

# Acknowledgements

First and foremost, I would like to express my sincere gratitude to my supervisors Stein W. Wallace, Julio C. Goetz and Francesca Maggioni. I would also like to extend my thanks to my co-authors Jamie Fairbrother and Geir D. Berentsen. I am very grateful for everything you have taught me during this journey, for interesting discussions, for your valuable feedback, and for your kindness. A special thanks to Francesca for inviting me to an extended stay in Bergamo (Italy), and to Jamie (and the STOR-i centre) for inviting me to Lancaster (UK). I would also like to extend my thanks to Erlend Torgenes and Penelope Melgarejo for good conversations and invaluable insights into aquaculture.

Many thanks to the Department of Business and Management Science at the Norwegian School of Economics for hosting my PhD and for providing a friendly working environment, and thank you to the Norwegian Research Council for financial support. I extend my thanks to all fellow PhD students for the time we spent together.

I want to thank for interesting discussions and good conversations at workshops and conferences, and special thanks to the organisers for providing this opportunity: Ettore Majorana Foundation, ESCO-CMS, ICSP, NORS, ULTRAOPTYMAL, EURO, CEMS SCM, INFORMS TSL, and BFE.

Last, but not least, I would like to thank all my friends and family for their support and good moments during this time.



# Contents

<b>Acknowledgements</b>	<b>i</b>
<b>Introduction and background</b>	<b>vii</b>
1 Decision modelling under uncertainty . . . . .	vii
1.1 Overview of decision modelling under uncertainty . . . . .	viii
1.2 Stochastic programming . . . . .	ix
1.3 Scenario generation . . . . .	x
2 Problem-based representations of uncertainty . . . . .	x
2.1 Sparsity in problem structure . . . . .	xi
3 Aquaculture Operations . . . . .	xii
<b>Papers</b>	<b>1</b>
<b>I Decomposing Output Distributions</b>	<b>3</b>
1 Introduction . . . . .	3
2 Background . . . . .	4
2.1 Difficult distributions . . . . .	5
2.2 Literature on problem-based scenario generation . . . . .	6
3 Method . . . . .	7
3.1 Mathematical developments . . . . .	8
3.2 Scenario reduction program . . . . .	13
4 Computational results . . . . .	15
4.1 Case study problems . . . . .	16
4.2 Scenario generation benchmark methods . . . . .	16
4.3 Empirical evaluation of scenario generation . . . . .	17
4.4 Experimental setup . . . . .	19
4.5 Results . . . . .	19
5 Discussion . . . . .	25
<b>II Bounds and Approximations</b>	<b>29</b>
1 Introduction . . . . .	29
2 Background and notation . . . . .	31
2.1 Stochastic linear programs . . . . .	33
3 Bounds on the evaluation of expectations . . . . .	34
3.1 Guaranteed bounds as a Generalised Moment Problem . . . . .	35
3.2 Generalisation to every decision and stochastic dominance . . . . .	38
3.3 Distribution approximations . . . . .	40

3.4	Function approximations . . . . .	47
3.5	Monte Carlo integration . . . . .	49
4	Bounds on optimal expected values . . . . .	49
4.1	Bounds based on candidate policies . . . . .	50
4.2	Bounds on refinement of information structure . . . . .	51
4.3	Bounds based on decision rules and aggregation . . . . .	54
5	Discussion and conclusions . . . . .	57
<b>III</b>	<b>Joint Forecasting</b>	<b>65</b>
1	Introduction . . . . .	65
2	Data . . . . .	67
3	Methodology . . . . .	70
3.1	Lice abundance model . . . . .	71
3.2	Treatment model . . . . .	78
3.3	Joint forecasting . . . . .	79
3.4	Estimation . . . . .	80
4	Results and validation . . . . .	81
4.1	Model interpretation and in-sample analysis . . . . .	81
4.2	One-week-ahead forecasting ability . . . . .	84
4.3	Long-term joint forecasting . . . . .	85
5	Discussion and concluding remarks . . . . .	88
<b>IV</b>	<b>Harvest Planning under Uncertainty</b>	<b>93</b>
1	Introduction . . . . .	93
1.1	Background . . . . .	93
1.2	Literature . . . . .	94
2	Problem statement and modelling considerations . . . . .	95
2.1	Modelling considerations . . . . .	95
3	Methodology and mathematical formulation . . . . .	98
3.1	Notation on stochastic processes and multistage stochastic programs . . . . .	98
3.2	Mathematical formulation . . . . .	99
3.3	Out-of-sample validation . . . . .	104
3.4	Scenario generation . . . . .	106
4	Case studies . . . . .	108
4.1	Evaluation metrics . . . . .	108
4.2	Experimental setup . . . . .	110
4.3	Validation . . . . .	111
4.4	Managerial insights . . . . .	114
5	Discussion and conclusion . . . . .	119
	<b>Supplemental Materials</b>	<b>123</b>
<b>A</b>	<b>Decomposing Output Distributions</b>	<b>125</b>
1	Telecommunication Network Planning . . . . .	125
2	Multidimensional News-vendor with Substitution . . . . .	127
3	Airlift Operations Scheduling . . . . .	130
4	Storage Layout and Routing . . . . .	133

