.04	0.32
10D30S	Deterministic	0.14	1.79	1.93	0.56	0.07	0.63
	Stochastic	0.38	1.01	1.39	0.54	0.06	0.60
15D30S	Deterministic	0.21	2.45	2.66	0.92	0.12	1.04
	Stochastic	0.78	1.45	2.23	0.76	0.07	0.83

* Total penalty for deviating from target

** Total penalty for not ensuring a stable diet

jective. This indicates that by using additional feed concentrate, the model is able to construct feed compositions of relatively high quality. The variation penalty is quite low for all instances, showing that the model in most cases is able to construct a stable diet.

To illustrate a specific outcome, Figure 8.9 shows the daily feed compositions for the lactating animal group for test instance 5D30S under one of the simulated scenarios. As the figure shows, the deterministic model uses significantly higher amounts of feed concentrate on a daily basis, explaining the higher costs compared to the stochastic model. Moreover, on days one and five the deterministic model does not use any transported silage. This indicates that the silage transported on that day was not sufficient to meet the dry matter requirements for all animal groups, making it necessary to rely on feed concentrate and reserve silage. It is worth noting that both the deterministic and stochastic models yield similar amounts of dry matter in the feed compositions. This supports the findings from Table 8.19 indicating a smaller improvement in quality compared to cost.

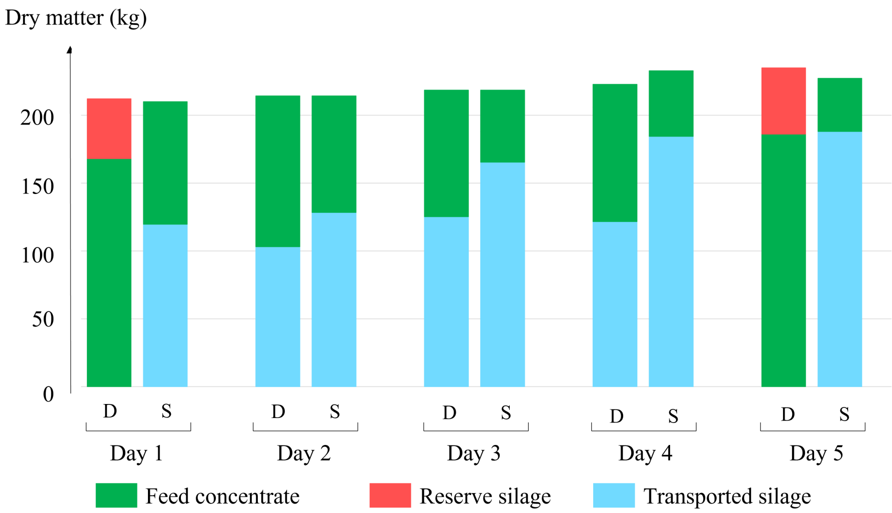


Figure 8.9: Feed composition in deterministic (D) and stochastic (S) solution for test instance 5D30S.

Figure 8.10 illustrates the transportation of silage bales for instance 5D30S, showing the increased transportation of silage bales in the stochastic model compared to the deterministic. Although this leads to a lower overall cost and a higher quality in our model, it is worth noting that this increased transportation has some negative effects. In scenarios where the actual silage quality is high, a considerable amount of the transported silage might end up being thrown away, leading to a waste of ingredients.

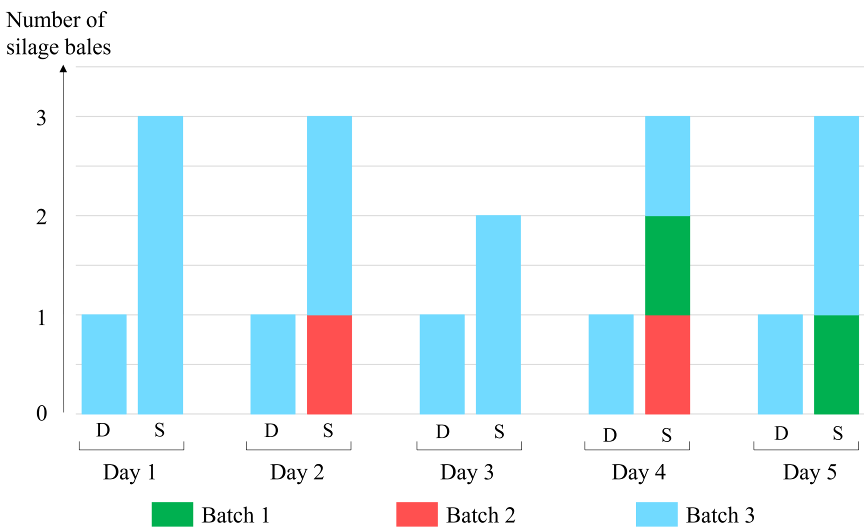


Figure 8.10: Silage bales transported in deterministic (D) and stochastic (S) solution for test instance 5D30S.

Overall, studying the impact on the cost and quality objectives, as well as the decisions made by the farmer, the results indicate that it is beneficial to plan with uncertainty. This makes it possible to avoid situations where it either is necessary to use an additional amount of costly ingredients such as feed concentrate or reserve silage, or compromise on the quality of the feed. The results illustrate the benefits of increasing the transportation of silage bales to mitigate the uncertainty in the nutritional content, even though it may lead to a higher degree of waste. To address both the uncertainty and waste concerns, potential strategies for future research are presented in Section 9.2.

8.5 The Value of Planning

This section studies the value of planning for a longer planning horizon, aiming to further address the goal of investigating the value of tactical planning with uncertainty. While Section 8.4 addresses the value of taking uncertainty into account, the focus in this section shifts towards evaluating the value of tactical planning. First, Section 8.5.1 illustrates the impact on the objective value when the length of the planning horizon is varied. Thereafter, Section 8.5.2 studies the impact on the cost and quality objectives, as well as the decisions made by the model.

8.5.1 Varying the Length of the Planning Horizon

To study the value of planning with a longer planning horizon, we employ the rolling horizon method. This involves testing the model performance by keeping the total length of the horizon constant, but varying the number of days the model optimizes for, hereby referred to as the central section. This approach allows us to examine the impact of looking further into the future when resources are limited.

We set the total length of the horizon to 20 days. This enables the model to solve instances over the full planning horizon close to optimal, making it easier to compare the models, while still being of adequate length to study the value of planning. The central section is set to 1, 10, and 20 days, and the forecasting section is set to zero. In addition, instead of setting the number of bales in each batch to be dependent on the length of the planning horizon as explained in Section 7.3, we limit the number of bales in each batch to 10. This is done to reflect the limited availability of silage bales over the total planning horizon of 20 days.

Table 8.20 presents the results from solving the model with the different lengths of the central section, where the models with a central section of 1, 10, and 20 days are referred to as 1R, 10R, and 20R, respectively. The findings show an improved objective value as the length of the central section increases. Specifically, the objective value is improved by 21.5% from 1R to 20R. This indicates a considerable value of lengthening the planning horizon.

Table 8.20: Results from solving the problem using a rolling horizon method with a central section of 1, 10, and 20 days.

Model	**Total Horizon**	**Central section**	**Obj.**
1R	20	1	6.27
10R	20	10	5.28
20R	20	20	4.92

8.5.2 Impact on Objectives and Decisions

The results obtained from Section 8.5.1 demonstrate that extending the planning horizon has a positive impact on the objective value. In this section, we study the cost and quality objectives, as well as the decision-making of the model, when solving models with different planning horizons. The aim is to provide a more comprehensive understanding of the underlying factors that contribute to the improvement observed when utilizing longer planning horizons.

Table 8.21 provides an overview of the cost and quality objectives for the three models introduced in Section 8.5.1. The cost objective is split into first- and second-stage costs, and the quality objective is split into the total penalty for deviating from the target of dry matter and nutrients (Q1), and the penalty for not ensuring a stable diet (Q2). The results shown in Table 8.21 reveal that the improvement in the objective function value primarily comes from improvements in the cost objective, particularly in terms of second-stage costs. As the length of the planning horizon is increased, the second-stage costs are significantly reduced, meaning that less reserve silage and feed concentrate are used. In contrast, the three models obtain relatively similar performance in the quality aspect. One interesting observation is that when the central section is 1, the models perform worse in terms of deviation from target, but better in terms of keeping a stable diet. This may suggest a trade-off between deviation from the target and diet stability, where the model prioritizes stable compositions when it is not able to be close to target. However, the three models overall demonstrate relatively comparable performance in terms of quality, suggesting that the lengthening of the planning horizon primarily affects the second-stage costs of the feeding plan.

Table 8.21: Breakdown of cost- and quality objectives for rolling horizon solutions 1R, 10R, 20R.

	Cost			**Quality**		
Model	**1st stage**	**2nd stage**	**Total**	**Q1***	**Q2****	**Total**
1R	0.54	4.21	4.75	1.44	0.07	1.51
10R	0.57	3.25	3.82	1.33	0.13	1.46
20R	0.57	2.86	3.43	1.35	0.13	1.48

*Total penalty for deviating from target

**Total penalty for not ensuring a stable diet

Figure 8.11 illustrates the aggregated cost objective over the total horizon of 20 days for the three models. The results reveal that models 1R and 10R initially have lower costs than model 20R. However, over time these models experience a substantial increase in costs, with the highest increase observed for model 1R. This indicates that while the models obtain high quality at a low cost at the beginning of the planning period, they struggle to maintain this over time. For model 20R, the costs remain relatively consistent, resulting in an overall lower aggregated cost compared to the other models.

