International Journal of Production Research

ISSN: 0020-7543 (Print) 1366-588X (Online) Journal homepage: https://www.tandfonline.com/loi/tprs20

Optimisation of the broiler production supply chain

Elisabeth Brevik, Anna Ø. Lauen, Maria C. B. Rolke, Kjetil Fagerholt & Jone R. Hansen

To cite this article: Elisabeth Brevik, Anna Ø. Lauen, Maria C. B. Rolke, Kjetil Fagerholt & Jone R. Hansen (2020): Optimisation of the broiler production supply chain, International Journal of Production Research, DOI: 10.1080/00207543.2020.1713415

To link to this article: https://doi.org/10.1080/00207543.2020.1713415

Published online: 24 Jan 2020.

International Journal of Production Research, 2020
https://doi.org/10.1080/00207543.2020.1713415

Optimisation of the broiler production supply chain

Elisabeth Brevik[a], Anna Ø. Lauen[a], Maria C. B. Rolke[a], Kjetil Fagerholt[a*] and Jone R. Hansen[a,b]

[a]*Department of Industrial Economics and Technology Management, Norwegian University of Science and Technology, Trondheim, Norway;* [b]*Norsk Kylling AS, Støren, Norway*

(*Received 9 August 2019; accepted 29 December 2019*)

In this paper, we propose a mixed integer programming (MIP) model for the Chicken Flock Sizing, Allocation and Scheduling Problem (CFSASP), which is an important planning problem in the broiler production supply chain. To solve the CFSASP efficiently, two variants of rolling horizon heuristics (RHHs) have been developed and applied on the case of a Norwegian broiler production company. Computational results show that the RHHs successfully obtain high-quality solutions within a reasonable time. The value of optimisation is verified through comparison with the case company's plans, where the solutions from optimisation outperforms the current solutions. Sensitivity analyses are also conducted to provide managerial insights regarding certain strategic decisions, such as how many and which days to use for hatching of chickens. Due to the promising results, the case company is now implementing an optimisation-based decision support system based on the MIP model and solution methods shown in this paper.

Keywords: broiler production; supply chain optimisation; mixed integer programming; rolling horizon heuristic; case study

1. Introduction

Chicken and poultry, in general, is an important source of nutrition. Chickens that are reared for meat consumption are called *broilers*, and the global broiler industry is growing year by year. In 2018, poultry exceeded pork as the most consumed type of meat (Statista 2019). Broiler production is time-efficient and adaptable to market changes, due to the relatively short life cycles of broilers (Yakovleva and Flynn 2004). Moreover, the feed conversion ratio (FCR) is relatively low, with a required input of 2.3 kg feed per kg output of chicken meat. In comparison, pigs and beef calves require a lot more resources for breeding, with FCRs of 4.0 and 4.6, respectively (Wilkinson 2011). To maintain an efficient and sustainable livestock production in the decades to come, Hume, Whitelaw, and Archibald (2011) emphasise the importance of low FCRs to reduce the ecological footprints.

Profit margins for individual farmers are often small. To achieve economies of scale, the livestock industry has over the past decades developed towards large companies governing many farms (Hume, Whitelaw, and Archibald 2011). According to Asche, Cojocaru, and Roth (2018), the broiler industry 'has evolved from fragmented, locally owned businesses into one of the most efficient, vertically integrated parts of agriculture production'. Vertical supply chain integration implies coordination and collaboration between several actors. This necessitates more sophisticated planning procedures that encompass multiple aspects simultaneously. Rodríguez-Sánchez, Plà-Aragonés, and Albornoz (2012) argue that the increased complexity from supply chain integration raises the need for optimisation models to enhance performance efficiency and market competitiveness. Plà, Sandars, and Higgins (2014) support this view, and claim that the agri-food industry can utilise optimisation to improve scheduling and logistics, and the harvesting of animals for slaughter.

This paper analyses supply chain optimisation through the industrial case of a Norwegian broiler production company, Norsk Kylling, governing approximately 150 broiler farms. The company vertically integrates the supply chain processes from reception of fertilised eggs at the hatchery to transporting full-grown chickens to the slaughterhouse. Allocating newly hatched chickens to various broiler farms, and harvesting chickens from all farms when they are ready for slaughter is a highly complicated planning problem. We denote this problem as the Chicken Flock Sizing, Allocation and Scheduling Problem, hereby abbreviated CFSASP. Little to no research has previously been conducted on this topic, and to the best of our knowledge, no other optimisation models have included the continuity of time caused by never-ending production cycles. This is captured in the mixed-integer programming (MIP) model that we propose for the CFSASP.

*Corresponding author. Email: kjetil.fagerholt@ntnu.no

Due to high problem complexity and large size of the real problem, the full-scale instance of the CFSASP cannot be solved to optimality with a commercial MIP solver. Therefore, two different rolling horizon heuristics (RHHs) have been developed. The idea of an RHH is to iteratively solve the problem by dividing the time horizon into sub-periods. LP-relaxation of some variables is applied to further reduce problem complexity and computation time. As a result, high-quality solutions can be found within fractions of the time needed when commercial optimisation software is applied directly on the MIP model.

The main contribution of this paper is therefore a novel sophisticated mathematical model for the integrated problem of planning eggs for incubation, allocation of chicken flocks to farms and collection of chickens for slaughter, i.e. for the CFSASP. Furthermore, we test and apply the RHHs on a real case and show through a comprehensive computational study how it can provide significant value in both operational and strategic planning. We show for example that by using the proposed optimisation framework, the case company can improve their solutions by 15–24%. Due to the very promising results, the case company is now running a project to implement a decision support system based on these results.

The remainder of this paper is organised as follows: Section 2 presents findings from our literature study of broiler production and similar industries. The CFSASP is described in detail in Section 3, before the MIP model is presented in Section 4. Section 5 describes the proposed RHHs. Computational results are presented and discussed in Section 6, followed by concluding remarks in Section 7.

2. Literature review

The CFSASP can be classified as a combination of the two problem types 'lot-sizing and scheduling' (LSS) and the 'livestock collection problem' (LCP). Following LSS principles, the CFSASP integrates the decisions of how many chickens to cluster in each flock, and scheduling deliveries of flocks to farms. The CFSASP also includes decisions regarding slaughtering age and transport schedules ensuring a steady inflow to the slaughterhouse. These problem characteristics are typical for the LCP, which optimises transport schedules for animals that are harvested for slaughter.

Ramezaniana, Saidi-Mehrabada, and Fattahi (2013) define lot-sizing as 'determining the production quantity of each product over a finite multi-period planning horizon'. When combining lot-sizing with scheduling, i.e. sequencing of products, this should be done through simultaneous rather than successive decisions (Maravelias and Sung 2009). Copil et al. (2017) discuss more than 160 examples of LSS implementations within a wide range of industries. They emphasise the importance of integrating the interdependent decisions of lot-sizing and scheduling to obtain efficient production.

LSS problems are often hard to solve. Clark, Almada-Lobo, and Almeder (2011) list several heuristics, such as rolling horizon, tabu search and genetic algorithms, that are applied to LSS problems. Boonmee and Sethanan (2016) present a multi-level LSS implementation within the poultry industry. Production of edible eggs and meat from hens is optimised through lot-sizing and allocation of chickens to pullet farms and subsequently hen farms. To be able to solve the problem, a population-based local search metaheuristic is developed. Computations revealed that an integrated approach combining lot-sizing and scheduling reduced overall production costs.

The LCP can be classified as a combination of a vehicle routing problem (VRP) and a production planning problem. This problem was studied by Gribkovskaia et al. (2006), where the long-term goal was to develop a decision support system for cost-efficient and animal-friendly collection of livestock for slaughter. In addition to the original VRP constraints, the LCP also tracks inventory levels and requires a steady inflow of animals to the slaughterhouse. Regulations regarding animal welfare, such as limited durations for transport and storage of live animals, are also included. Oppen and Løkketangen (2008) present a tabu search method to solve the LCP for collection of cattle and pigs for slaughter. Heuristics and simulation frameworks have also been applied to similar optimisation problems for the collection of broilers. Hart, Ross, and Nelson (1999) explain how genetic algorithms can produce daily collection schedules within minutes. Oliveira and Lindau (2012) present simulated schedules for broiler collection, with the objective of avoiding stoppages at the slaughterhouse's production line.

As recognised by Oppen and Løkketangen (2008), 'It would be beneficial to integrate larger parts of the value chain in the same planning system to avoid suboptimization'. The LCP focuses on transportation and not production of livestock. Little research has previously been conducted on combining LCP and LSS, especially not within livestock production. The CFSASP combines LSS and LCP aspects to ensure a holistic optimisation of broiler production, from fertilised eggs to full-grown chickens. While LSS integrates two types of problems, the CFSASP can be viewed as a 'triple integration model', where lot-sizing, scheduling and collection are considered simultaneously.

We have found only two studies that consider optimisation of the broiler production supply chain. Taube-Netto (1996) studies the production chain of the largest poultry producer in Brazil and describes the implementation of a decision support system that seeks to optimise decisions throughout the production stages. Since no mathematical models are provided, it is hard to make precise comparisons with our study. The other study we have found that combines the LSS and LSP in broiler production is You and Hsieh (2018)'s 'production and harvesting problem'. Both this optimisation problem and the CFSASP integrate decisions for how many broilers to raise at which farms, and when to collect the broilers for slaughter. Varying barn capacities, restricted slaughtering age intervals and required cleaning days are other common aspects. Nevertheless, there are differences between the two optimisation models. You and Hsieh (2018) apply what they call 'batch-by-batch raising' which allows splitting of flocks into multiple batches through selective harvesting. The CFSASP follows the 'all-in, all-out' principle given by the Norwegian law for poultry production, where all chickens must remain clustered in the same flock until slaughtering day. Furthermore, we take end-of-period effects into account. The model presented by You and Hsieh (2018) establish a 'T-week production plan' for each broiler producer, and does not consider situations beyond the time horizon. The CFSASP, on the other hand, incorporates the continuity of time by including decisions that are made before and after the planning period.