<b>B Joint Forecasting</b>	<b>137</b>
1 Biological relations . . . . .	137
1.1 Development time between stages and hatch rate . . . . .	137
1.2 Time window and infectivity . . . . .	137
2 Structural instability . . . . .	139
3 Quantifying the cost of lice treatments . . . . .	139
4 Forecasts distributions for additional sites . . . . .	141
<b>C Harvest Planning under Uncertainty</b>	<b>147</b>
1 Statistical and biological models . . . . .	147
1.1 Market prices . . . . .	147
1.2 Lice and treatments . . . . .	148
1.3 Mortality . . . . .	149
1.4 Growth . . . . .	150
2 Dual formulation under relaxation of integers . . . . .	150
3 Problem instances . . . . .	152
4 K-means clustering on trees . . . . .	152



# Introduction and background

## 1 Decision modelling under uncertainty

Decision-making under uncertainty is concerned with modelling availability of information and how to effectively act in uncertain environments. Rarely are important decisions in logistics, finance or operations made under complete information, either because such information is inherently unavailable or because it is difficult or very expensive to obtain. Dealing with such uncertainty requires a clear understanding of how to act in uncertain environments: Should we accept uncertainty as a characteristic in itself, or is it sufficient to plan for a best-guess deterministic estimate of the future? As it turns out, lack of information requires different *strategies* than if planning for a deterministic future (Wallace, 2010). Once we accept uncertainty as a relevant characteristic, we would apply hedging strategies using options to deal with lack of information. This is best handled by considering all possible realisations of uncertainty simultaneously, and incorporating the flexibility to deal with all of them.

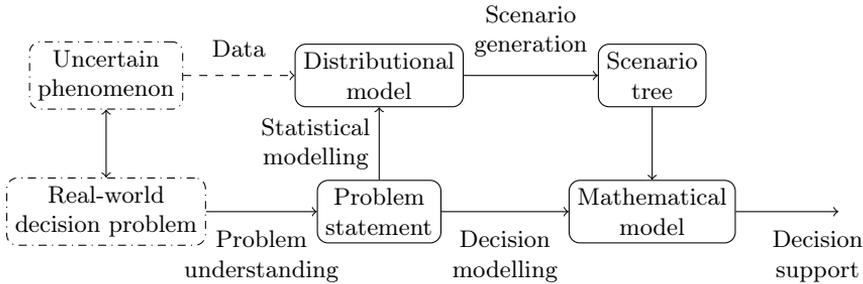
Uncertainty is decision-relevant if obtaining more information would change decisions to some meaningful degree. When accounting for uncertainty in decision problems, there is often an *emergence of options* in optimal solutions. An *option* (in a wide sense) here refers to a strategy that enables other actions in the future. If options come at a cost, they can often seem unprofitable with respect to a single realisation of uncertainty, but optimal when accounting for multiple realisations simultaneously. Formally, this effect can be quantified by a *shadow price of information* which provides a certificate of optimality (Rockafellar & Wets, 1991).

In logistics applications, options may take the form of strategically placed buffers to hedge the uncertainty of not knowing where it is needed. Allocating too much resources where demand is unfulfilled is sub-optimal for a single realisation, but the lack of information about where resources are needed *makes* such buffers optimal. Some other examples of options are financial options that enable buying or selling stocks in the future, and insurances that pay off in unfortunate situations; however, both of these require subjective risk preferences to be deemed profitable in a zero-sum game. In logistics and operations, however, there exist situations where the costs of options are less than their value, meaning their surplus value need only be collected.

Uncertainty can be classified into two broad categories characterised as (i) unknown unknowns, and (ii) known but unpredictable. We primarily refer to the latter, also known as *stochastic* uncertainty, where a phenomenon is unpredictable but can be *described precisely*. Unknown unknowns are relevant from a risk management perspective, but are much more difficult to model.

## 1.1 Overview of decision modelling under uncertainty

Modelling is a very important aspect of developing computational decision support tools since many real-world decision problems contain too much complexity to be formulated as mathematical models that can also be solved within reasonable time. Modelling aims to capture the most important aspects of the decision problem at hand, in order to provide valuable insights and actionable recommendations. Figure 1 gives a schematic overview of decision modelling under uncertainty.