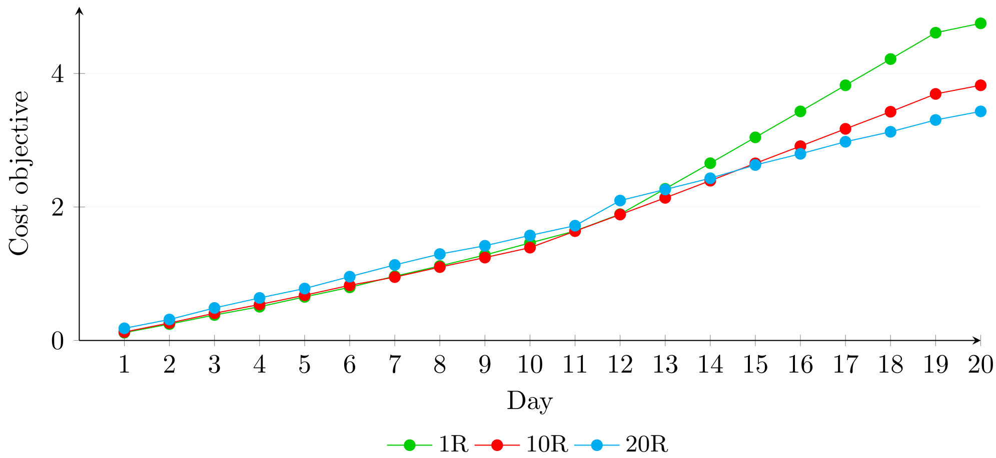


Figure 8.11: Aggregated cost objective when solving the model using a rolling horizon method with varying length of the central section. The green, red and blue line represents models with a central section of 1, 10 and 20 days, respectively.

The results observed in Figure 8.11 suggest that a shorter planning horizon could lead to inefficient allocation of resources. This is supported by Figure 8.12, which illustrates the daily transportation of silage bales throughout the planning horizon for the three models.

The figure reveals a clear trend in the transportation behavior of the models, where both models 1R and 10R transport many silage bales at the beginning of the planning horizon, but significantly fewer towards the end. In particular, from day 13 to day 19, model 1R does not transport any silage bales at all. This indicates that the model uses all its available silage bales by day 13 and is unable to transport any more until a new batch becomes available on day 20. Consequently, the model is forced to compensate with reserve silage and feed concentrate, resulting in significantly increased costs. However, when the model is forced to only use reserve silage and feed concentrate in the feed compositions, it is able to compose the exact same feed composition each day for each animal group. This may explain the low value in the variability objective (Q1) for 1R.

In contrast, model 20R demonstrates a more even distribution of bales throughout the planning horizon. Initially, the model transports a lower quantity of bales compared to 1R and 10R, which may be the reason why 20R incurs higher costs during the initial phase of the planning period, as transporting fewer bales can result in an increased demand for feed concentrate and reserve silage. However, due to the even distribution of resources, it is able to avoid situations where it becomes completely reliant on reserve silage and feed concentrate. Consequently, the model achieves lower overall costs as compared to the other models.

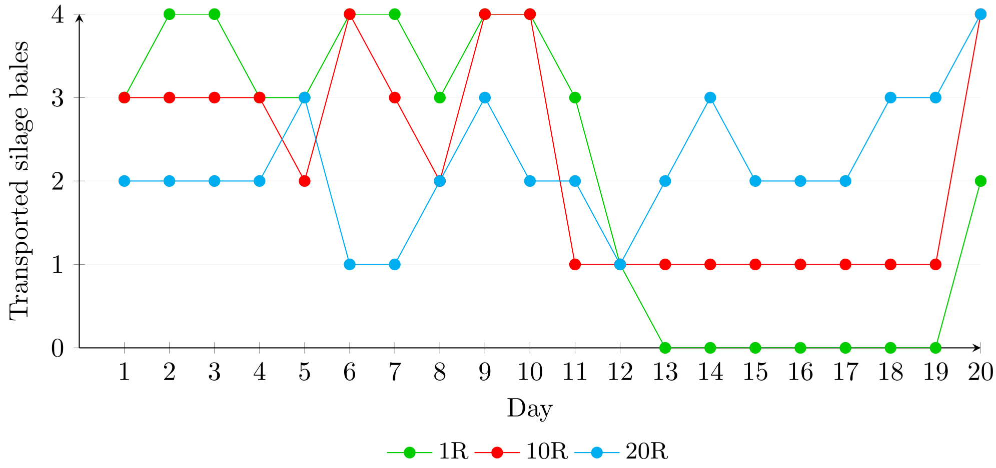


Figure 8.12: Transported silage bales when solving the model usina a rolling horizon method with varying length of the central section. The green, red and blue lines represent models that is solved with a central section of 1, 10 and 20 days, respectively.

The findings from this section indicate that planning for a longer planning horizon leads to a more efficient allocation of resources and a better ability to maintain cost control over an extended planning horizon. While a shorter planning horizon may provide immediate favorable outcomes in terms of cost and quality, it fails to account for future needs, leading to increased costs over time. This highlights the value of considering longer planning horizons when determining feeding plans, especially when there are limited resources available.

8.6 Effect of Reducing the Number of Days with Uncertainty

In Section 8.4, the value of uncertainty is studied, and the results indicate a significant improvement when the problem is solved with a stochastic model compared to a deterministic one. Thereafter, Section 8.5 illustrates the value of planning, showing that the length of the planning horizon has a significant impact on the choices that are made, and that solving the model with a longer planning horizon distributes the resources more evenly. However, as the number of days and scenarios increase, the model performance is significantly reduced, as evident from the findings presented in Section 8.2. Therefore, this section investigates the possibility of combining uncertainty and a longer planning horizon, by limiting the number of days where uncertainty is included.

In this section, the model presented in Section 6.4 is used, allowing the decision maker to only consider uncertainty on a subset of days during the planning horizon. To perform the analysis, we run several models for a fixed number of days in the planning horizon, with a varying number of days where uncertainty is taken into account. After the models have been solved, they are evaluated using the simulation-based approach presented in Section 6.5, where five simulations are run for each model. Finally, the average objective values in the simulated scenarios are compared, in addition to the optimality gap of the original models. The goal is to investigate the trade-off between the quality of the solution and the computational efficiency when uncertainty is included.

Figure 8.13 illustrates the results for test instance 10D30S. In the figure, the x-axis represents the number of days where uncertainty is considered, ranging from 1 to 10. The primary y-axis represents the simulated objective values, while the secondary y-axis represents the optimality gap (%). The average simulated objective values for all models are presented by the red line, while the gap is presented by the blue line.

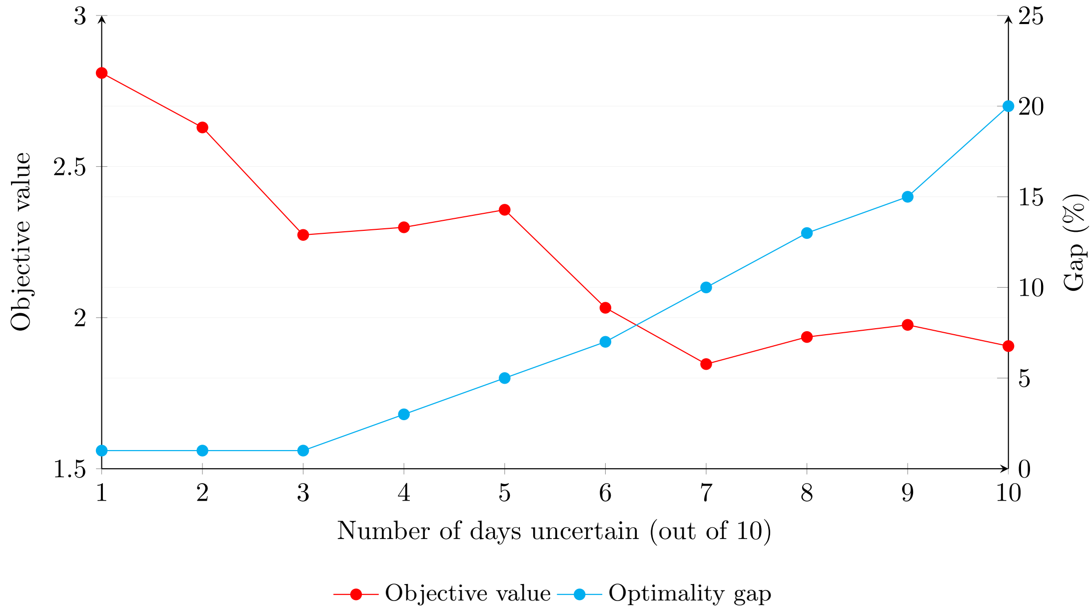


Figure 8.13: Effect on objective value and optimality gap when varying the number of days where uncertainty is taken into account. Effect on objective value and optimality gap when varying the number of days where uncertainty is taken into account.

As illustrated in Figure 8.13, the simulated objective value is significantly reduced when the number of days that takes uncertainty into account increases. This is expected due to the value of uncertainty studied in Section 8.4. Furthermore, as the number of days with uncertainty increase, so do the optimality gaps, reflecting the increased computational complexity as studied in Section 8.2.

It is interesting to look at these two metrics in relation. Figure 8.13 illustrates that the most substantial improvement in objective value occurs when the number of days increases from one to three and from five to seven. Furthermore, as the number of days with uncertainty increases beyond seven, the improvement in the objective values starts to level off. Taking the substantial increase in computational complexity as the number of days with uncertainty increases into account, these findings may suggest that it could be beneficial to not consider uncertainty during the entire planning horizon. Instead, by choosing a subset of days where uncertainty is considered, decision-makers may strike a balance between improved solutions and manageable computational complexity.

In conclusion, the findings presented in this section illustrate the potential benefits of only including uncertainty for a subset of days during the planning horizon, to reduce the computational complexity. These results open up possibilities for future research, which is further discussed in Section 9.2.

8.7 Trade-off Between Cost and Quality

In this section, we address the third goal for this thesis, namely investigating the trade-off between cost and quality. In the previous sections of this chapter, all models are solved with $\alpha = 0.5$, i.e an equal weighing of the two objective terms, cost and quality. However, in reality, one objective may be more important to prioritize than the other, likely affecting the decisions made when determining the feeding plan. In Section 8.7.1 we study the trade-off through a Pareto front analysis. Thereafter, in Section 8.7.2, we study the impact the prioritization has on the value of the objective terms and the decisions made.