We have also looked into optimisation studies in other related industries. Perez, de Castro, Font i Furnols (2009) and Rodríguez, Plà, and Faulin (2014) examine the pork industry, and argue that vertical coordination and integration of these complex supply chains can reduce uncertainty, increase productivity and meat quality, and enable more innovation. The pork supply chain stages resemble those of broiler production, and vertical integration can improve efficiency for production of pigs, chickens and related species (Yakovleva and Flynn 2004; Rodríguez-Sánchez, Plà-Aragonés, and Albornoz 2012; Rodríguez, Plà, and Faulin 2014). Asche, Cojocaru, and Roth (2018) argue that the supply chains of salmon farming and broiler production are highly comparable, with many similar production steps such as breeding, hatching and raising for slaughter. Abedi and Zhu (2016) consider production costs together with fish growth and customer demand, to find the optimal quantity of eggs to purchase, and the most profitable slaughtering schedules. Forsberg (1996) applies probability-based estimations of fish growth rates, and uses this to maximise profits through optimal flock sizes and slaughtering ages. Finding optimal harvesting schedules for slaughter is also discussed by Asche and Bjørndal (2011, 163–200).

To summarise the literature review, the CFSASP combines problem characteristics from both LSS and LCP in an integrated approach that has rarely been implemented in previous studies. Furthermore, we implement and test our model and methods on a real case. This is in line with the recommendations from Clark, Almada-Lobo, and Almeder (2011) and Hart, Ross, and Nelson (1999), which stress the importance of solving real-life cases rather than to aim for exact solutions based on artificial ones. The next two sections thoroughly explain the CFSASP and the MIP model.

3. Problem description

This section describes the CFSASP and its direct application to the industrial case of a broiler production company in Norway. An overview of the supply chain is given in Figure 1. Briefly, the process starts at the broiler breeders, where parent hens produce fertilised eggs which are to become broilers. These eggs are incubated and hatched at the hatchery for three weeks, before day-old-chickens (DOCs) are transported in fixed flocks to the various broiler farms. After 6–7 weeks, full-grown chickens are collected by catching teams and transported to the slaughterhouse.

The CFSASP integrates three key decisions for the broiler production company. The first is how many eggs, and from which broiler breeder, to start incubating at the hatchery on a certain day. Secondly, the company must determine chicken flock sizes, and to which broiler farm each flock should be sent. Lastly, the slaughtering age must be set for each flock, and the collection of full-grown chickens must be organised in line with given regulations for catching teams. Following the 'all in, all out' principle, all chickens must be kept in fixed flocks.

Broiler breeders also keep parent hens in fixed flocks, and must spend some days cleaning their barns before receiving a new flock. The parent hens delivering eggs to the hatchery are of different ages, and supply from each broiler breeder varies with time. The age of the parent hens affects the eggs' hatching percentage, and the DOCs' weight. If the size of chickens

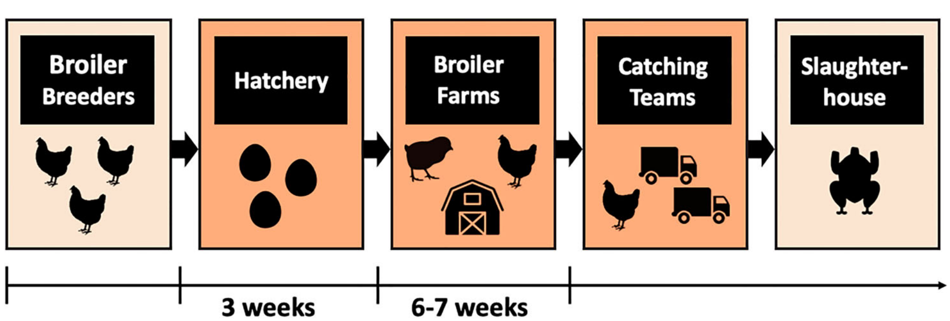

Figure 1. Broiler production supply chain, with duration of hatching and raising processes.

within a flock varies a lot, the smaller chickens often do not survive. To minimise mortality rates at broiler farms, eggs from so-called *compatible* broiler breeders, with hens of about the same age, should be flocked together.

Delivery of eggs to the hatchery is given by a predetermined supply schedule from all broiler breeders. Eggs can be stored for a limited number of days, otherwise the eggs must be discarded. Furthermore, eggs can only be put into incubation on specific incubation days, and the total number of eggs in the incubators must be within the hatchery's capacity at all times. The size of the crates used in the hatching machine, and later for transport out to the broiler farms, set a minimum batch size for the number of DOCs sent from a specific broiler breeder to a specific broiler farm.

When sizing chicken flocks, future demand as well as barn capacities and mortality rates for the different farms must be considered. The allowed flock size for each broiler farm has a lower and upper limit, given as a percentage of the barn capacity. The broiler production company wishes to fill the barn capacity, at least to a certain degree, every time they send DOCs to a farm. This is also advantageous for the farmers, since they receive wages based on the number of chickens delivered to the slaughterhouse.

Slaughtering can only take place on weekdays, and the slaughterhouse is closed on public holidays. On slaughtering day, the chickens' age must be within a given interval, and their weight should be as close to the *target slaughter weight* as possible. The farm-specific predicted chicken growth rate should be taken into account when determining the slaughtering age for each flock. Before a farm can receive a new chicken flock, a minimum number of cleaning days is required.

Some farms have two barns. Since both reception and collection of chickens is demanding, these pairs of barns should not be visited on the same day. This means that neither allocation of DOCs nor collection for slaughter can happen on the same day at both barns.

Catching teams collect chickens for slaughter, and several scheduling rules apply for the transport from broiler farms to the slaughterhouse. Each farm is assigned to one catching team, whereas one catching team can be assigned to several farms. Based on their distance to the slaughterhouse, all broiler farms are classified as either green, yellow or red. Green farms are closest to the slaughterhouse, while red farms are furthest away. The daily transport schedules should combine farms from different distance zones in a way that ensures a steady inflow to the slaughterhouse throughout the day. Therefore, only a limited number of red or yellow farms can be visited each slaughtering day. There are also upper limits for the number of visits both in total and for each catching team.

The broiler production company has a two-year agreement with each broiler farm regarding the minimum number of chickens they should receive over this period. The company must pay a downtime compensation fee to farmers that receive fewer chickens than the agreed-upon minimum number.

The objective of the CFSASP is to minimise total costs and target deviations. Actual costs include downtime compensation and costs of discarding eggs. The alternative cost for unhatched eggs is also included here. Penalty costs are added for deviations between actual and target slaughter weight, and deliveries versus desired demand at the slaughterhouse. Since we here combine actual costs and penalty costs for these deviations, the problem can be considered as a bi- (or even multi-) objective optimisation problem, where there might be a trade-off between these different objectives. However, by setting appropriate weights for these penalty costs, we assume in the following that it can be solved as a single-objective optimisation problem.

To model the problem, some assumptions have been made in accordance with how the case company operates:

- The hatching percentage only depends on the age of the parent hen, with no variations between the broiler breeders. Furthermore, the allowed storing period at the hatchery is too short to cause major changes in hatching percentage.
- Eggs with higher hatching percentage should be prioritised at incubation, to reduce discarding of eggs that are not successfully hatched. Therefore, the opportunity cost of unhatched eggs is set slightly higher than the discarding cost.
- Storing costs are considered insignificant.
- There are no specific restrictions for the transportation of DOCs from the hatchery out to broiler farms.
- Compliance to animal welfare regulations is assumed to be included in the transportation of both DOCs to broiler farms and full-grown chickens to the slaughterhouse.
- Transport costs are omitted, since they are not affected much by changes in decisions.
- Uncertainty is not a major issue in any stages of the problem.

3.1. Simplified example problem

To help the reader understand the CFSASP, we introduce in this subsection a small and simplified example with a solution. Figure 2 illustrates the supply chain flow, from eggs at broiler breeders to full-grown chickens at the slaughterhouse. The

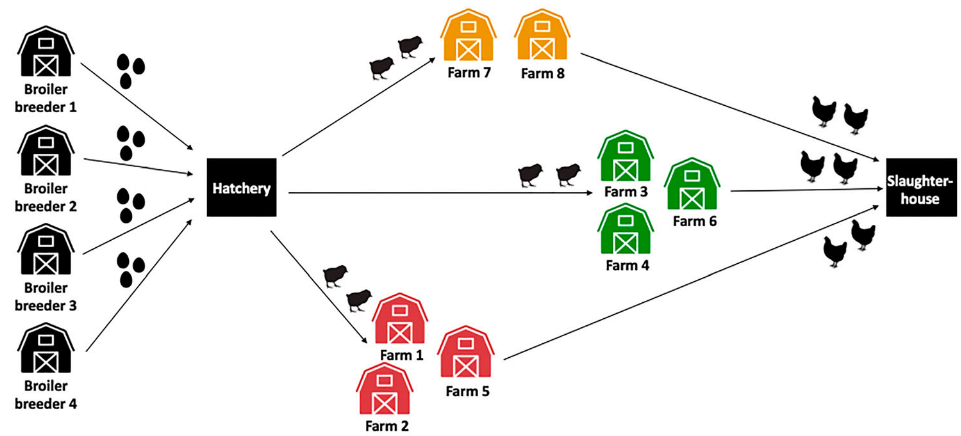

Figure 2. Overview of example problem.

problem instance is reduced to include only four broiler breeders, eight farms and three catching teams. The farms are divided into green, yellow and red distance zones, as illustrated in the figure.

On day 1, the hatchery receives 10,000 eggs from each of the broiler breeders 1, 2 and 3, and 20,000 from broiler breeder 4. Incubation of all eggs starts immediately after arrival at the hatchery. Thus, no eggs are stored or discarded. On day 22, the eggs hatch, and DOCs are sent out to broiler farms. For simplicity, the hatching percentage is set to 90% regardless of the age of parent hens. Thus, 45,000 DOCs are hatched from the 50,000 eggs received on day 1. Table 1 quantifies the example problem's flow of eggs to the hatchery, DOCs to broiler farms and full-grown chickens to the slaughterhouse.

It is desirable to flock DOCs from hens with a maximum age difference of eight weeks. Farm 1 receives 9000 DOCs from broiler breeder 1 and 4500 from broiler breeder 2, while farm 3 receives 4500 DOCs from broiler breeder 2 and 9000 from broiler breeder 3. Broiler breeder 2 is compatible with broiler breeders 1 and 3, with hens' age differences of six and five weeks, respectively. Farms 4 and 7 receive DOCs from broiler breeder 4's hens only. In theory, these farms could have received DOCs from broiler breeder 3 as well, but the batches sent from broiler breeder 4 are large enough to fulfil the capacity requirements alone.

Each of the farms 1 and 3 receive 13,500 chickens, while farms 4 and 7 receive 9000 chickens each. The size of each flock is assumed to be within the lower and upper limits of 90% and 100% of the receiving farm's barn capacity. With a mortality rate of 3% at all farms, 13,095 chickens are collected from farms 1 and 3, and 8730 from farms 4 and 7.

Chickens can stay at the farms between 45 and 48 days before the catching teams collect them for slaughter. In this example, maximum two farms can be visited by catching teams each day, and at most one of these visits can be in the red or yellow zone. Furthermore, each catching team cannot visit more than one farm each day.

45 days after hatching, on day 67, chickens from farms 1 and 4 are collected for slaughter. On day 68, chickens from farms 3 and 7 are collected. This schedule complies to all rules for catching teams. Each day there is only one visit within red or yellow zone. Two farm visits each day are also within the restrictions, and each catching team visits maximum one farm each day. Lastly, the catching teams only visit farms that they are allocated to.

The collections for slaughter gives a daily delivery of 21,825 chickens to the slaughterhouse on days 67 and 68. If for example the daily demand is 22,000, this results in a slight under-delivery of chickens on days 67 and 68.

The timeline presented in Figure 3 summarises the example problem.

Table 1. Quantification of eggs, DOCs and full-grown chickens in the example problem.

Supply of eggs from broiler breeders (BB)			Allocation of DOCs to broiler farms			
BB no.	Age of hens	Supply	Farm 1	Farm 3	Farm 4	Farm 7
1	31 weeks	10,000	9000			
2	37 weeks	10,000	4500	4500		
3	42 weeks	10,000		9000		
4	49 weeks	20,000			9000	9000
DOCs received at broiler farms			13,500	13,500	9000	9000
Chickens collected for slaughter			13,095	13,095	8730	8730

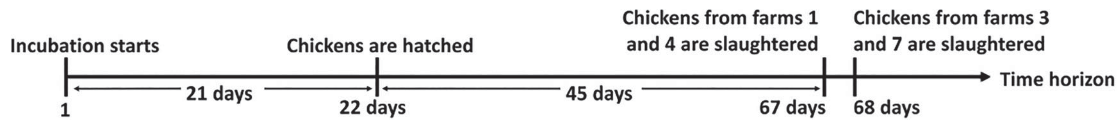

Figure 3. Timeline for the example problem's solution.

4. Mathematical model

The CFSASP is modelled as a static and deterministic discrete-time MIP problem, where each time increment equals one day. This implies that all decisions are made simultaneously, and uncertainty of parameters is disregarded. However, by re-optimising the problem regularly as new information becomes available, uncertainty can be taken into account indirectly.

4.1. Assumptions

The following assumptions aim towards reducing complexity while maintaining quality and validity of the solution.

- Flow variables for eggs and chickens are continuous. Since their values are large, the effect of LP-relaxing these variables is considered negligible.
- Eggs can only be discarded after they have been stored for the maximum number of storing days; they cannot be discarded earlier.
- Incubation and hatching is standardised, with a fixed process duration. Thus, incubation must start a certain number of days before the corresponding hatching day.
- The hatching percentage is calculated based on the age of the parent hen on the day the egg is put into incubation, not the day the egg was received. This simplification may cause minor deviations from the actual hatching percentage if the egg has been stored for many days, but these deviations are considered insignificant.
- Both the goals of meeting demand and collecting uniform chickens approaching the target slaughter weight are modelled as soft constraints, where deviations for both the constraints regarding meeting demand and the target slaughter weight are penalised in the objective function.

4.2. Notation

The model's notation, with sets, parameters and variables, is defined in Tables 2–4.

Table 2. Sets.

Set	Definition	
$\mathcal{T}$	Days	
$\mathcal{T}^{\mathrm{P}}$	Planning days,	$\mathcal{T}^{\mathrm{P}} \subseteq \mathcal{T}$
$\mathcal{T}^{\mathrm{G}}$	Storing days,	$\mathcal{T}^{\mathrm{G}} \subseteq \mathcal{T}$
$\mathcal{T}^{\mathrm{I}}$	Incubation days,	$\mathcal{T}^{\mathrm{I}} \subseteq \mathcal{T}$
$\mathcal{T}^{\mathrm{H}}$	Hatching days,	$\mathcal{T}^{\mathrm{H}} \subseteq \mathcal{T}$
$\mathcal{T}^{\mathrm{S}}$	Slaughtering days	$\mathcal{T}^{\mathrm{S}} \subseteq \mathcal{T}$
$\mathcal{T}_{\mathrm{t}}^{\mathrm{A}}$	Possible slaughtering days, for chickens hatched on day t,	$\mathcal{T}_{\mathrm{t}}^{\mathrm{A}} = \{t+\underline{A},\ldots,t+\overline{A}\} \cap \{\mathcal{T}^{\mathrm{S}}\}$
$\mathcal{T}_{\mathrm{t'}}^{\mathrm{B}}$	Possible hatching days for chickens slaughtered on day t',	$\mathcal{T}_{\mathrm{t'}}^{\mathrm{B}} = \{t'-\overline{A},\ldots,t'-\underline{A}\} \cap \{\mathcal{T}^{\mathrm{H}}\}$
$\mathcal{B}$	Broiler breeders	
$\mathcal{B}_{\mathrm{tb}}^{\mathrm{I}}$	Incompatible broiler breeders (too large age difference between hens) for broiler breeder b on day t,	$\mathcal{B}_{\mathrm{tb}}^{\mathrm{I}} \subseteq \mathcal{B}$
$\mathcal{F}$	Farms	
$\mathcal{F}^{\mathrm{Y}}$	Farms in yellow zone,	$\mathcal{F}^{\mathrm{Y}} \subseteq \mathcal{F}$
$\mathcal{F}^{\mathrm{R}}$	Farms in red zone,	$\mathcal{F}^{\mathrm{R}} \subseteq \mathcal{F}$
$\mathcal{F}_{\mathrm{c}}^{\mathrm{C}}$	Farms that are assigned to catching team c,	$\mathcal{F}_{\mathrm{c}}^{\mathrm{C}} \subseteq \mathcal{F}$
$\mathcal{F}^{\mathrm{2B}}$	Barns (f_1,f_2) that belong to the same farm,	$\mathcal{F}^{\mathrm{2B}} \subseteq \mathcal{F} \times \mathcal{F}$
$\mathcal{C}$	Catching teams	

Table 3. Parameters.

Parameter	Definition
C^{DE}	Cost of discarding one egg before incubation
C^{UE}	Opportunity cost for each unhatched egg
C^{C}	Downtime compensation cost, per under-delivered chicken
C^{NU}	Penalty cost for each non-uniform chicken, per kg deviation from target weight
C^{OD}	Penalty cost per chicken for over-delivered demand
C^{UD}	Penalty cost per chicken for under-delivered demand
I_b	Initial inventory of eggs from broiler breeder b
S_{tb}	Supply of eggs from broiler breeder b on day t
G	Maximum number of storing days for eggs at the hatchery
O	Number of days from start of incubation until hatching
R_{tb}	Hatching percentage for eggs from broiler breeder b set to incubation on day t
K^{H}	Total incubation capacity at the hatchery, given in number of eggs
K^{B}_f	Barn capacity at farm f, in kg chickens
B^{Min}	Minimum batch size for eggs from one broiler breeder to one farm
P^{Min}	Minimum percentage of barn capacity that must be filled when having a flock
L_f	Mortality rate of chickens at broiler farm f
N^{Min}_f	Minimum number of chickens farm f should receive over a two-year period
N^{LY}_f	Number of chickens farm f received last year
U^{C}_c	Maximum number of farm visits allowed per day for catching team c
U^{RY}	Maximum number of farm visits per day within red and yellow distance zone
U^{M}	Maximum number of farm visits per day in total
$\underline{A}$	Minimum slaughtering age, in days
$\overline{A}$	Maximum slaughtering age, in days
W^{T}	Target slaughter weight
$W^{\mathrm{E}}_{(t'-t)f}$	Estimated weight of chickens at age $(t'-t)$, received at farm f on day t
V	Minimum number of days required for cleaning and disinfection of barns
Q_t	Desired inflow to the slaughterhouse on day t, given in number of chickens

Table 4. Variables.

Variable	Definition
w_{tfb}	1 if broiler farm f receives chickens from broiler breeder b on day t, 0 otherwise
x_{tf}	1 if broiler farm f receives chickens on day t, 0 otherwise
y_{tf}	1 if chickens are collected for slaughter from broiler farm f on day t, 0 otherwise
$z_{tt'f}$	1 if broiler farm f receives chickens on day t, and these are collected for slaughter on day t', 0 otherwise
i_{tb}	Inventory of eggs from broiler breeder b at the hatchery at the beginning of day t
e_{tb}	Number of eggs from broiler breeder b put into incubation on day t
g_{tb}	Number of eggs from broiler breeder b discarded at the hatchery on day t
d_{tfb}	Number of chickens broiler farm f receives from broiler breeder b on day t
$j_{tt'f}$	Number of chickens sent to slaughter from broiler farm f on day t', that were hatched on day t
k_f	Number of unreceived chickens broiler farm f must be compensated for
s^+_t, s^-_t	Positive and negative deviation from demand on day t, in number of chickens

Figure 4 gives an overview by connecting variables and parameters to the flow of eggs, Day-Old-Chickens (DOCs) and full-grown chickens.

4.3. Handling end effects

An important aspect of the CFSASP is the continuity of time caused by never-ending production cycles. To cope with this issue, the time horizon is split into several subdivisions, as illustrated in Figure 5. End-of-period effects are handled by extending the total time period. At the end of the current planning period follows an 'after-period' with no more supply, where the remaining eggs and chickens are hatched and slaughtered. To capture relevant decisions made in the previous planning period, *minus-time* sets are created. To correctly model the hatching of eggs incubated before the planning period, incubation minus-days must go O days back into the previous planning period. Analogously, to model the slaughtering of

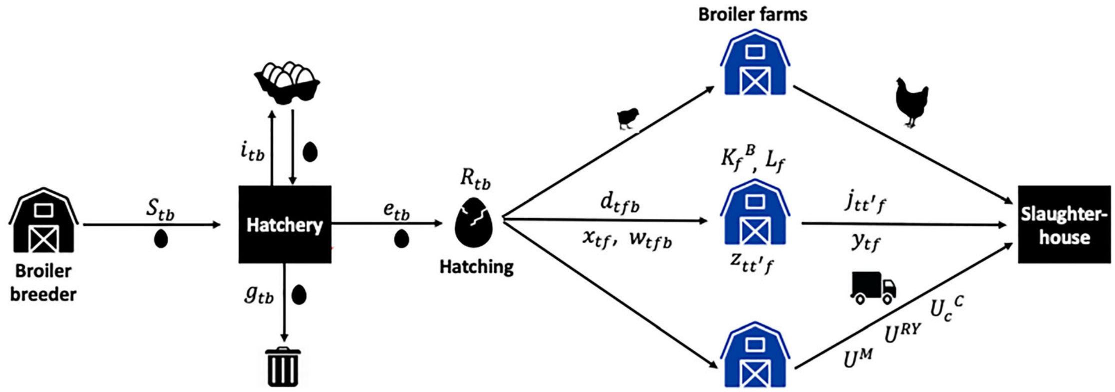

Figure 4. Variables and parameters for the flow of eggs, DOCs and full-grown chickens.