8.7.1 Pareto Front Analysis

The Pareto front analysis is a tool for studying the trade-off between two objectives. Each point on the Pareto front represents a solution that is not dominated by any other solution, i.e a solution that cannot be further improved without compromising the performance of at least one other objective. It is important to note that as our model uses a weighted-sum method to address the two objectives, it may not identify all Pareto-optimal points on the front. However, it still offers valuable insights into the trade-offs between the different objectives.

Figure 8.14 illustrates the Pareto front for a test instance of 10 days and 30 scenarios. The figure shows that as α increases, the quality is improved while the costs are increased. From the figure, we see that the slope of the front is steeper along the quality axis than the cost axis. This means that it is possible to achieve substantially improved quality, with only a small increase in costs. For instance, as the value of α increase from 0.01 to 0.4, the quality improves by 42%, while the cost only increases by 8%. However, the figure also indicates that it becomes more costly to achieve an improvement in quality when the quality reaches a specific point. This can be seen by the substantial increase in cost compared to quality when α increases from 0.9 to 0.99. This illustrates the importance of weighing the two objectives in a way that strikes a desirable balance between the final cost and quality of the feed.

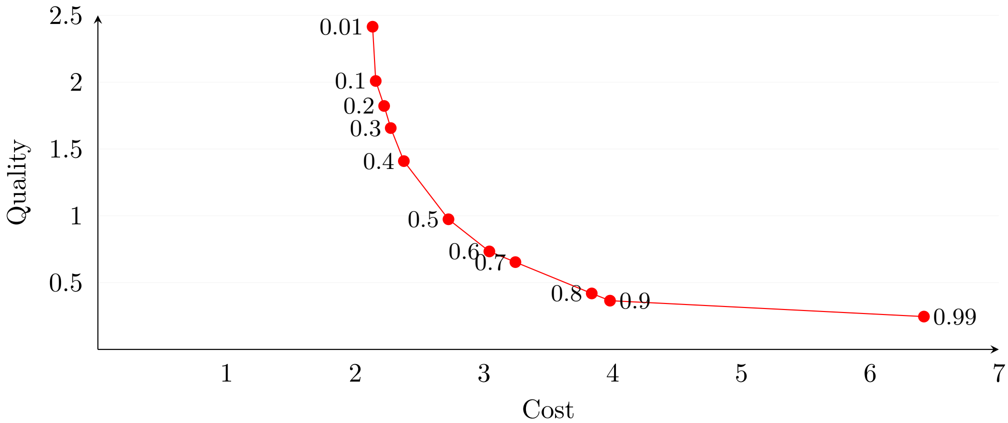


Figure 8.14: Pareto front for test instance 10D30S. The number at each of the points indicate the value of α that is used when solving the problem.

8.7.2 Impact on Objectives and Decisions

To further analyze the effect of different prioritization, this section investigates the impact on the different parts of the cost and quality objective. To perform this analysis, we focus on three of the instances presented in Figure 8.14; one mainly prioritizing cost ($\alpha = 0.2$), another equally weighing the two objectives ($\alpha = 0.5$), and lastly, one prioritizing quality ($\alpha = 0.8$). Table 8.22 presents the breakdown of the cost and quality objectives for the three instances. The cost objective consists of the first- and second-stage costs, while the quality objective consists of the penalty for deviating from the target (Q1) and the penalty for not having a consistent diet (Q2). Note that the values in the table are not multiplied with their corresponding weights α and $1 - \alpha$ to better be able to compare the results.

Table 8.22 shows that when the cost objective is prioritized ($\alpha = 0.2$), the second-stage costs are significantly smaller than for the other models, while the quality objective is higher. This indicates that in situations where the silage bale quality is insufficient, the model rather obtains a lower quality of the feed composition than supplying with additional feed concentrate or reserve silage. When the quality objective is prioritized ($\alpha = 0.8$), the opposite effect can be observed. The second-stage costs are higher, reflecting a higher usage of feed concentrate and reserve silage to ensure lower deviations from the target and improved diet consistency. Lastly, when the two objectives are equally prioritized, a trade-off between the two objectives is made to a larger extent. As shown in Table 8.22, the values for all metrics are between the values when $\alpha = 0.2$ and $\alpha = 0.8$, indicating a balanced consideration of both objectives.

Table 8.22: Breakdown of the cost- and quality objective for test instance 10D30S with different values of α.

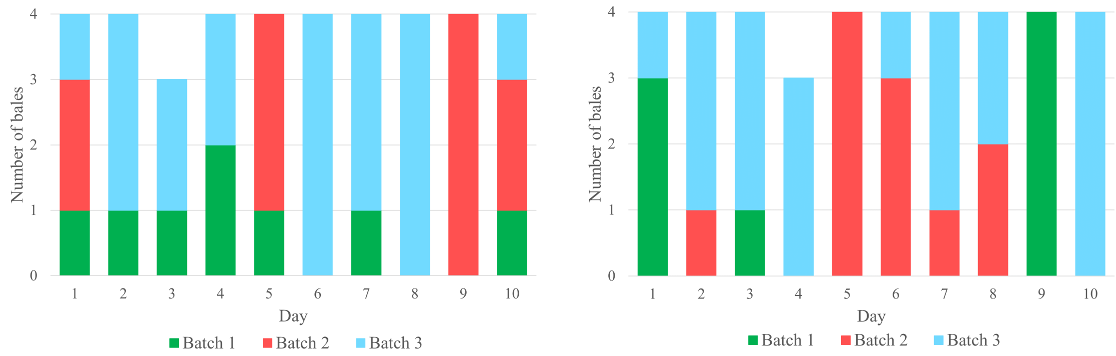
	Cost*			**Quality***		
α	**1st stage**	**2nd stage**	**Total**	**Q1****	**Q2*****	**Total**
0.2	1.01	1.22	2.22	1.70	0.12	1.82
0.5	1.02	1.71	2.72	0.88	0.09	0.97
0.8	1.02	2.82	3.84	0.34	0.08	0.42

* The objectives are not multiplied with α and $1 - \alpha$

** Total penalty for deviating from target

*** Total penalty for not ensuring a stable diet

One interesting observation is that the first-stage costs are comparable for all three instances, indicating that the instances make similar decisions with regard to the silage bales transported every day. Figure 8.15 illustrates the silage bales transported every day during the planning horizon for the models mainly prioritizing cost and quality. Figure 8.15a illustrates the transportation when $\alpha = 0.2$ and Figure 8.15b illustrates the transportation when $\alpha = 0.8$. The figures show that the models transport different silage bales on different days, likely related to the varying quality of the bales. However, the total number of silage bales, as well as the total number from each batch is approximately the same for the three models. Consequently, the trade-off seems to be more concentrated on the actual feed composition and how much of what ingredient to use, rather than the transportation of silage. This observation is likely related to the fact to the second-stage decisions provide an opportunity to obtain a high quality by using feed concentrate and reserve silage, even when the silage bale is of poor quality. However, this comes at a significant cost, and the usage of these ingredients will consequently depend on the decision-makers' prioritization between the two objectives.



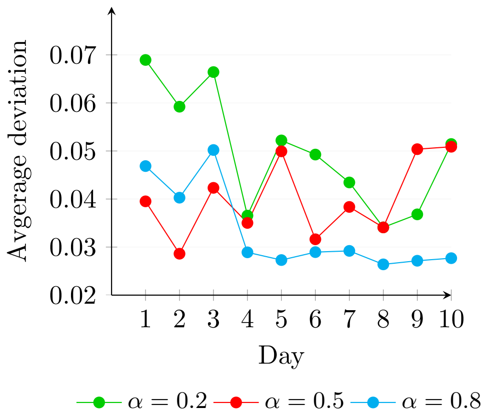
(a) $\alpha = 0.2$ (b) $\alpha = 0.8$

Figure 8.15: Number of silage bales transported for test instance 10D30S with different prioritization of the objectives.

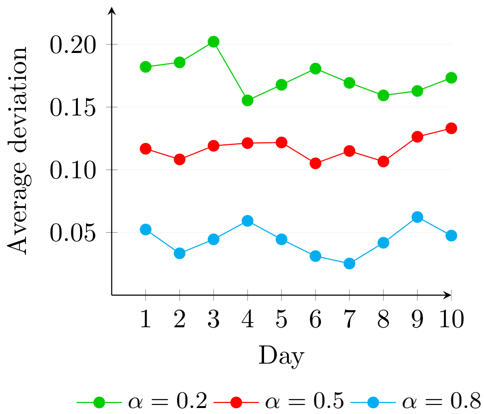
Table 8.22 also illustrates a significant variation in the penalty for deviating from the target (Q1) as the value of α varies. This penalty is visualized in Figure 8.16, illustrating the average absolute deviation from the target for dry matter, protein and NDF for the three instances. As expected, when quality is prioritized ($\alpha = 0.8$), a lower deviation from the target for all three requirements is achieved. Oppositely, when the cost is prioritized ($\alpha = 0.2$), the deviation is higher, whereas ($\alpha = 0.5$) illustrates deviations in between.

For the dry matter content, illustrated in Figure 8.16a, all three instances obtain a relatively low deviation, with deviations ranging from 4% to 7%. However, Figure 8.16b shows that all instances have higher deviations from the NDF target. Specifically, when costs are prioritized, the model experiences deviations of up to 20%, indicating that it is more difficult and costly to satisfy this requirement. Lastly, for the protein requirement, illustrated in Figure 8.16c, all instances have low deviations from the target. Although the deviations are significantly increased when the cost is prioritized ($\alpha = 0.2$), the deviations are still low, ranging from 3% to 6%.