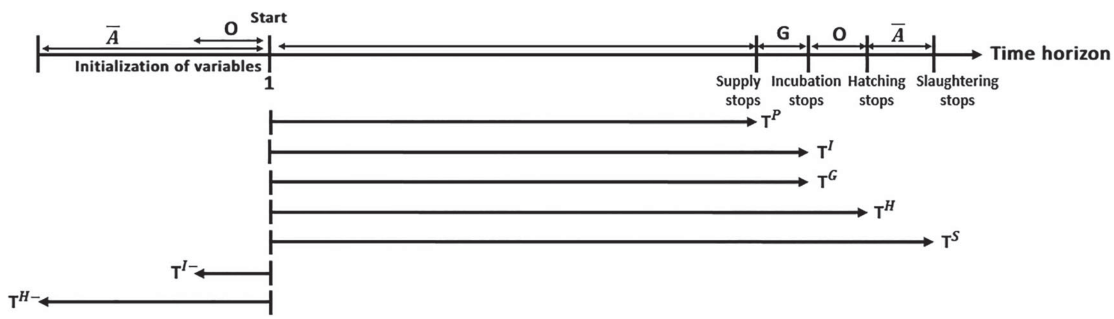

Figure 5. Different time sets handling end effects.

chickens hatched before the current planning period, hatching minus-days must go $\bar{A}$ days back. Relevant slaughtering decisions made in minus-time are also captured by appropriate modifications of time sets. Transition constraints are established by fixing initialisation variables based on decisions made in the previous planning period. We assume that the initialisation parameters are feasible and comply with all constraints.

4.4. Model formulation

In the remainder of this section, we present the overall objective of minimising costs and deviations, followed by constraints for every supply chain stage. For readability, adjustments to avoid out-of-range calculations are omitted from the model formulation.

4.4.1. Objective function

The overall objective is to minimise the sum of the actual costs and the penalty costs for not fulfilling demand and for not reaching the target slaughter weight. Part (1a) of the objective function summarises the cost of discarding eggs, the alternative cost of unhatched eggs and the compensation cost for insufficient delivery of DOCs to farms. Part (1b) penalises deviations from target slaughter weight, and over- and under-deliveries to the slaughterhouse.

$$\min \quad C^{DE} \sum_{t \in \mathcal{T}^G} \sum_{b \in \mathcal{B}} g_{tb} + C^{UE} \sum_{t \in \mathcal{T}^{\mathrm{I}}} \sum_{b \in \mathcal{B}} (1 - R_{tb}) e_{tb} + C^{C} \sum_{f \in \mathcal{F}} k_f \tag{1a}$$

$$+ C^{NU} \sum_{t \in \mathcal{T}^{\mathrm{H}}} \sum_{t' \in \mathcal{T}_t^{A} \cap \mathcal{T}^{\mathrm{P}}} \sum_{f \in \mathcal{F}} |W^E_{(t'-t)f} - W^T| j_{tt'f} + \sum_{t \in \mathcal{T}^{\mathrm{S}} \cap \mathcal{T}^{\mathrm{P}}} (C^{OD} s_t^{+} + C^{UD} s_t^{-}). \tag{1b}$$

Note that all costs are summarised over the planning period, except the discarding cost and the cost of unhatched eggs. For these two cost components, we include the extended periods of storing and incubation days. This way, we avoid that eggs are kept in storage just to be discarded right after the planning period, and ensure that eggs with higher hatching percentage still are prioritised.

4.4.2. The hatchery

All variables for inventory, discarding, incubation and hatching keep track of which broiler breeder the eggs stem from. Constraints for the processes at the hatchery are provided in the following.

$$I_b + S_{1b} - e_{1b} - g_{1b} = i_{2b} \quad b \in \mathcal{B} \tag{2}$$

$$i_{tb} + S_{tb} - e_{tb} - g_{tb} = i_{(t+1)b} \quad t \in \mathcal{T}^G \setminus \{1\},\ b \in \mathcal{B} \tag{3}$$

$$\sum_{t'=1}^{1+G} (e_{t'b} + g_{t'b}) \geq I_b \quad b \in \mathcal{B} \tag{4}$$

$$\sum_{t'=t}^{t+G} (e_{t'b} + g_{t'b}) \geq i_{tb} \quad t \in \mathcal{T}^{\mathrm{P}} \setminus \{1\},\ b \in \mathcal{B} \tag{5}$$

$$R_{tb} e_{tb} = \sum_{f \in \mathcal{F}} d_{(t+O)fb} \quad t \in \mathcal{T}^I,\ b \in \mathcal{B} \tag{6}$$

$$\sum_{t'=t}^{t+O} \sum_{b \in \mathcal{B}} e_{t'b} \leq K^H \quad t \in \mathcal{T}^I \tag{7}$$

$$e_{tb} \geq B^{\mathrm{Min}} \sum_{f \in \mathcal{F}} w_{(t+O)fb} \quad t \in \mathcal{T}^I,\ b \in \mathcal{B} \tag{8}$$

$$e_{tb} \leq \sum_{f \in \mathcal{F}} \frac{K_f^B}{R_{tb}(1 - L_f) W_{\underline{A}f}^E} w_{(t+O)fb} \quad t \in \mathcal{T}^I,\ b \in \mathcal{B}. \tag{9}$$

Initial inventory balances are presented in constraints (2), followed by the general inventory balances for all storing days in constraints (3). Outgoing inventory equals the sum of incoming inventory and supply, minus the eggs put into incubation and the discarded eggs. The outgoing inventory on the last storing day will always be zero, since supply has stopped, and all eggs must have been either put into incubation or discarded. This last matter is modelled in constraints (4) and (5) for the inital and general case, respectively, which require that all supply of eggs must be put into incubation within the limited number of storing days, otherwise the eggs will be discarded.

Mass conservation is ensured through constraints (6), where the number of eggs put into incubation times their hatching percentage equals the output of DOCs after O days. Constraints (7) set the incubator capacity as the upper limit for the total number of eggs that can be incubated during each period of O days.

Constraints (8) and (9) link the flow variables for incubation of eggs from certain broiler breeders with the binary variables for farms receiving eggs from these broiler breeders. Constraints (8) provide a lower limit for the number of eggs that can be put into incubation, given by the minimum batch size times the number of farms that receives eggs from the specific broiler breeder O days later. The upper limit in constraints (9) is given by an adjusted sum of barn capacities at the receiving farms. Since not all eggs hatch and not all chickens survive, the barn capacity is divided by the eggs' hatching percentage and the survival rate of chickens (i.e. one minus the farms' mortality rate). The denominator also includes the lowest possible slaughtering weight, i.e. the estimated weight at the lowest allowed slaughtering age. This last division converts the unit from kg to number of chickens.

4.4.3. Broiler farms

DOCs are sent from the hatchery to various broiler farms, where they can stay for a period of $\underline{A}$ to $\overline{A}$ days. The constraints associated with this part of the supply chain are as follows:

$$w_{tfb} \leq x_{tf} \quad t \in \mathcal{T}^H,\ f \in \mathcal{F},\ b \in \mathcal{B} \tag{10}$$

$$x_{tf} \leq \sum_{b \in \mathcal{B}} w_{tfb} \quad t \in \mathcal{T}^H,\ f \in \mathcal{F} \tag{11}$$

$$d_{tfb} \geq B^{\mathrm{Min}} R_{(t-O)b} w_{tfb} \quad t \in \mathcal{T}^H,\ f \in \mathcal{F},\ b \in \mathcal{B} \tag{12}$$

$$d_{tfb} \leq \frac{K_f^B}{(1 - L_f) W_{\underline{A}f}^E} w_{tfb} \quad t \in \mathcal{T}^H,\ f \in \mathcal{F},\ b \in \mathcal{B} \tag{13}$$

$$\sum_{t' \in \mathcal{T}_t^A} j_{tt'f} = \sum_{b \in \mathcal{B}} (1 - L_f) d_{tfb} \quad t \in \mathcal{T}^H, f \in \mathcal{F} \tag{14}$$

$$W^{\mathrm{E}}_{(t'-t)f} j_{tt'f} \leq K_f^{\mathrm{B}} z_{tt'f} \quad t \in \mathcal{T}^H,\ t' \in \mathcal{T}_t^A, f \in \mathcal{F} \tag{15}$$

$$W^{\mathrm{E}}_{(t'-t)f} j_{tt'f} \geq P^{Min} K_f^{B} z_{tt'f} \quad t \in \mathcal{T}^H,\ t' \in \mathcal{T}_t^A, f \in \mathcal{F} \tag{16}$$

$$\sum_{t' \in \mathcal{T}_t^A} z_{tt'f} = x_{tf} \quad t \in \mathcal{T}^H, f \in \mathcal{F} \tag{17}$$

$$\sum_{t \in \mathcal{T}_{t'}^B} z_{tt'f} = y_{t'f} \quad t' \in \mathcal{T}^S, f \in \mathcal{F} \tag{18}$$

$$w_{tfb} + w_{tfb'} \leq 1 \quad t \in \mathcal{T}^H, f \in \mathcal{F},\ b \in \mathcal{B},\ b' \in \mathcal{B}^{\mathrm{I}}_{\mathrm{tb}} \tag{19}$$

$$x_{tf_1} + x_{tf_2} + y_{tf_1} + y_{tf_2} \leq 1 \quad t \in (\mathcal{T}^H \cup \mathcal{T}^S),\ (f_1,f_2) \in \mathcal{F}^{2B} \tag{20}$$

$$N_f^{LY} + \sum_{t \in \mathcal{T}^{\mathrm{H}} \cap \mathcal{T}^{\mathrm{P}}} \sum_{b \in \mathcal{B}} d_{tfb} + k_f \geq N_f^{Min} \quad f \in \mathcal{F}. \tag{21}$$

Constraints (10) and (11) connect the general binary variables x_{tf} for allocation of DOCs to a farm with the corresponding binary variables w_{tfb} that specify which broiler breeder the eggs stem from. Constraints (10) are binding when w_{tfb} equals 1, while constraints (11) are binding when the sum of all w_{tfb} for a given t and f equals 0.