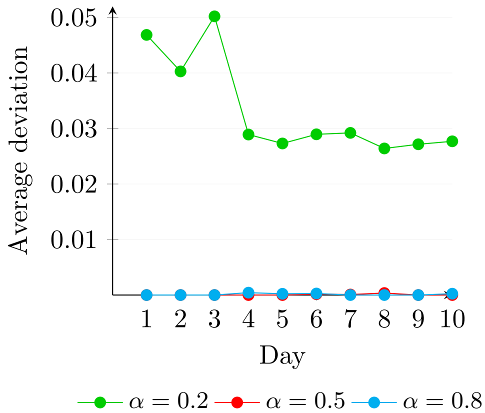
These results indicate that a major part of the deviation penalty comes from deviation from the NDF target. In practice, this suggests that farmers could benefit from focusing on strategies to meet the NDF requirement more effectively. However, it is important to note that these specific results are the outcome of generated data, as explained in Chapter 7. Therefore, the results and the recommendations may differ from reality. Nevertheless, it provides insights into how prioritizing the objectives impacts overall deviations and affects the decision-making processes.



(a) Average absolute deviation from target in dry matter content



(b) Average absolute deviation from target in NDF content.



(c) Average absolute deviation from target in protein content.

Figure 8.16: Average absolute deviation from target of dry matter and nutritional content for test instance 10D30S when $\alpha = 0.2$, $\alpha = 0.5$ and $\alpha = 0.8$.

In conclusion, the weighing of the cost and quality objective has a considerable impact on the objectives and decisions of the model. An interesting finding from this analysis is that although the cost and quality objectives are greatly affected by the prioritization between the two, most of the changes in the decisions are related to the second-stage decisions. In practice, this means that the trade-off between cost and quality is most important to consider during the daily feed construction, and does not affect the tactical planning to a great extent. However, the insights gained from this analysis contribute to understanding the trade-off between cost and quality, and how it impacts the decisions that are made in the context of animal feeding.

Chapter 9

Concluding Remarks and Future Research

This chapter highlights the findings of this thesis and discusses some opportunities for future research. First, a summary of the findings and the concluding remarks are presented in Section 9.1. As the model still requires further modifications and work to be applicable in real life, we propose several areas for improvement and development. These possibilities for future research are presented in Section 9.2.

9.1 Concluding Remarks

This thesis studies the Stochastic Cattle Feeding Planning Problem (SCFPP). The SCFPP is formulated as a two-stage stochastic problem where the first-stage decisions decide how to distribute silage bales over the planning horizon. The second-stage decisions are operational decisions related to the daily construction of feed compositions. These decisions are made after the actual quality of the silage bales is known, and involve choosing a combination of silage and feed concentrate that satisfies the animals' daily feeding requirements. The problem is a multi-objective problem, aiming to both minimize costs and ensure a correct and stable quality of the diet. The purpose of the thesis is to investigate the value of using mathematical optimization to help farmers overcome challenges in determining a feeding plan for dairy cattle. This objective is pursued by working towards three goals.

The first goal is to develop a solution method that is capable of effectively solving the SCFPP. In the preparatory project for this thesis (Fosen & Nygaard, 2022), the Cattle Feeding Planning Problem (CFPP) proved to be computationally heavy to solve, and as uncertainty is included in the SCFPP, the complexity of the problem is further increased. Therefore, the L-shaped method, along with several acceleration methods are investigated to solve the two-stage stochastic formulation of the SCFPP. The proposed accelerated L-shaped method incorporates multiple cuts for each iteration, solves the problem with a two-phase approach, generates Pareto-optimal cuts, and approximates the solution of the master problem. The computational results show a considerable effect of accelerating the L-shaped method, where the proposed solution method is able to obtain a gap lower than 10% for the largest instances of 20 days and 100 scenarios, while the Gurobi solver terminates with a gap of 100%.

The second goal is to investigate the value of planning with uncertainty and the potential to improve decision-making for dairy farmers by doing so. Although further work is necessary to make the model applicable in real-life, the computational study reveals significant benefits of tactical planning with uncertainty. By incorporating uncertainty into the planning process, both cost reduction and improved feed quality are achieved compared to feeding plans made with a deterministic model. Specifically, the computational study reveals an improvement in the objective function of up to 25% when uncertainty is considered. Furthermore, the computational study highlights the impact of the length of the planning horizon, where shorter planning horizons prioritize immediate cost and quality benefits without considering future needs. In contrast, longer planning horizons lead to a more balanced distribution of ingredients, resulting in a more consistent and cost-effective feeding approach for the dairy herd. However, solving the model for long planning horizons with uncertainty leads to computational challenges. Therefore, the potential benefits of only including uncertainty for a subset of days during the planning horizon are investigated. Overall, the computational results successfully demonstrate a high potential of planning with uncertainty and emphasize the value of further development and investigation of systems that enables farmers to do so in practice.

Lastly, the third goal is to study the trade-off between cost and quality and develop a model that allows the decision-maker to prioritize between the two. This is achieved by incorporating the two objectives through a weighted sum approach, whereas the weighting factor represents the relative importance of the two objectives. The trade-off is analyzed by performing a Pareto-front analysis, as well as a detailed examination of the solutions and the decisions obtained by the model when the objectives are prioritized differently. The analysis reveals a significant trade-off between cost and quality, where the weighing of the objectives has a particularly large impact on the second-stage decisions. These findings provide valuable insights into the trade-offs between cost and quality, emphasizing the importance of carefully considering this trade-off when developing feeding plans for dairy cattle.

In conclusion, this thesis successfully achieves its purpose of investigating the value of using mathematical optimization to address key challenges farmers face when determining a feeding plan. By working towards three specific goals, the thesis develops a solution method that more effectively handles the complexity of the SCFPP, demonstrates the value of tactical planning with uncertainty, and investigates the trade-off between cost and quality. While the model requires further modifications for practical implementation, the findings from this thesis illustrate a significant potential of using mathematical optimization to help farmers make improved decisions for their feeding plans.

9.2 Future Research

In this section, future research opportunities for the SCFPP are presented. Section 9.2.1 discusses potential areas of research to improve the real-life applicability of the SCFPP. This involves developing input data that more accurately reflects reality, addressing simplifying assumptions that have been made in the CFPP, and expanding the model to also handle waste concerns. Furthermore, although the solution method presented in this thesis performs significantly better than the original Gurobi solver, it is only able to solve for a limited length of the planning horizon. Therefore, Section 9.2.2 presents possible future research to further improve the solution method of the SCFPP, enabling it to solve the model for a longer planning horizon.

9.2.1 Improving the Real-Life Applicability of the SCFPP

As previously addressed, this thesis relies on a generated data set, which does not fully reflect real life. This is likely to have an impact on the results and may reduce the accuracy of the findings. Therefore, one suggested area of future research involves improving the data foundation for the problem. This may involve improving the data related to the feeding requirements by including a wider set of requirements, as well as accounting for individual differences between animals in the animal groups. Additionally, this thesis describes the uncertainty in the silage bales using a triangular distribution, which is a simplification of reality. Conducting a more comprehensive analysis of the variation in the actual quality of silage bales, and developing a more correct distribution would improve the accuracy of the model, and possibly have an effect on the results.

Furthermore, throughout this thesis, assumptions have been made to simplify the problem. For instance, we assume that all ingredients are loaded into the mixing machine at the same time. However, as described in Section 2.2, silage bales are often loaded sequentially, which complicates the mixing problem. A possible way to extend the SCFPP is therefore to incorporate the sequencing of silage bales during the mixing. This puts further limitations on the combination of ingredients that can be used in the feed compositions, which may have an impact on the final composition and nutritional value of the feed. Including this in the model would improve the credibility of the model and make the solutions more relevant in real-life.

Lastly, as the results from the computational study show, the model presented in this report accounts for the uncertainty by transporting additional silage bales. Although this leads to lower costs and better quality of the feed composition, it may also lead to a higher degree of waste in situations where the silage bales are of sufficient quality. As this is not desirable in real-life, a suggestion for future research is to modify and extend the model to also handle concerns regarding waste. For instance, dynamic re-planning can be explored as a strategy to adapt the feeding plan based on unexpected events. Furthermore, advancing the inventory aspects of the SCFPP model can contribute to minimizing waste and optimizing resource utilization. A possible modification is to allow silage bales that are transported on a given day, but not used, to be stored in the kitchen storage and used the next day. Incorporating such features to improve resource utilization and reduce waste would improve the applicability of the model.

9.2.2 Develop Solution Methods for Extended Planning Horizons

Although the accelerated L-shaped method proves to be significantly more effective than the standard Gurobi solver, it is only able to solve instances with a planning horizon of up to 20 days while keeping the gap relatively small. However, this thesis demonstrates the advantages of a longer planning horizon, primarily because it leads to a more even distribution of limited resources. Therefore, we suggest further research related to the solution method of the SCFPP, focusing on methods that are capable of solving the problem for a longer planning horizon.

One possibility is to utilize the rolling horizon method with deterministic forecasting sections. Findings from Section 8.6 suggest an approach where the uncertainty is only incorporated on a subset of days, thereby reducing the computational complexity of the model. However, to be able to both address uncertainty throughout the entire horizon and consider a longer planning horizon, we suggest employing a rolling horizon framework. This method can involve shorter central sections taking uncertainty into account, accompanied by longer and deterministic forecasting sections to ensure an even distribution of the limited resources. This can enable the model to solve the model accurately for longer periods by reducing the complexity in each solution process, making it an interesting potential for future research.

Bibliography

Amin, S. H., Mulligan-Gow, S., & Zhang, G. (2019). Selection of food items for diet problem using a multi-objective approach under uncertainty. In *Application of decision science in business and management* (p. 181). IntechOpen.

Bas, E. (2014). A robust optimization approach to diet problem with overall glycemic load as objective function. *Applied Mathematical Modelling*, *38*(19-20), 4926–4940.

Bellingeri, A., Gallo, A., Liang, D., Masoero, F., & Cabrera, V. E. (2020). Development of a linear programming model for the optimal allocation of nutritional resources in a dairy herd. *Journal of Dairy Science*, *103*(11), 10898–10916.

Birge, J. R., & Louveaux, F. (2009). L-shaped method for two-stage stochastic programs with recourse. In C. A. Floudas & P. M. Pardalos (Eds.), *Encyclopedia of optimization* (pp. 1943–1945). Springer US. https://doi.org/10.1007/978-0-387-74759-0_351

Birge, J. R., & Louveaux, F. (2011). *Introduction to stochastic programming*. Springer Science & Business Media.

Birge, J. R., & Louveaux, F. V. (1988). A multicut algorithm for two-stage stochastic linear programs. *European Journal of Operational Research*, *34*(3), 384–392.

Cakrak, M., & Cimen, E. B. (2017). A novel approach for Stigler's diet problem in genetic algorithm. *Proceedings of the 9th International Conference on Information Management and Engineering*, 201–206.

Chen, J. T. (1973). Quadratic programming for least-cost feed formulations under probabilistic protein constraints. *American Journal of Agricultural Economics*, *55*(2), 175–183. Retrieved 14th November 2022, from http://www.jstor.org/stable/1238434

Crabtree, J. (1981). The appraisal of machinery investment. *Journal of Agricultural Economics*, *32*(3), 365–375.

Dean, G., Bath, D., & Olayide, S. (1969). Computer program for maximizing income above feed cost from dairy cattle. *Journal of Dairy Science*, *52*(7), 1008–1016.

Department of Agriculture, Fisheries and Forestry and Dairy Australia. (2013a). Feed intake. http://northernaustraliandairyhub.com.au/wp-content/uploads/2014/11/N03_Feed_Intake.pdf

Department of Agriculture, Fisheries and Forestry and Dairy Australia. (2013b). Managing for healthy rumen function. http://northernaustraliandairyhub.com.au/wp-content/uploads/2014/11/N02_Managing_for_healthy_rumen_function.pdf

Flaten, O., & Lien, G. (2007). Stochastic utility-efficient programming of organic dairy farms. *European Journal of Operational Research*, *181*(3), 1574–1583.

Fomeni, F. D. (2018). A multi-objective optimization approach for the blending problem in the tea industry. *International Journal of Production Economics*, *205*, 179–192.

Fosen, M., & Nygaard, E. (2022). *A multi-objective approach to the dairy cattle feeding planning problem* [Available upon request].

Garille, S. G., & Gass, S. I. (2001). Stigler's diet problem revisited. *Operations Research*, *49*(1), 1–13.

Gephart, J. A., Davis, K. F., Emery, K. A., Leach, A. M., Galloway, J. N., & Pace, M. L. (2016). The environmental cost of subsistence: Optimizing diets to minimize footprints. *Science of the Total Environment*, *553*, 120–127.

Glen, J. J. (1980). A mathematical programming approach to beef feedlot optimization. *Management Science*, *26*(5), 524–535.

Glen, J. J. (1987). Mathematical models in farm planning: A survey. *Operations Research*, *35*(5), 641–666.

Gómez-Pantoja, J. Á., Salazar-Aguilar, M. A., & González-Velarde, J. L. (2021). The food bank resource allocation problem. *Top*, *29*(1), 266–286.

Goswami, S., Chaturvedi, A., Chatterji, S., Patil, N., Sen, T., Hajare, T., & Gawande, R. (2013). Least cost diet plan of cows for small dairy farmers of central india.

Gupta, R., Chandan, M., & Kuntal, R. S. (2013). Heuristic approaches in solving non linear model of livestock ration formulation. *International Journal of Engineering Sciences & Emerging Technologies*, *6*, 37–48.

Hardaker, J. B., Lien, G., Anderson, J. R., & Huirne, R. B. (2015). *Coping with risk in agriculture: Applied decision analysis*. Cabi.

Kikuhara, K., Kumagai, H., & Hirooka, H. (2009). Development and evaluation of a simulation model for dairy cattle production systems integrated with forage crop production. *Asian-Australasian Journal of Animal Sciences*, *22*(1), 57–71.

King, A. J., & Wallace, S. W. (2012). *Modeling with stochastic programming*. Springer Science & Business Media.

Kuudela, J., & Popela, P. (2017). Warm-start cuts for generalized benders decomposition. *Kybernetika*, *53*(6), 1012–1025.

Landbruksdirektoratet. (2022). *Norsk landbruk - tall og fakta*. Retrieved 7th December 2022, from https://www.landbruksdirektoratet.no/nb/norsk-landbruk-tall-og-fakta

Magnanti, T., & Wong, R. (1981). Accelerating benders decomposition: Algorithmic enhancement and model selection criteria. *Operations Research*, *29*, 464–484.

McDaniel, D., & Devine, M. (1977). A modified benders' partitioning algorithm for mixed integer programming. *Management Science*, *24*(3), 312–319.

Meyer, C., & Newett, R. (1970). Dynamic programming for feedlot optimization. *Management Science*, *16*(6), B–410.

Moghaddam, K. S., & DePuy, G. W. (2011). Farm management optimization using chance constrained programming method. *Computers and Electronics in Agriculture*, *77*(2), 229–237.

Nordic Feed Evaluation System. (2023). Norfor feedstuff search 1.24.0.691. http://feedstuffs.norfor.info/

Notte, G., Cancela, H., Pedemonte, M., Chilibroste, P., Rossing, W., & Groot, J. C. (2020). A multi-objective optimization model for dairy feeding management. *Agricultural Systems*, *183*, 102854.

Opplysningskontoret for Meieriprodukter. (2022). *Melkeproduksjon*. Retrieved 7th December 2022, from https://www.melk.no/Melkekilden/Melkeproduksjon/Noekkeltall/Hvor-mange-melkegaarder-har-vi-i-Norge

Osman, B. I., & Sufahani, S. F. (2022). Nutritious menu planning for sinusitis patient in malaysian through optimization approach. *Proceedings of the Third International Conference on Trends in Computational and Cognitive Engineering*, 31–44.

Patil, V., Gupta, R., Duraisamy, R., & Kuntal, R. S. (2022). Nutrient requirement equations for indian goat by multiple regression analysis and least cost ration formulation using a linear and non-linear stochastic model. *Journal of Animal Physiology and Animal Nutrition*, *106*(5), 968–977.

Peña, T., Lara, P., & Castrodeza, C. (2009). Multiobjective stochastic programming for feed formulation. *Journal of the Operational Research Society*, *60*(12), 1738–1748.

Perrykkad, A., Ernst, A. T., & Krishnamoorthy, M. (2022). A simultaneous magnanti-wong method to accelerate benders decomposition for the metropolitan container transportation problem. *Operations Research*, *70*(3), 1531–1559.

Polimeno, F., Rehman, T., Neal, H., & Yates, C. (1999). Integrating the use of linear and dynamic programming methods for dairy cow diet formulation. *Journal of the Operational Research Society*, *50*(9), 931–942.

Rahman, R. A., Ramli, R., Jamari, Z., & Ku-Mahamud, K. R. (2015). Evolutionary algorithm approach for solving animal diet formulation. *Proc. 5th Int. Conf. Comput. Informatics, ICOCI*, (32), 274–279.

Rahman, S. A., & Bender, F. (1971). Linear programming approximation of least-cost feed mixes with probability restrictions. *American Journal of Agricultural Economics*, *53*(4), 612–618.

Rahmaniani, R., Crainic, T. G., Gendreau, M., & Rei, W. (2017). The benders decomposition algorithm: A literature review. *European Journal of Operational Research*, *259*(3), 801–817.

Regjeringen. (2021). Matsikkerhet. https://www.regjeringen.no/no/tema/mat-fiske-og-landbruk/mat/innsikt/matsikkerhet/id2357158/

Rodríguez, J. A., Anjos, M. F., Côté, P., & Desaulniers, G. (2021). Accelerating benders decomposition for short-term hydropower maintenance scheduling. *European Journal of Operational Research*, *289*(1), 240–253.

Shalloo, L., Dillon, P., Rath, M., & Wallace, M. (2004). Description and validation of the moorepark dairy system model. *Journal of Dairy Science*, *87*(6), 1945–1959.

Sirisatien, D., Wood, G. R., Dong, M., & Morel, P. C. (2009). Two aspects of optimal diet determination for pig production: Efficiency of solution and incorporation of cost variation. *Journal of Global Optimization*, *43*(2), 249–261.

SSB. (2022). *Gardsbruk, jordbruksareal og husdyr*. Retrieved 7th December 2022, from https://www.ssb.no/jord-skog-jakt-og-fiskeri/jordbruk/statistikk/gardsbruk-jordbruksareal-og-husdyr

Statistisk sentralbyrå. (2023). Gårdsbruk, jordbruksareal og husdyr. https://www.ssb.no/jord-skog-jakt-og-fiskeri/jordbruk/statistikk/gardsbruk-jordbruksareal-og-husdyr#om-statistikken

Stigler, G. J. (1945). The cost of subsistence. *Journal of Farm Economics*, *27*(2), 303–314. Retrieved 25th May 2023, from http://www.jstor.org/stable/1231810

Sufahani, S. F., & Ismail, Z. (2015). Planning a nutritious and healthy menu for malaysian school children aged 13-18 using" delete-reshuffle algorithm" in binary integer programming. *Journal of Applied Sciences*, *15*(10), 1239.

TINE Rådgivning. (2012). Fôringsstrategier. https://medlem.tine.no/gard-og-drift/foring

Tozer, P. R. (2000). Least-cost ration formulations for holstein dairy heifers by using linear and stochastic programming. *Journal of Dairy Science*, *83*(3), 443–451.

Trebeck, D. B., & Hardaker, J. B. (1972). The integrated use of simulation and stochastic programming for whole farm planning under risk. *Australian Journal of Agricultural Economics*, *16*(2), 115–126.

Udomsungworagul, A., & Charnsethikul, P. (2018). A hybrid dantzig-wolfe, benders decomposition and column generation procedure for multiple diet production planning under uncertainties. *IOP Conference Series: Materials Science and Engineering*, *332*(1), 012038.