Constraints (12) and (13) link the binary and flow variables for farms receiving DOCs from broiler breeders. The minimum number of DOCs that can be sent to a farm from a specific broiler breeder equals the minimum batch size for eggs from one broiler breeder, adjusted by the hatching percentage at the time of incubation. The maximum number equals the full barn capacity, adjusted by the mortality rate. As in constraints (9), the barn capacity is divided by the minimum allowed slaughtering age, to convert the unit from kg to number of chickens.

Mass conservation at the broiler farms is ensured by constraints (14). The number of chickens sent to the slaughterhouse equals the number of chickens sent to the farm, adjusted by the mortality rate. Chicken flock size regulations are given in constraints (15) and (16), which also connect flow variables $j_{tt'f}$ with binary variables $z_{tt'f}$. Compliance with capacity regulations is ensured by stating that the estimated slaughter weight times the number of chickens that are collected for slaughter can not exceed the broiler farm's barn capacity. The lower limit for each flock size is given as a percentage of the barn capacity.

Constraints (17) and (18) connect the binary variables for reception and collection of broilers at farms. They also determine the possible slaughtering days for each hatching, based on the allowed slaughtering age interval. If DOCs are sent to broiler farm f on day t, they must be collected on one of the corresponding possible slaughtering days. Analogously, DOCs that are collected for slaughter on day t' must have been sent to the farm on one of the corresponding possible hatching days.

DOCs in the same flock must stem from compatible broiler breeders, i.e. broiler breeders with hens of age within a given interval. This is ensured by constraints (19). Constraints (20) state that pairs of barns that belong to the same farm cannot be visited on the same day, neither by people from the hatchery allocating DOCs nor by catching teams collecting chickens for slaughter. Lastly, constraints (21) calculate the amount of undelivered chickens that case company must compensate for, based on agreements with each broiler farm. This amount is calculated by comparing the two-year minimum requirement with the sum of last year's and this year's delivery.

4.4.4. Catching teams and the slaughterhouse

The constraints regarding catching team schedules, cleaning days at broiler farms and desired inflow to the slaughterhouse are presented in the following.

$$\sum_{f \in \mathcal{F}_c^{\mathrm{C}}} y_{tf} \leq U_c^{\mathrm{C}} \quad t \in \mathcal{T}^S,\ c \in \mathcal{C}, \tag{22}$$

$$\sum_{f \in \mathcal{F}^{\mathrm{R}} \cup \mathcal{F}^{\mathrm{Y}}} y_{tf} \leq U^{\mathrm{RY}} \quad t \in \mathcal{T}^S, \tag{23}$$

$$\sum_{f \in \mathcal{F}} y_{tf} \leq U^{\mathrm{M}} \quad t \in \mathcal{T}^S, \tag{24}$$

$$x_{tf} \leq \sum_{t' \in \mathcal{T}_t^A} y_{t'f} \quad t \in \mathcal{T}^H, f \in \mathcal{F}, \tag{25}$$

$$\sum_{t'=t}^{t+\underline{A}+V} x_{t'f} \leq 1 \quad t \in \mathcal{T}^{\mathcal{H}}, f \in \mathcal{F}, \tag{26}$$

$$\sum_{t'=t}^{t+\underline{A}+V} y_{t'f} \leq 1 \quad t \in \mathcal{T}^{S}, f \in \mathcal{F}, \tag{27}$$

$$\sum_{t \in \mathcal{T}_{t'}^B} \sum_{f \in F} j_{tt'f} - s_{t'}^{+} + s_{t'}^{-} = Q_{t'} \quad t' \in \mathcal{T}^S. \tag{28}$$

The number of daily visits for each catching team is restricted in constraints (22), while constraints (23) and (24) restrict the number of visits in red and yellow zone, and the total number of visits, respectively. Constraints (25) ensure that chickens can only be collected for slaughter when their age is within the allowed slaughtering age interval. Constraints (26) and (27) regulate cleaning days between flocks by requiring that chickens can only be received or collected for slaughter at most once every $\underline{A} + V$ days. Lastly, constraints (28) calculate deviations between deliveries and demand at the slaughterhouse.

The daily transport schedules should combine farms from different distance zones in a way that ensures a steady inflow to the slaughterhouse throughout the day. Similar types of constraints also appear in other agricultural settings and there are alternative ways to cope with such constraint, see for example Junqueira and Morabito (2019) in the sugarcane harvest front scheduling problem in sugarmill supply logistics.

4.4.5. *Variable declarations*

Binary restrictions are given to the variables w_{tfb}, x_{tf}, y_{tf} and $z_{tt'f}$, while non-negativity requirements are imposed on the remaining variables.

5. Rolling horizon heuristics

The MIP model for the CFSASP presented in Section 4 is extremely hard to solve for real-sized instances. Therefore, we propose two different rolling horizon heuristics (RHHs). An RHH iteratively solves sub-problems for shorter planning horizons, and reduces the solution space by gradually fixing variables and has proven to be an efficient metaheuristic for various production planning and scheduling problems (Rakke et al. 2011; Andersson, Fagerholt, and Hobbesland 2015). Ramezaniana, Saidi-Mehrabada, and Fattahi (2013) implement different RHH approaches to solve a lot-sizing and scheduling problem. They argue that even though RHHs are usually applied for dynamic problems, where information is continuously revealed, this type of heuristic is also efficient for static planning problems with long planning horizons. This view is supported by Andersson, Fagerholt, and Hobbesland (2015), who argue that the RHH is powerful when long planning horizons make it hard to solve the problem.

Figure 6 illustrates the general RHH structure. The planning horizon is divided into three periods, denoted *fixed, central,* and *forecast.* In each iteration, the optimisation problem is solved for all three periods. The central period is optimised based on the original mathematical model, while in the forecast period, binary variables are LP-relaxed. Before proceeding to the next iteration, a given subset of decision variables from the central period are fixed, and these values are stored for future iterations. The central and forecast periods have constant lengths $|\mathcal{T}^C|$ and $|\mathcal{T}^F|$, and their starting points are shifted by a fixed increment Δt in every iteration. The length of the fixed period increases with the same increment Δt, and the iteration procedure is repeated for N iterations until the entire planning horizon, $|\mathcal{T}^{Tot}|$, is planned for.

When determining the length of the central and forecast periods, a trade-off arises between reduced computation time for shorter periods and increased solution quality for longer periods (Andersson, Fagerholt, and Hobbesland 2015). Stolletz and Zamorano (2014) emphasise the importance of proper parameter tuning for the RHH's performance.

Strategic decisions must also be made for the fixed and forecast periods. The subset of variables that should be fixed in each iteration must be determined. Mercé and Fontan (2003) present the alternatives of fixing both binary decision variables and continuous flow variables, or only the binary variables. They favour the latter approach, and this strategy is also applied by Andersson, Fagerholt, and Hobbesland (2015). As for the forecast period, Rakke et al. (2011) argue that LP-relaxing binary variables is an effective simplification strategy, since continuous variables require significantly less computational effort than binary variables. Implementing appropriate variable fixing and simplification strategies is crucial to obtain an effective RHH.

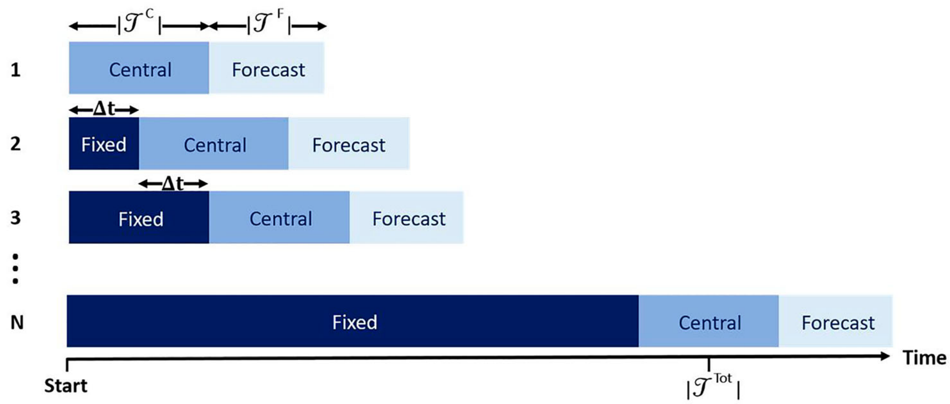

Figure 6. Iterative structure of the rolling horizon heuristic.

It can be noted that the RHH can be viewed as a special case of the well-known constructive relax-and-fix heuristics, where the variables are grouped by periods and the heuristic iteration number is the number of periods; see for example Dillenberger et al. (1994) and Baldo et al. (1994). Depending on the lengths of Δt and the central period, the RHH is a relax-and-fix heuristic without or with overlapping.

In the remainder of this section, we present two different RHH approaches for the CFSASP. First, we describe a traditional RHH similar to the ones found in the existing literature, where all variables w_{tfb}, x_{tf}, y_{tf} and $z_{tt'f}$ are kept binary in the central period and LP-relaxed in the forecast period. We have also developed an alternative RHH, where the w_{tfb} variables remain LP-relaxed in all iterations 1 through N. The binary restrictions are re-added after the last iteration has been executed. To the best of our knowledge, such a separation with two distinct stages of binary restrictions has not been implemented in previous research.

5.1. Original RHH

Following the standard RHH procedure, the planning horizon for a given iteration n is divided into three parts; $\mathcal{T}_n^{\text{Fixed}}$, $\mathcal{T}_n^{\text{Central}}$ and $\mathcal{T}_n^{\text{Forecast}}$. $\mathcal{T}_n^{\text{Fixed}}$ contains all fixed binary variables w_{tfb}, x_{tf}, y_{tf} and $z_{tt'f}$, and the length of this period increases by Δt with every iteration. Δt is defined as the length of the total planning horizon divided by the number of iterations. The central period contains variables as declared in the MIP model, while in the forecasting period, all binary variables are LP-relaxed. $\mathcal{T}_n^{\text{Central}}$ and $\mathcal{T}_n^{\text{Forecast}}$ have constant lengths, and are iteratively shifted by Δt. The maximum allowed computation time for each iteration is a parameter that must be determined during implementation.

In every iteration, the optimisation problem is solved for all three parts of the planning horizon. The fixed period is then expanded for the next iteration, and all binary variables from the best solution found within the maximum allowed iteration time are fixed and stored. The central and forecasting periods are shifted, before the next iteration starts. The final output is the best solution from the last iteration, where all binary variables except for the last Δt days have been fixed in previous iterations. The pseudo-code provided in Algorithm 1 summarises our RHH.