Van de Panne, C., & Popp, W. (1963). Minimum-cost cattle feed under probabilistic protein constraints. *Management Science*, *9*(3), 405–430.

Van Slyke, R. M., & Wets, R. (1969). L-shaped linear programs with applications to optimal control and stochastic programming. *SIAM Journal on Applied Mathematics*, *17*(4), 638–663.

West, J., Mullinix, B., & Sandifer, T. (1991). Changing dietary electrolyte balance for dairy cows in cool and hot environments. *Journal of Dairy Science*, *74*(5), 1662–1674.

Zarandi, M. M. F. (2010). *Using decomposition to solve facility location/fleet management problems* (Master's thesis). University of Toronto. https://tspace.library.utoronto.ca/handle/1807/24945

Zhang, X., Shen, H., & Wu, Z. (2021). Feed formula optimization based on improved tabu search algorithm. *International Conference on Neural Computing for Advanced Applications*, 446–457.

Zhao, J., Gendreau, M., Potvin, J.-Y., & Rousseau, L.-M. (2017). Hybrid large neighborhood search for the heterogeneous fixed fleet open vehicle routing problem. *European Journal of Operational Research*, *262*(1), 81–93.

Appendix A

Compressed Model

Sets and Indices

$\mathcal{A}$	Set of animal groups, $a \in \mathcal{A}$
$\mathcal{D}$	Set of days in the planning horizon, $d \in \mathcal{D}$
$\mathcal{I}$	Set of silage bale batches, $i \in \mathcal{I}$
$\mathcal{F}$	Set of feed concentrate types, $f \in \mathcal{F}$
$\mathcal{K}$	Set of kitchens, $k \in \mathcal{K}$
$\mathcal{N}$	Set of nutrients, $n \in \mathcal{N}$
$\mathcal{S}$	Set of possible scenarios, $s \in \mathcal{S}$
$\mathcal{B}_i$	Set of silage bales from batch i
$\mathcal{D}_i$	Set of days when silage bale batch i is available, $\mathcal{D}_i \subseteq \mathcal{D}$
$\mathcal{I}_d$	Set of silage bale batches available on day d, $\mathcal{I}_d \subseteq \mathcal{I}$

Parameters

C^B_{di}	Cost per silage bale from batch i on day d
C^R	Cost of using reserve silage
C^F_f	Cost per kg used of feed concentrate type f
D^A_i	The day silage bale batch i becomes available
F^B_{bdis}	Amount (kg) dry matter in silage bale b from batch i in scenario s on day d
F^F_f	Fraction of dry matter in feed concentrate type f
F^{MAX}_a	Maximum fraction of dry matter from feed concentrate in the feed composition for animal group a
K_k	Daily storage capacity in kitchen k
N^B_{bdins}	Nutritional density of nutrient n (per kg dry matter) in silage bale b from batch i in scenario s on day d
N^F_{fn}	Nutritional density of nutrient n (per kg dry matter) in feed concentrate type f
N^R_n	Nutritional density of of nutrient n (per kg dry matter) in reserve silage bale
L^M_{ad}	Minimum dry matter (kg) for animal group a on day d
U^M_{ad}	Maximum dry matter (kg) for animal group a on day d
T^M_{ad}	Target dry matter (kg) for animal group a on day d
L^N_{adn}	Minimum amount (per kg dry matter) of nutrient n for animal group a on day d
U^N_{adn}	Maximum amount (per kg dry matter) of nutrient n for animal group a on day d
T^N_{adn}	Target amount (per kg dry matter) of nutrient n for animal group a on day d
P_s	Probability of scenario s
P^M	Penalty for deviating from dry matter target
P^{IN}	Penalty for deviating from nutrient target when the content is between target and minimum or maximum value
P^{ON}	Penalty for deviating from nutrient target when the content is outside the minimum or maximum limit
R^M_a	Relative deviation from dry matter target for animal a on the last day before planning horizon start
R^N_{an}	Relative deviation from target for nutrient n for animal a on the last day before planning horizon start

Decision Variables

x_{abdkis}	Amount (kg) dry matter from silage bale b from silage bale batch i used in the feed composition for animal group a on day d in kitchen k in scenario s
r_{ads}	Dry matter (kg) from reserve bale used in feed composition for animal group a on day d in scenario s
y_{adfs}	Amount (kg) of feed concentrate type f used in feed composition for animal group a on day d in scenario s
d^M_{ads}	Deviation in dry matter content in feed composition for animal group a on day d in scenario s
d^N_{adns}	Deviation in nutritional content of nutrient n for animal group a on day d in scenario s
p^M_{ads}	Penalty related to deviation from dry matter target in feed composition for animal a on day d in scenario s
p^N_{adns}	Penalty related to deviation from target for nutrient n in feed composition for animal a on day d in scenario s
$v^M_{a(d-1)ds}$	Variation in deviation from target for dry matter content in two consecutive feed compositions, from day $(d-1)$ to day d for animal group a
$v^N_{a(d-1)dns}$	Variation in deviation from target for nutrient n in two consecutive feed compositions, from day $(d-1)$ to day d for animal group a in scenario s
m_{bdik}	1 if silage bale b from batch i is transported to kitchen k on day d, 0 otherwise
w_{adk}	1 if the feed composition for animal group a on day d is produced in kitchen k, 0 otherwise

Weight Parameters

α	Weight of quality objective

Quality Objective

$$f^Q = \min \sum_{s\in\mathcal{S}} P_s(\sum_{a\in\mathcal{A}}\sum_{d\in\mathcal{D}}((p^M_{ads} + \sum_{n\in\mathcal{N}} p^N_{adns}) + (v^M_{a(d-1)ds} + \sum_{n\in\mathcal{N}} v^N_{a(d-1)dns})) \tag{A.1}$$

Cost Objective

$$f^C = \min \sum_{d\in\mathcal{D}}\sum_{i\in\mathcal{I}_d}\sum_{b\in\mathcal{B}_i}\sum_{k\in\mathcal{K}} C^B_{di} m_{bdik} + \sum_{s\in S} P_s(\sum_{a\in\mathcal{A}}\sum_{d\in\mathcal{D}}(\sum_{f\in\mathcal{F}} C^F_f y_{adfs} + C^R r_{ads})) \tag{A.2}$$

Weighted Sum Objective

$$Z = \min\ \alpha f^Q + (1-\alpha) f^C \tag{A.3}$$

Constraints

$$\sum_{k\in\mathcal{K}} w_{adk} = 1 \qquad a\in\mathcal{A}, d\in\mathcal{D} \quad (A.4)$$

$$\sum_{d\in\mathcal{D}_i}\sum_{k\in\mathcal{K}} m_{bdik} \leq 1 \qquad i\in\mathcal{I}, b\in\mathcal{B}_i \quad (A.5)$$

$$\sum_{d'=1}^{d}\sum_{k\in\mathcal{K}} (m_{(b-1)d'ik} - m_{bd'ik}) \geq 0 \qquad d\in\mathcal{D}, i\in\mathcal{I}_d, b\in\mathcal{B}_i\setminus\{1\} \quad (A.6)$$

$$\sum_{i\in\mathcal{I}_d}\sum_{b\in\mathcal{B}_i} m_{bdik} \leq K_k \qquad d\in\mathcal{D}, k\in\mathcal{K} \quad (A.7)$$

$$x_{abdiks} \leq \min\{U^M_{ad}, F^B_{bdis}\} w_{adk} \qquad a\in\mathcal{A}, d\in\mathcal{D}, i\in\mathcal{I}_d, b\in\mathcal{B}_i, k\in\mathcal{K}, s\in\mathcal{S} \quad (A.8)$$

$$\sum_{a\in\mathcal{A}} x_{abdiks} \leq F^B_{bdis} m_{bdik} \qquad d\in\mathcal{D}, i\in\mathcal{I}_d, b\in\mathcal{B}_i, k\in\mathcal{K}, s\in\mathcal{S} \quad (A.9)$$

$$\sum_{k\in\mathcal{K}}\sum_{i\in\mathcal{I}_d}\sum_{b\in\mathcal{B}_i} x_{abdiks} + \sum_{f\in\mathcal{F}} F^F_f y_{adfs} + r_{ads} = T^M_{ad} + d^M_{ads} \qquad a\in\mathcal{A}, d\in\mathcal{D}, s\in\mathcal{S} \quad (A.10)$$

$$L^M_{ad} \leq T^M_{ad} + d^M_{ads} \leq U^M_{ad} \qquad a\in\mathcal{A}, d\in\mathcal{D}, s\in\mathcal{S} \quad (A.11)$$

$$\sum_{k\in\mathcal{K}}\sum_{i\in\mathcal{I}_d}\sum_{b\in\mathcal{B}_i} N^B_{bdns} x_{abdiks} + \sum_{f\in\mathcal{F}} N^F_{fn} F^F_f y_{adfs} + N^R_n r_{ads} = T^N_{adn} + d^N_{adns} \qquad a\in\mathcal{A}, d\in\mathcal{D}, n\in\mathcal{N}, s\in\mathcal{S} \quad (A.12)$$

$$\sum_{f\in\mathcal{F}} F^F_f y_{adfs} \leq F^{MAX}_a T^M_{ad} \qquad a\in\mathcal{A}, d\in\mathcal{D}, s\in\mathcal{S} \quad (A.13)$$

$$p^M_{ads} \geq \frac{P^M d^M_{ads}}{U^M_{ad} - T^M_{ad}} \qquad a\in\mathcal{A}, d\in\mathcal{D}, s\in\mathcal{S} \quad (A.14)$$

$$p^M_{ads} \geq \frac{P^M d^M_{ads}}{L^M_{ad} - T^M_{ad}} \qquad a\in\mathcal{A}, d\in\mathcal{D}, s\in\mathcal{S} \quad (A.15)$$

$$p^N_{adns} \geq \frac{P^{IN} d^N_{adns}}{U^N_{adn} - T^N_{adn}} \qquad a\in\mathcal{A}, d\in\mathcal{D}, n\in\mathcal{N}\ s\in\mathcal{S} \quad (A.16)$$