5.2. RHH-LIBR: last iteration binary restrictions

Even though the original RHH can reduce problem complexity, solving the CFSASP might still be very time-consuming. To further shorten computation time, we propose a new RHH, where binary restrictions for w_{tfb} are removed also in the central period. This LP-relaxation lessens the computational effort required to solve the optimisation problem. We use the label RHH-LIBR to denote this alternative RHH.

In RHH-LIBR, binary restrictions for w_{tfb} are re-added after all iterations 1 to N are executed. The optimisation problem is then solved for the entire planning period in an additional iteration '$\text{N}+1$'. Despite this extra iteration, we expect that the total computation time can be reduced without significant degradation of solution quality. We argue that the maximum allowed iteration time can be lowered, since the LP-relaxation of w_{tfb} lessens the computational effort required for Constraints (8)–(13) in each iteration from 1 to N. Thus, good solutions can hopefully be found within shorter iteration times with RHH-LIBR.

Algorithm 1 Rolling Horizon Heuristic

$n = 1$
$\mathcal{T}_1^{\text{Fixed}} = \emptyset$; $\mathcal{T}_1^{\text{Central}} = \{\ 1, \ldots, |\mathcal{T}^{\text{C}}|\ \}$; $\mathcal{T}_1^{\text{Forecast}} = \{\ |\mathcal{T}^{\text{C}}|, \ldots, |\mathcal{T}^{\text{C}}| + |\mathcal{T}^{\text{F}}|\ \}$
$\Delta t = \frac{|\mathcal{T}^{Tot}|}{N}$
while $n < N$ **do**
  Solve the problem for $\mathcal{T}_n^{\text{Fixed}} + \mathcal{T}_n^{\text{Central}} + \mathcal{T}_n^{\text{Forecast}}$ (within max. iteration time)
  Expand $\mathcal{T}_n^{\text{Fixed}}$ by Δt days to obtain $\mathcal{T}_{n+1}^{\text{Fixed}}$
  Fix binary variables w_{tfb}, x_{tf} and y_{tf}, for $t \in \mathcal{T}_{n+1}^{\text{Fixed}}$, for all f, and for all b
  Fix binary variables $z_{tt'f}$, for $t' \in \mathcal{T}_{n+1}^{\text{Fixed}}$, $t \in \mathcal{T}_{t'}^{\text{B}}$, and for all f
  Shift $\mathcal{T}_n^{\text{Central}}$ and $\mathcal{T}_n^{\text{Forecast}}$ by Δt days to obtain $\mathcal{T}_{n+1}^{\text{Central}}$ and $\mathcal{T}_{n+1}^{\text{Forecast}}$
  $n = n + 1$
end while
Solve the problem for $\mathcal{T}_N^{\text{Fixed}} + \mathcal{T}_N^{\text{Central}} + \mathcal{T}_N^{\text{Forecast}}$ (within max. iteration time)
Output: Best solution from iteration N

6. Computational study

The MIP model and the RHHs have been implemented in the modelling language Mosel (version 4.8.3) and solved with the optimisation software FICO® Xpress (IVE version 1.24.24.64 bit and optimiser version 33.01.02). The computer processor used is Intel® Core™ i7-6700 CPU @ 3.40 GHz, with 32 GB RAM, and the operating system Windows 10 education 64-bit. This section first presents the test instances, followed by results from the testing of the rolling horizon heuristics (RHHs). Economic aspects are discussed through bi-objective programming and lastly a summary of managerial insight gained from various analyses of more strategic nature is provided.

6.1. Test instances

To validate the model formulation and analyse the performance of the RHHs, test instances with fewer farms (F^{Inst}) than the case company have been generated. Some modifications have been needed to maintain essential problem characteristics and obtain comparable results for these. Based on the number of farms, supply (i.e. the number of broiler breeders, B^{Inst}) and demand (Q^*) have been downscaled accordingly. The length of the planning horizon is set to 360 days (one year), which is in accordance with the case company's planning horizon. We have test instances with 30, 60, 100 and 145 farms, where the test instance with 145 farms correspond to the real problem for the case company. Table 5 presents the key information for the test instances.

6.2. Results from the rolling horizon heuristics

Solving the CFSASP is very difficult or even impossible using a commercial MIP solver. However, an encouraging discovery is that the instances with only 90 days planning horizon give small optimality gaps also for the full-scale number of farms. This motivates the use of RHHs with iterative periods of up to 90 days.

6.2.1. Original RHH

Regarding the length of the central and forecast periods, we have experimented with different combinations of 30 and 60 days for the original RHH. Table 6 lists the resulting variants. The increment for the fixed period is kept constant at 30 days, which gives 12 iterations for the one-year planning horizon.

Table 5. Key information for the test instances.

F^{Inst}	B^{Inst}	Q^*
30	3	12,500
60	6	25,000
100	10	37,500
145	14	50,000

Table 6. Three RHH variants with different lengths of periods.

Method	Length of period [days]		
	Δt	Central	Forecast
RHH_1	30	30	30
RHH_2	30	60	30
RHH_3	30	30	60

Table 7. Computational results from solving the T360-instances using optimisation software and all RHH variants.

Instance	Method	Objective value	Time [s]	Gap [%]
F30	MIP solver	12,260	259,200	3.0
	RHH_1	13,638	84.2	14.6
	RHH_2	12,370	7418	3.9
	RHH_3	12,327	233.4	3.6
F60	MIP solver	50,590	259,200	198.7
	RHH_1	21,884	14,979	29.2
	RHH_2	17,754	52,752	4.8
	RHH_3	17,931	34,515	5.9
F100	MIP solver	92,760	259,200	333.2
	RHH_1	26,693	9145	24.7
	RHH_2	22,030	66,373	2.9
	RHH_3	22,674	69,765	5.9
F145	MIP solver	–	259,200	–
	RHH_1	30,771	14,747	20.0
	RHH_2	27,027	79,299	5.4
	RHH_3	26,880	72,946	4.8

Notes: The maximum computation time is set to three days (i.e. 259,200 seconds) for the MIP solver, and 24 hours (i.e. 86,400 seconds) for the RHHs. Objective values are given in costs of 1000 NOK.

The maximum iteration time must be determined properly. For the MIP solver, we set the maximum computation time to three days (i.e. 259,200 seconds). After various experiments and parameter tuning for the RHHs, we concluded that a maximum iteration time of two hours (i.e. 7200 seconds) is sufficient, giving a maximum total computation time of 24 hours (i.e. 86,400 seconds).

The computational results from the three RHH variants are presented in Table 7. The gaps reported are calculated as '(best solution – best bound)/best bound', where the best bound is the lower bound from the MIP solver after three days of running time.

RHH_1 uses much less time than RHH_2 and RHH_3, although at the cost of significantly larger gaps. It appears that an iteration planning period of only $30 + 30$ days is too myopic. RHH_2 and RHH_3 are relatively close to each other in performance, but it can be noted that RHH_3 is somewhat better than RHH_2 on the full-scale instance, with both a shorter computation time and smaller optimality gap.

We can also note from Table 7 that the MIP solver provides best solutions on the smallest instance F30, but RHH_2 and RHH_3 are not far behind. However, for the larger instances, the RHHs outperform the MIP solver, with smaller gaps within less than a third of the computation time. Using the MIP solver, no integer solution could be found for F145 even after three days, while RHH_3 has a relatively small gap of 4.8% in about 20 hours (72,946 seconds). These results are very promising, especially when we keep in mind that the gap is calculated relative to the lower bound from the MIP solver, which means that the gap from the optimal solution could be even smaller.

6.2.2. *RHH-LIBR*

Recall that RHH-LIBR might find good solutions even faster than the original RHH. The LP-relaxation of w_{tfb} variables in the central period reduces problem complexity, which allows for a shorter iteration time. Preliminary analysis revealed that the iteration time could be halved from two hours for the original RHH to one hour (i.e. 3600 seconds) for RHH-LIBR. The maximum computation time for all 13 iterations then becomes 13 hours (i.e. 46,800 seconds).

Table 8. Computational results from RHH-LIBR applied on the T360-instances.

Instance	Objective value	Time [s]	Gap [%]
F30	12 650	1 912	6.3
F60	18 095	22 632	6.8
F100	22 919	20 297	7.0
F145	27 743	41 746	8.1

Note: Objective values are given in costs of 1000 NOK.

Table 8 presents computational results obtained from implementing RHH-LIBR with iteration planning periods of 30 + 60 days, analogous to the RHH_3 variant from Table 6. With the fixed increment of $\Delta t = 30$ days, we get 12 iterations with LP-relaxed w_{tfb} variables, and a 13th iteration where the binary restrictions are re-added.

We observe that RHH-LIBR can solve the full-scale problem in almost half of the time required by RHH_2 and RHH_3, although with optimality gaps that are somewhat larger.

6.3. Bi-objective optimisation

So far, we have analysed the CFSASP as a Single Objective Problem (SOP), where all costs are considered simultaneously. However, in reality we have a mix of actual and penalty costs for not meeting demand and for deviations from the target slaughter weight. It might therefore be interesting to distinguish actual from the penalty costs through bi-objective optimisation, to examine whether these two types of costs are conflicting. The actual and penalty costs are defined in Equations (29) and (30) , respectively. The former type includes discarding costs and downtime compensation fees, as well as the alternative cost of unhatched eggs. The latter type includes penalties for non-uniform chickens and deviations from demand. The internal ratios of the different cost elements within the two objectives are kept constant.

$$C^{\text{Actual}} = C^{DE} \sum_{t \in \mathcal{T}^G} \sum_{b \in \mathcal{B}} g_{tb} + C^{UE} \sum_{t \in \mathcal{T}^1} \sum_{b \in \mathcal{B}} (1 - R_{tb}) e_{tb} + C^C \sum_{f \in \mathcal{F}} k_f \tag{29}$$

$$C^{\text{Penalty}} = C^{NU} \sum_{t \in \mathcal{T}^{\text{H}}} \sum_{t' \in \mathcal{T}_t^{A} \cap \mathcal{T}^{\text{P}}} \sum_{f \in \mathcal{F}} |W^E_{(t'-t)f} - W^T| j_{tt'f} + \sum_{t \in \mathcal{T}^{\text{S}} \cap \mathcal{T}^{\text{P}}} (C^{OD} s_t^+ + C^{UD} s_t^-). \tag{30}$$

To solve the bi-objective problem, we apply ‘linear combination of weights’, as presented by Jaimes, Martínez, and Coello (2009). This method transforms a multi-objective problem into a SOP by minimising the weighted sum of all objectives. The CFSASP's bi-objective problem is represented by constraints (31)–(34). We use the weights λ_1 and λ_2 for the actual costs and the penalty costs, respectively. Note that we multiply each weight by 2 in the objective function. This is just a formality, to make the combination $\lambda_1 = \lambda_2 = 0.5$ correspond to the original objective function for the CFSASP, where actual and penalty costs are weighted equally. Constraints (32) represent all constraints given by the MIP model described in Section 4. The non-negative weights should always sum up to one.

$$\min \quad z = 2\lambda_1 C^{\text{Actual}} + 2\lambda_2 C^{\text{Penalty}} \tag{31}$$

$$\text{s.t.} \quad x \in \mathcal{X} \tag{32}$$

$$\lambda_1 + \lambda_2 = 1 \tag{33}$$

$$\lambda_1, \lambda_2 \geq 0. \tag{34}$$

According to Jaimes, Martínez, and Coello (2009), all solutions to the weighting problem where all weights have non-zero values are Pareto optimal. In order to find such solutions of sufficient quality within a reasonable computation time, the RHH_3 is applied on the full-scale instance for the bi-objective problem. Figure 7 shows the resulting Pareto front from using weights $\lambda_1, \lambda_2 \in (0, 1)$. The solutions obtained when setting the weights equal to 0 and 1 are out of range, and thus excluded.

The curve through all points in Figure 7 is not perfectly convex, even though this is a requirement for Pareto fronts. This is because we use a heuristic, so that we obtain only an approximation of the Pareto front. Another important observation is that the range of costs for all solutions is relatively narrow, with approximately 6% difference for actual costs, and only 3.5% for the penalty costs. It appears that the weighting of actual versus penalty costs can be chosen rather arbitrarily without major impact on the objective value, as long as both types of costs are considered (i.e. $\lambda_1, \lambda_2 \in (0, 1)$). This indicates that the two types of costs are not contradictory, and that all combinations tend to produce similar solutions.

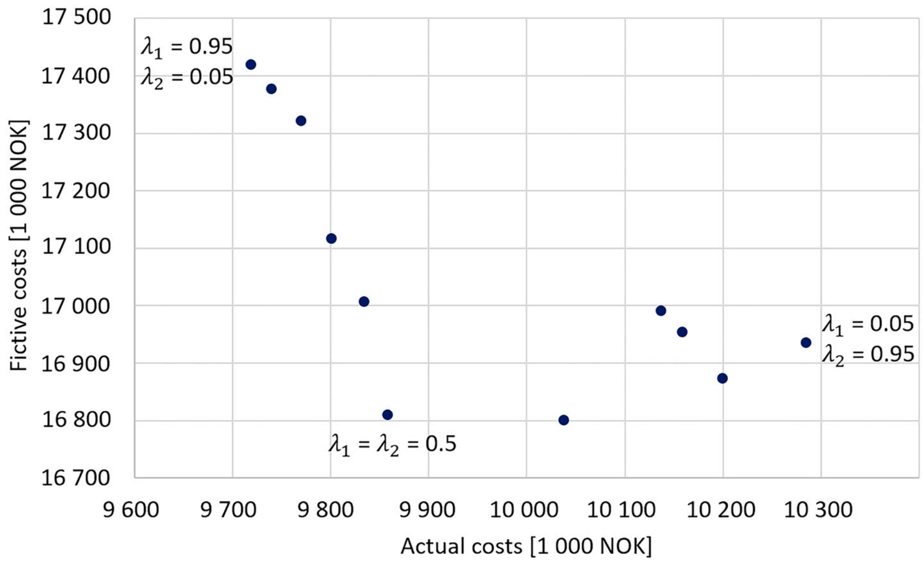

Figure 7. Approximated Pareto front from bi-objective optimisation with different weights λ_1 and λ_2.

6.4. Managerial insights

A number of economic analyses have been performed using the RHH_3 solution method. The sensitivity analyses presented in the following study the effect of altering the values of certain parameters, representing strategic choices. Afterwards, the value of optimisation-based decision support is emphasised by comparing the optimised schedules with the current production plans at the Norwegian case company producing broilers.

6.4.1. Sensitivity analyses

The first aspect we examined was the allowed age difference limit between parent hens for each chicken flock, i.e. regulations for the set $\mathcal{B}^{\mathrm{I}}_{\mathrm{tb}}$. For the case company, the goal is to keep the age difference at maximum eight weeks, otherwise the size of the chickens will differ so much that many small chickens will not survive. Higher mortality rates for larger age differences is not captured by the CFSASP, where the eight-weeks limit is modelled as a hard constraint. By varying the allowed age difference, we observed that the objective value is approximately equal from four weeks and upwards, while an allowed age difference of only two weeks significantly restricts the problem and increases discarding and under-delivery costs. This indicates that an allowed age difference of maximum four weeks could give better result in practice than the current eight-weeks limit, since a stricter maximum level lowers the mortality rate. Hence, the production efficiency can be enhanced.

Secondly, we experimented with different sets of weekly hatching days, i.e. $\mathcal{T}^{\mathrm{H}}$, as shown in Table 9.

Comparing with the current practice of hatching on Mondays and Thursdays, we observed how the objective value was affected by adding another hatching day and allowing for hatching on weekends. Table 10 shows the detailed cost distribution for the different sets of hatching days. By changing the set of hatching days to Thursdays and Sundays, the objective value was decreased by around 11%. When evaluating this option, the cost of weekend work should be considered, as this is not included in the objective function now. Adding another hatching day, Friday, reduced the objective value with

Table 9. Sets of hatching days. $\mathcal{T}^{H}_{0}$ denotes the current practice.

Set	Hatching days
$\mathcal{T}^{\mathrm{H}}_{0}$	Monday and Thursday
$\mathcal{T}^{\mathrm{H}}_{1}$	Monday and Friday
$\mathcal{T}^{\mathrm{H}}_{2}$	Thursday and Sunday
$\mathcal{T}^{\mathrm{H}}_{3}$	Friday and Saturday
$\mathcal{T}^{\mathrm{H}}_{4}$	Monday, Thursday and Friday
$\mathcal{T}^{\mathrm{H}}_{5}$	Tuesday, Thursday and Sunday

Table 10. Detailed cost distribution for the different sets of hatching days.

Set of hatching days	Total discarding	Downtime compensation	Non-uniform chickens	Over-delivery	Under-delivery
$\mathcal{T}_0^{\mathrm{H}}$	8791	1026	14,813	367	1884
$\mathcal{T}_1^{\mathrm{H}}$	8740	1101	14,986	423	1981
$\mathcal{T}_2^{\mathrm{H}}$	8823	1044	11,318	523	2326
$\mathcal{T}_3^{\mathrm{H}}$	8818	1026	14,588	424	2090
$\mathcal{T}_4^{\mathrm{H}}$	8739	1043	10,025	420	1976
$\mathcal{T}_5^{\mathrm{H}}$	8801	1047	11,297	410	2056

Note: All costs are given in 1000 NOK.

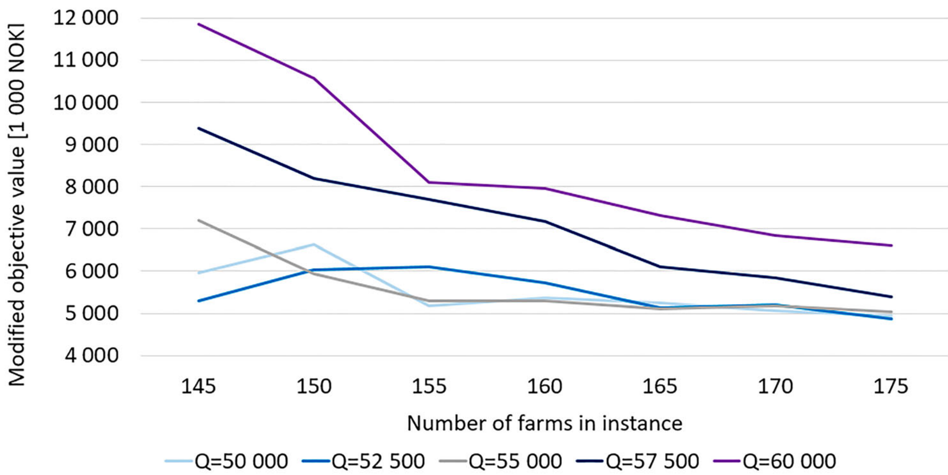

Figure 8. Modified objective value for various number of farms and demand scenarios. The modified objective function summarises the compensation cost and the costs of over- and under-delivery.

approximately 17%. With this alternative, we avoid weekend work. However, the extra wages and costs of adding this hatching day should be weighted against the potential savings. As can be observed from Table 10, the most noteworthy cost savings arise for the issue of non-uniform chickens. By adding a weekly hatching day or allowing for weekend work at the hatchery, the slaughterhouse can reduce waste by processing chickens that are closer to the target slaughter weight.

Thirdly, since the case company has ambitions to grow, we investigated the effect of extending the number of farms in combination with increased demand. The target demand, which is 50,000 chickens per day, was increased with 5%, 10%, 15% and 20%, i.e. up to 60,000 chickens. To avoid lack of eggs, supply is set infinitely high. As a consequence, the discarding cost will dominate the objective value, and is therefore excluded from this analysis. Since the non-uniform cost naturally increases with the number of chickens sent to slaughter, this cost is also excluded. Hence, the cost elements included in this analysis are the compensation cost, and over- and under-delivery costs.

Figure 8 shows how the modified objective value varies for the five demand scenarios, and the number of farms ranging from 145 (as of today) up to 175. By looking at the current demand scenario of 50,000 chickens, it can be observed that the curve stabilises after 155 farms. The 55,000-scenario shows the same behaviour, while the curve for the 52,500-scenario peaks at 155 farms before it stabilises around 165 farms. Based on this, it seems that 155 farms is sufficient to handle an increased demand of up to 10%. Since more farms probably implies higher fixed costs, it can be argued that 155 farms is strictly better than the larger sets of farms for a daily demand up to 55,000. For even higher demands of 57,500 and 60,000, it cannot be concluded what the optimal number of farms is since the modified objective value for these scenarios do not stabilise, indicating that it is probably necessary with 175 or more farms.

6.4.2. *The value of optimisation*

By comparing optimised schedules with the current plans at the case company, we may uncover the potential for cost efficiency improvement. Scheduling broiler production might be difficult using manual planning and spreadsheets as the case company uses today. By examining the current plans at the case company, we observe that they are not always able to satisfy all constraints given by the mathematical model. For example, there are sometimes fewer cleaning days between

Table 11. Comparison of the detailed cost distribution for the case company's plan and RHH_3 applied on T90.