$$p^N_{adns} \geq \frac{P^{IN} d^N_{adns}}{L^N_{adn} - T^N_{adn}} \qquad a\in\mathcal{A}, d\in\mathcal{D}, n\in\mathcal{N}, s\in\mathcal{S} \quad (A.17)$$

$$p^N_{adns} \geq P^{IN} + P^{ON}(T^N_{adn} + d^N_{adns} - U^N_{adn}) \qquad a\in\mathcal{A}, d\in\mathcal{D}, n\in\mathcal{N}, s\in\mathcal{S} \quad (A.18)$$

$$p^N_{adns} \geq P^{IN} - P^{ON}(T^N_{adn} + d^N_{adns} - L^N_{adn}) \qquad a \in \mathcal{A}, d \in \mathcal{D}, n \in \mathcal{N}, s \in \mathcal{S} \quad (A.19)$$

$$v^M_{a(d-1)ds} \geq \frac{d^M_{ads}}{T^M_{ad}} - \frac{d^M_{a(d-1)s}}{T^M_{a(d-1)}} \qquad a \in \mathcal{A}, d \in \mathcal{D} \setminus \{1\}, s \in \mathcal{S} \quad (A.20)$$

$$v^M_{a(d-1)ds} \geq \frac{d^M_{a(d-1)s}}{T^M_{a(d-1)}} - \frac{d^M_{ads}}{T^M_{ad}} \qquad a \in \mathcal{A}, d \in \mathcal{D} \setminus \{1\}, s \in \mathcal{S} \quad (A.21)$$

$$v^N_{a(d-1)dns} \geq \frac{d^N_{adns}}{T^N_{adn}} - \frac{d^N_{a(d-1)ns}}{T^N_{a(d-1)n}} \qquad a \in \mathcal{A}, d \in \mathcal{D} \setminus \{1\}, n \in \mathcal{N}, s \in \mathcal{S} \quad (A.22)$$

$$v^N_{a(d-1)dns} \geq \frac{d^N_{a(d-1)ns}}{T^N_{a(d-1)n}} - \frac{d^N_{adns}}{T^N_{adn}} \qquad a \in \mathcal{A}, d \in \mathcal{D} \setminus \{1\}, n \in \mathcal{N}, s \in \mathcal{S} \quad (A.23)$$

$$v^M_{a01s} \geq \frac{d^M_{a1s}}{T^M_{a1}} - R^M_a \qquad a \in \mathcal{A}, s \in \mathcal{S} \quad (A.24)$$

$$v^M_{a01s} \geq R^M_a - \frac{d^M_{a1s}}{T^M_{a1}} \qquad a \in \mathcal{A}, s \in \mathcal{S} \quad (A.25)$$

$$v^N_{a01ns} \geq \frac{d^N_{a1ns}}{T^N_{a1n}} - R^N_{an} \qquad a \in \mathcal{A}, n \in \mathcal{N}, s \in \mathcal{S} \quad (A.26)$$

$$v^N_{a01ns} \geq R^N_{an} - \frac{d^N_{a1ns}}{T^N_{a1n}} \qquad a \in \mathcal{A}, n \in \mathcal{N}, s \in \mathcal{S} \quad (A.27)$$

$$x_{abdiks} \geq 0 \qquad a \in \mathcal{A}, d \in \mathcal{D}_i, i \in \mathcal{I}_d, b \in \mathcal{B}_i, k \in \mathcal{K}, s \in \mathcal{S} \quad (A.28)$$

$$y_{adfs} \geq 0 \qquad a \in \mathcal{A}, d \in \mathcal{D}, f \in \mathcal{F}, s \in \mathcal{S} \quad (A.29)$$

$$r_{ads},\ p^M_{ads},\ v^M_{a(d-1)ds} \geq 0 \qquad a \in \mathcal{A}, d \in \mathcal{D}, s \in \mathcal{S} \quad (A.30)$$

$$p^N_{adns},\ v^N_{a(d-1)dns} \geq 0 \qquad a \in \mathcal{A}, d \in \mathcal{D}, n \in \mathcal{N}, s \in \mathcal{S} \quad (A.31)$$

$$m_{bdik} \in \{0,1\} \qquad i \in \mathcal{I}, b \in \mathcal{B}_i, d \in \mathcal{D}_\rangle, k \in \mathcal{K} \quad (A.32)$$

$$w_{adk} \in \{0,1\} \qquad a \in \mathcal{A}, d \in \mathcal{D}, k \in \mathcal{K} \quad (A.33)$$

$$d^M_{ads} \ \ free \qquad a \in \mathcal{A}, d \in \mathcal{D}, s \in \mathcal{S} \quad (A.34)$$

$$d^N_{adns} \ \ free \qquad a \in \mathcal{A}, d \in \mathcal{D}, n \in \mathcal{N}, s \in \mathcal{S} \quad (A.35)$$

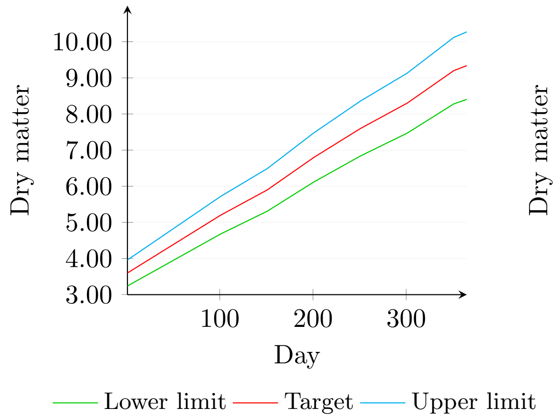
Appendix B

Feeding Requirements

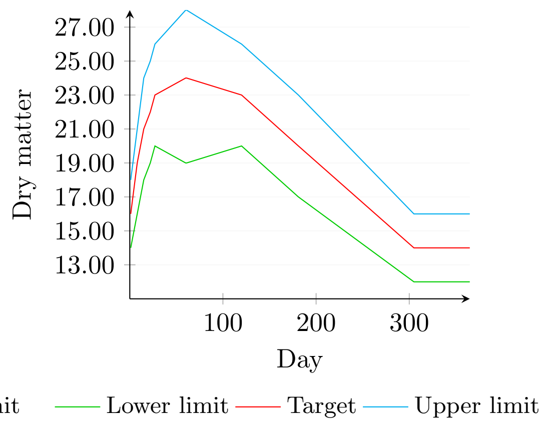
The feeding requirements for NDF is presented in Table B.1 . Figure B.1 and Figure B.2 illustrate the feeding requirements for the three animal groups in this thesis. The yellow, blue and green line in each plot represent the upper limit, target and lower limit, respectively.

Table B.1: NDF requirements for all three animal groups.

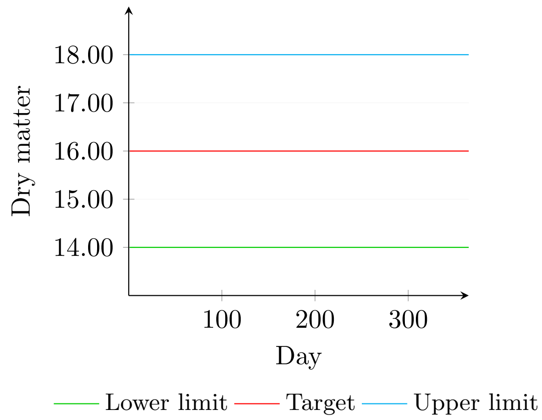
Minimum (% of dry matter)	**Target (% of dry matter)**	**Maximum (% of dry matter)**
30	40	51



(a) Growing animals

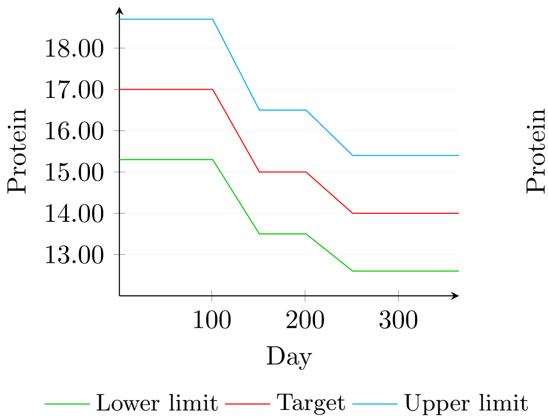


(b) Lactating animals

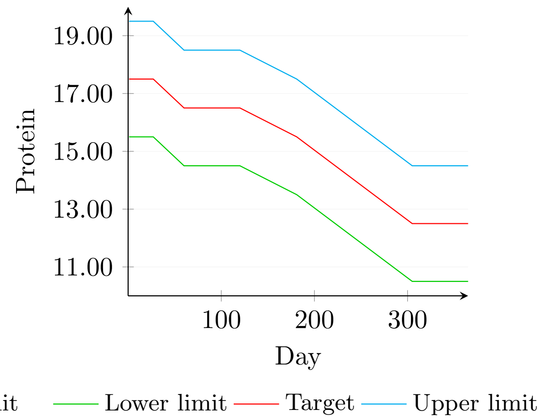


(c) Maintained animals.

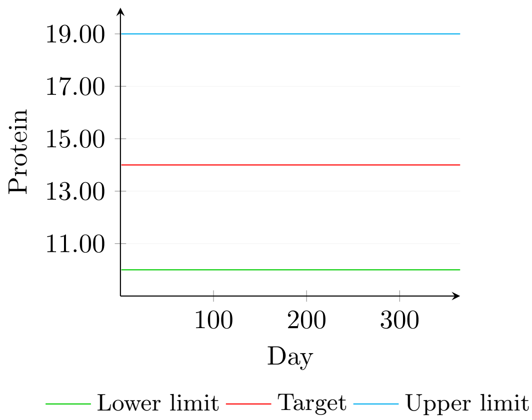
Figure B.1: Dry matter requirements for each animal group. The yellow line represents the upper bound, the blue line represents the target and the green line represents the lower bound.