	Total discarding	Downtime compensation	Non-uniform chickens	Over-delivery	Under-delivery
Case company	2306	420	5294	175	1611
RHH_3	2320	402	4381	176	1067
Δ Cost [%]	+ 0.6	− 4.3	− 17.2	+ 0.6	− 33.8

Note: All costs are given in 1000 NOK.

chicken flocks than originally required. Furthermore, the catching team rules are sometimes disobeyed by allowing for more than the maximum number of visits. In the schedules obtained from optimisation, all constraints are satisfied and thus the schedules are more in line with the case company's goals and requirements. To avoid infeasibility, and make the case company's current schedules comparable with the optimised ones, we have relaxed some constraints for the current scheduling practice.

We have compared the objective values for the full-scale instance F145 with 90 and 180 days planning horizons using RHH_3 with the case company's three- and six-month schedules (due to insufficient data for the full-year plan, we were not able to compare this with our solution). Our results showed a potential cost savings of 15% and 24% for the cases with 90 and 180 days planning horizons, respectively. Table 11 shows the detailed results for the comparison broken down to the different cost components on the case with 90 days planning horizon. As we can observe, by using optimisation we obtain a significant reduction in the costs related to non-uniform chickens and under-delivery. We also get a reduction in the downtime compensation costs, while the the costs related to discarding and over-delivery are slightly higher.

All in all, we observe that optimisation significantly improves several cost elements of the CFSASP, especially the penalty costs of non-uniform chickens and under-delivery of demand, which might be difficult to capture with manual planning. Through optimisation, several issues are mitigated simultaneously, resulting in holistic and cost-efficient schedules.

7. Concluding remarks

In this paper, we present a new mixed integer programming (MIP) model for solving the Chicken Flock Sizing, Allocation and Scheduling Problem (CFSASP) arising in the supply chain for broiler production. The CFSASP is an extremely complex problem and to efficiently solve it, we have proposed two rolling horizon heuristics (RHHs). It is shown on a number of real test instances from a Norwegian case company (Norsk Kylling) that the RHHs provide very good results and outperform the commercial MIP solver.

One of the RHHs is further applied to investigate the CFSASP through economic analyses. Sensitivity analyses indicated that better solutions can be obtained by reducing the allowed age difference limits between parent hens for each chicken flock, and altering the sets of weekly hatching days by adding another hatching day or allowing for weekend work. Lastly, comparisons with current production plans at the case company showed that optimisation can significantly improve cost efficiency. The results uncovered largest potential for cost savings regarding the penalty costs of non-uniform chickens and under-delivery of demand. Such aspects are difficult to capture without the use of optimisation software. This emphasises the value of optimisation-based decision support for the scheduling of broiler production.

Due to the very promising results, the Norwegian case company, Norsk Kylling, is now running a project to develop and implement an optimisation-based decision support system based on the MIP model and solution methods shown in this paper.

As mentioned previously, the RHH can be viewed as version of constructive relax-and-fix heuristics. It could be interesting to also apply a local search fix-and-optimise heuristic on top of the RHH to attempt to iteratively improve this solution, see for example Baldo et al. (1994). However, since the solution quality of the RHH is at a very acceptable level and the case company cannot accept longer solution times, we leave this for future research. Another interesting thing to explore in the future, could be to see whether it is possible to tighten the mathematical formulation and reduce the solution times of the MIP solver, for example by adding valid inequalities. If so, this could also improve the efficiency of the RHH (and any potential local search fix-and-optimise heuristics).

The optimisation framework proposed in this paper is specifically developed for the Norwegian case company and includes real-life details that enhance applicability and relevance, such as for example that it follows the 'all in–all out' principle according to Norwegian regulations. Nevertheless, we argue that our contributions can also be of great value for and adapted to other chicken producers, as well as to other food producing industries (e.g. salmon) that follow the 'all in–all out' principle and have one type of breed.

Acknowledgements

We thank the anonymous reviewers for insightful comments and suggestions, which helped us improve the paper.

Disclosure statement

No potential conflict of interest was reported by the authors.

References

Abedi, A., and W. Zhu. 2016. “An Optimisation Model for Purchase, Production and Distribution in Fish Supply Chain – A Case Study.” *International Journal of Production Research* 55 (12): 3451–3464.

Andersson, H., K. Fagerholt, and K. Hobbesland. 2015. “Integrated Maritime Fleet Deployment and Speed Optimization: Case Study From RoRo Shipping.” *Computers & Operations Research* 55 (1): 233–240.

Asche, F., and T. Bjørndal. 2011. *The Economics of Salmon Aquaculture*. 2nd ed. Chichester: Blackwell.

Asche, F., A. L. Cojocaru, and B. Roth. 2018. “The Development of Large Scale Aquaculture Production: A Comparison of the Supply Chains for Chicken and Salmon.” *Aquaculture* 493 (1): 446–455.

Baldo, T. A., M. O. Santos, B. Almada-Lobo, and R. Morabito. 1994. “On Practical Resource Allocation for Production Planning and Scheduling with Period Overlapping Setups.” *European Journal of Operational Research* 75 (2): 275–286.

Boonmee, A., and K. Sethanan. 2016. “A GLNPSO for Multi-level Capacitated Lot-sizing and Scheduling Problem in the Poultry Industry.” *European Journal of Operational Research* 250 (2): 652–665.

Clark, A., B. Almada-Lobo, and C. Almeder. 2011. “Lot Sizing and Scheduling: Industrial Extensions and Research Opportunities.” *International Journal of Production Research* 49 (9): 2457–2461.

Copil, K., M. Wörbelauer, H. Meyr, and H. Tempelmeier. 2017. “Simultaneous Lotsizing and Scheduling Problems: A Classification and Review of Models.” *OR Spectrum* 39 (1): 1–64.

Dillenberger, C., L. F. Escudero, A. Wollensak, and W. Zhang. 1994. “On Practical Resource Allocation for Production Planning and Scheduling with Period Overlapping Setups.” *European Journal of Operational Research* 75 (2): 275–286.

Forsberg, O. I. 1996. “Optimal Stocking and Harvesting of Size-structured Farmed Fish: A Multi-period Linear Programming Approach.” *Mathematics and Computers in Simulation* 42 (2–3): 299–305.

Gribkovskaia, I., B. O. Gullberg, K. J. Hovden, and S. W. Wallace. 2006. “Optimization Model for a Livestock Collection Problem.” *International Journal of Physical Distribution & Logistics Management* 36 (2): 136–152.

Hart, E., P. Ross, and J. A. D. Nelson. 1999. “Scheduling Chicken Catching – An Investigation Into the Success of a Genetic Algorithm on a Real-world Scheduling Problem.” *Annals of Operations Research* 92 (1): 363–380.

Hume, D. A., C. B. A. Whitelaw, and L. A. Archibald. 2011. “The Future of Animal Production: Improving Productivity and Sustainability.” *The Journal of Agricultural Science* 149 (1): 9–16.

Jaimes, A. L., S. Z. Martínez, and C. A. C. Coello. 2009. “An Introduction to Multiobjective Optimization Techniques.” Chap. 3 in *Optimization in Polymer Processing*, edited by António Gaspar-Cunha and José António Covas, 29–57. New York: Nova Science Publishers.

Junqueira, R., and R. Morabito. 2019. “Modeling and Solving a Sugarcane Harvest Front Scheduling Problem.” *International Journal of Production Economics* 213: 150–160.

Maravelias, C. T., and C. Sung. 2009. “Integration of Production Planning and Scheduling: Overview, Challenges and Opportunities.” *Computers and Chemical Engineering* 33 (12): 1919–1930.

Mercé, C., and G. Fontan. 2003. “MIP-based Heuristics for Capacitated Lotsizing Problems.” *International Journal of Production Economics* 85 (1): 97–111.

Oliveira, G. A., and L. A. Lindau. 2012. “A Framework for Delivery Scheduling in the Poultry Industry.” *Journal of Scheduling* 15 (1): 757–772.

Oppen, J., and A. Løkketangen. 2008. “A Tabu Search Approach for the Livestock Collection Problem.” *Computers & Operations Research* 35 (10): 3213–3229.

Perez, C., R. de Castro, and M. Font i Furnols. 2009. “The Pork Industry: A Supply Chain Perspective.” *British Food Journal* 111 (3): 257–274.

Plà, L. M., D. L. Sandars, and A. J. Higgins. 2014. “A Perspective on Operational Research Prospects for Agriculture.” *The Journal of the Operational Research Society* 65 (7): 1078–1089.

Rakke, J. G., M. Stålhane, C. R. Moe, M. Christiansen, H. Andersson, K. Fagerholt, and I. Norstad. 2011. “A Rolling Horizon Heuristic for Creating a Liquefied Natural Gas Annual Delivery Program.” *Transportation Research Part C: Emerging Technologies* 19 (5): 896–911.

Ramezaniana, R., M. Saidi-Mehrabada, and P. Fattahi. 2013. “MIP Formulation and Heuristics for Multi-stage Capacitated Lot-sizing and Scheduling Problem with Availability Constraints.” *Journal of Manufacturing Systems* 32 (2): 392–401.

Rodríguez, S. V., L. M. Plà, and J. Faulin. 2014. “New Opportunities in Operations Research to Improve Supply Chain Efficiency.” *Annals of Operations Research* 219 (1): 5–23.

Rodríguez-Sánchez, S. V., L. M. Plà-Aragonés, and V. M. Albornoz. 2012. "Modeling Tactical Planning Decisions Through a Linear Optimization Model in Sow Farms." *Livestock Science* 143 (2–3): 162–171.

Statista. 2019. "Production of Meat Worldwide from 2016 to 2018 (in Million Metric Tons)." Accessed 23 April 2019. https://www.statista.com/statistics/237632/.

Stolletz, R., and E. Zamorano. 2014. "A Rolling Planning Horizon Heuristic for Scheduling Agents with Different Qualifications." *Transportation Research Part E: Logistics and Transportation Review* 68 (1): 39–52.

Taube-Netto, M. 1996. "Integrated Planning for Poultry Production At Sadia." *Interfaces* 26 (1): 38–53.

Wilkinson, J. M. 2011. "Re-defining Efficiency of Feed Use by Livestock." *Animal* 5 (7): 1014–1022.

Yakovleva, N., and A. Flynn. 2004. "Innovation and Sustainability in the Food System: A Case of Chicken Production and Consumption in the UK." *Journal of Environmental Policy & Planning* 6 (3–4): 227–250.

You, P.-S., and Y.-C. Hsieh. 2018. "A Study of Production and Harvesting Planning for the Chicken Industry." *Agricultural Economics – Czech* 64 (7): 316–327.