(a) Growing animals



(b) Lactating animals



(c) Maintained animals.

Figure B.2: Protein requirements for each animal group. The yellow line represents the upper bound, the blue line represents the target and the green line represents the lower bound.

Appendix C

Data from TINE

Table C.1: Silage batch quality analysis

Batch ID	**Dry matter (g/kg)**	**Protein (g/kg dry matter)**	**NDF (g/kg dry matter)**
Batch 1	369	145	505
Batch 2	328	162	477
Batch 3	441	160	435
Batch 4	290	154	526
Batch 5	397	183	434
Batch 6	279	169	514

Table C.2: Feed concentrate types

Type	**Dry matter (g/kg)**	**Protein (g/kg dry matter)**	**NDF (g/kg dry matter)**
FormelProtein32FKA	884	362	186
FormelEliteNormalFKA	873.9	178	210
FormelSolidNormalFKA	871	184	192
NaturaProteinFKA	907	470	124
FormelFiberEliteFKA	878	174	271
NaturaFiberFKA	889	150	600

Appendix D

Model Performance

Table D.1: Complete results from solving the problem instances using the standard Gurobi solver.

Instance	5 days		10 days		15 days		20 days	
	time	gap	time	gap	time	gap	time	gap
xD2S	2.72s	0.52%	1576.15s	1.00%	-*	2.16%	-*	8.43%
xD10S	54.83s	0.00%	-*	3.95%	-*	16.98%	-*	32.03%
xD20S	453.75s	0.71%	-*	16.18%	-*	28.65%	-*	32.14%
xD60S	251.68s	0.00%	-*	34.21%	-*	61.73%	-*	100.00%
xD100S	722.34s	0.00%	-*	99.99%	-*	50.05%	-*	100.00%

* timed out (runtime > 10 800s)

Table D.2: Complete results from solving the problem instances using the single-cut version of the L-shaped method.

Instance	5 days		10 days		15 days		20 days	
	time	gap	time	gap	time	gap	time	gap
xD2S	195.93s	0.99%	-*	12.47%	-*	91.87%	-*	99.43%
xD10S	-*	15.73%	-*	96.21%	-*	98.02%	-*	96.71%
xD20S	-*	75.49%	-*	83.93%	-*	96.99%	-*	98.85%
xD60S	-*	68.43%	-*	96.80%	-*	98.90%	-*	98.06%
xD100S	-*	69.17%	-*	96.77%	-*	96.28%	-*	95.67%

* timed out (runtime > 10 800s)

Table D.3: Complete results from solving the problem instances using the multi-cut version of the L-shaped method.

Instance	5 days		10 days		15 days		20 days	
	time	gap	time	gap	time	gap	time	gap
xD2S	62.04s	0.96%	-*	10.02%	-*	92.52%	-*	95.91%
xD10S	162.79s	0.94%	-*	13.26%	-*	29.67%	-*	96.91%
xD20S	2225.56s	1.00%	-*	13.44%	-*	24.89%	-*	98.93%
xD60S	-*	1.41%	-*	15.92%	-*	98.77%	-*	97.76%
xD100S	-*	2.10%	-*	18.35%	-*	93.81%	-*	95.85%

* timed out (runtime > 10 800s)

Table D.4: Complete results from solving test instances with the Robust Warm start method added to the Base Case.

Instance	5 days		10 days		15 days		20 days	
	time	gap	time	gap	time	gap	time	gap
xD2S	72.77s	0.99%	-*	9.66%	-*	91.87%	-*	92.72%
xD10S	278.34s	0.94%	-*	12.58%	-*	50.92%	-*	95.86%
xD20S	2783.91s	0.96%	-*	14.68%	-*	42.21%	-*	99.01%
xD60S	-*	2.01%	-*	19.52%	-*	98.90%	-*	97.93 %
xD100S	-*	3.31%	-*	26.02%	-*	96.27%	-*	96.67%

* timed out (runtime > 10 800s)

Table D.5: Complete results from solving test instances with the Initializing Warm start method added to the Base Case.

Instance	5 days		10 days		15 days		20 days	
	time	gap	time	gap	time	gap	time	gap
xD2S	131.02s	0.00%	-*	12.29%	-*	91.87%	-*	92.72%
xD10S	572.28s	0.71%	-*	15.19%	-*	50.39%	-*	96.95%
xD20S	9563.87s	0.58%	-*	18.53%	-*	46.16%	-*	98.85%
xD60S	-*	4.81%	-*	18.43%	-*	98.90%	-*	97.94%
xD100S	-*	75.08%	-*	25.25%	-*	96.27%	-*	96.35%

* timed out (runtime > 10 800s)

Table D.6: Complete results from solving test instances with the Two-Phase approach added to the Base Case.

Instance	**5 days**		**10 days**		**15 days**		**20 days**	
	time	gap	time	gap	time	gap	time	gap
xD2S	103.14s	0.00%	-*	20.38%	-*	23.57%	-*	29.33%
xD10S	350.50s	0.56%	-*	18.17%	-*	47.09%	-*	94.91%
xD20S	7193.22s	0.76%	-*	18.71%	-*	76.50%	-*	87.45%
xD60S	-*	4.48%	-*	53.37%	-*	64.17%	-*	30.66%
xD100S	-*	5.15%	-*	63.38%	-*	31.71%	-*	67.60%

* timed out (runtime > 10 800s)

Table D.7: Complete results from solving the problem instances with the Magnanti-Wong method added to the Base Case.

Instance	**5 days**		**10 days**		**15 days**		**20 days**	
	time	gap	time	gap	time	gap	time	gap
xD2S	6.48s	0.26%	7495.755	0.24%	-*	16.52%	-*	11.72%
xD10S	20.94s	0.81%	-*	4.02%	-*	10.25%	-*	61.32%
xD20S	42.56s	0.30%	-*	4.25%	-*	15.01%	-*	81.93%
xD60S	380.60	0.85%	-*	6.64%	-*	28.88%	-*	70.54%
xD100S	443.87s	0.96%	-*	26.06%	-*	78.52%	-*	61.54%

* timed out (runtime > 10 800s)

Table D.8: Complete results from solving the problem instances with the LB-stop method added to the Base Case.

Instance	**5 days**		**10 days**		**15 days**		**20 days**	
	time	gap	time	gap	time	gap	time	gap
xD2S	7.20s	0.26%	6063.50s	0.24%	-*	4.31%	-*	11.36%
xD10S	25.44s	0.88%	3908.23s	0.82%	-*	11.84%	-*	14.67%
xD20S	72.18s	0.99%	-*	3.91%	-*	10.16%	-*	69.43%
xD60S	178.64s	0.81%	-*	7.31%	-*	39.59%	-*	60.55%
xD100S	451.80s	0.96%	-*	8.20%	-*	49.12%	-*	38.38%

* timed out (runtime > 10 800s)

Table D.9: Complete results from solving the problem instances with the ϵ-approach added to the Base Case.

Instance	**5 days**		**10 days**		**15 days**		**20 days**	
	time	gap	time	gap	time	gap	time	gap
xD2S	6.64s	0.94%	-*	1.22%	-*	5.25%	-*	17.89%
xD10S	23.70s	0.83%	-*	6.04%	-*	8.04%	-*	40.65%
xD20S	39.04s	0.96%	-*	5.12%	-*	9.32%	-*	69.33%
xD60S	142.77s	0.86%	-*	6.04%	-*	58.25%	-*	59.89%
xD100S	-*	2.21%	-*	25.48%	-*	11.21%	-*	65.68%

* timed out (runtime > 10 800s)

Table D.10: Complete results from solving the problem instances with the SolNum approach added to the Base Case.

Instance	**5 days**		**10 days**		**15 days**		**20 days**	
	time	gap	time	gap	time	gap	time	gap
xD2S	7.05s	0.26%	7301.76s	0.24%	-*	1.27%	-*	15.31%
xD10S	25.35s	0.88%	-*	4.02%	-*	1.57%	-*	10.23%
xD20S	39.58s	0.73%	-*	4.29%	-*	4.35%	-*	9.85%
xD60S	171.85s	0.72%	3151.38s	0.79%	-*	7.26%	-*	7.94%
xD100S	133.09s	0.66%	-*	3.68%	-*	5.58%	-*	9.47%

* timed out (runtime > 10 800s)

Appendix E

Simulated Value of Uncertainty

Table E.1: EEV^{SIM} and RP^{SIM} for test instance 5D30S for all the simulated scenarios.

Simulation number	$\mathbf{EEV^{SIM}}$	$\mathbf{RP^{SIM}}$	$\mathbf{VSS^{SIM}}$
1	1.243	1.053	0.190
2	1.170	0.993	0.177
3	1.335	0.906	0.430
4	0.992	0.852	0.141
5	1.637	0.961	0.676
Average	**1.276**	**0.953**	**0.323**

Table E.2: EEV^{SIM} and $RP^{SIM}$for test instance 10D30S for all the simulated scenarios.

Simulation number	$\mathbf{EEV^{SIM}}$	$\mathbf{RP^{SIM}}$	$\mathbf{VSS^{SIM}}$
1	2.209	1.854	0.355
2	2.411	1.895	0.516
3	3.001	2.223	0.778
4	2.873	2.063	0.809
5	2.234	1.894	0.340
Average	**2.546**	**1.986**	**0.560**

Table E.3: EEV^{SIM} and RP^{SIM} for test instance 15D30S for all the simulated scenarios.

Simulation number	$\mathbf{EEV^{SIM}}$	$\mathbf{RP^{SIM}}$	$\mathbf{VSS^{SIM}}$
1	4.105	3.085	1.020
2	3.616	2.919	0.697
3	3.780	2.970	0.811
4	3.428	3.1433	0.285
5	3.520	3.131	0.389
Average	**3.690**	**3.049**	**0.640**

NTNU
Norwegian University of
Science and Technology