**Figure 1:** Overview of decision modelling under uncertainty.

A *real-world decision problem* represents some situation where we think results can be improved by applying decision support tools. Common applications include logistics, finance, engineering, and generally complicated decision-making problems. An important consideration is that models are often much better than humans at making well-balanced trade-offs in large complicated systems. Another factor is that humans are inherently bad at accounting for uncertainty unless specifically trained for it (and sometimes not even that helps) (Kahneman et al., 1982). In complex environments, real-world decision problems can be difficult to approach, and we rely on a good *problem understanding* to formulate a more precise *problem statement*. The aspect of understanding a decision problem within its real-world environment often cannot be validated until its recommendations are put into practice, it must instead rely on diligent reasoning.

An unpredictable *uncertain phenomenon* may cause some decision-relevant lack of information. Uncertain phenomena might represent nature, our beliefs about the world, or future states of unpredictable systems. To account for uncertainty, the uncertain phenomenon must be described precisely using a *distributional model*. Prescribing a distributional model is referred to as *statistical modelling*, and is often supported by data used for estimation.

A *mathematical model* is formulated with respect to the problem statement, and the aim of the mathematical model is to prescribe effective actions. Mathematical models are here solved by optimisation, where prescribed actions are found by optimising an objective function with respect to some decision variables. There is a correspondence between models and prescribed actions to a problem statement, so that one maps to the other, but not necessarily by exact representation. An important aspect of decision modelling is to account for the technical capabilities of solving optimisation models computationally, which might require simplification.

Distributional models are often too rich to be used directly in a mathematical model, either because they are described by continuous distributions or because their outcome space is too large to allow for numerical implementation. Instead, we approximate uncertainty using a discrete and parsimonious *scenario tree* representation of the distributional

model, and we refer to the techniques of do so as *scenario generation*. A fundamental concern is whether scenario tree representations give solutions that perform well with respect to the distributional model we originally prescribed. Validation is essential to ensure solutions are sufficiently precise, which can be achieved by evaluating mathematically derived bounds on quantities of interest.

Paper I proposes a scenario generation method that exploits characteristics of the mathematical model to find more parsimonious representations of uncertainty with very weak assumptions about the specific model. Paper II describes approaches of evaluating and comparing results from a mathematical model directly with respect to a distributional model, even when scenario generation is required to actually find solutions. The last two papers address harvest planning under uncertainty where the entire modelling process must be considered. Paper III proposes a distributional forecasting model used to assess biological risk (developing in space and time), while Paper IV proposes a mathematical model for harvest planning to account for biological, operational and market risk, within a portfolio of heterogeneous sites.

## 1.2 Stochastic programming

Stochastic programming was first introduced by Dantzig (1955) as an extension to linear programming to account for unknown demand in logistics. Today, one of the main advantages of stochastic programming is its close ties to mathematical programming and its ability to effectively deal with constraints and combinatorial considerations through integer variables. The technology to solve large-scale mathematical programs is also well developed. There are still many aspects that are specific to stochastic programming (Birge & Louveaux, 2011; Kall & Wallace, 1994; King & Wallace, 2012; Shapiro et al., 2014). As opposed to other approaches to decision-making under uncertainty, and perhaps the most important aspect of stochastic programming, is that it *finds* new hedging strategies instead of simply evaluating the value of a predetermined strategy (Wallace, 2010).

Stochastic programs address different *stages* of decision-making, where earlier stages lack information about what will happen in later stages. Typically, this is stated as an optimisation problem

$$\min_{x \in \mathcal{X}} \{\mathbb{E} [F(x, \xi)]\} \quad (1)$$

where  $F(x, \xi)$  represents a *recourse function*<sup>1</sup> to describe the effects of a decision  $x$  across uncertain realisations of the parameters  $\xi$ . Typically, the recourse function is a (reasonably well behaved) optimisation problem to reflect that corrective actions are applied once new information arrives. In this sense, the aim of choosing an appropriate decision  $x$  is to enable enough flexibility to deal with a variety of future outcomes when current decisions constrain future actions. Naturally, evaluation of the expectation

$$\mathbb{E} [F(x, \xi)], \quad (2)$$

is challenging since it consists of solving an integral over an optimisation problem. To reduce computational effort, we must limit the number of evaluation points, and instead approximate (2) using a scenario representation.

*Two-stage programs* are often posed to reflect interactions between different scopes of planning. For example, the initial decision could be to determine production capacities while the second decision is to determine production plans within these capacities for a given realisation of demand. The performance of production plans then guide how

---

<sup>1</sup>The recourse function here represents either a second-stage or a multistage program.

capacities are allocated. *Multistage programs*, on the other hand, typically represent operational problems of a dynamic nature where timing might be important. For two-stage programs, uncertainty is described by a *random variable* while for multistage programs, uncertainty is described by a *stochastic process*. A major difference between these is that approximations of stochastic processes also require representations of information structure (i.e., how information develops over time). Ultimately, these formulations aim to reflect the effect of decisions made today, subject to uncertain realisations of the future.

### 1.3 Scenario generation

Scenario generation is effectively about approximating integrals, and extra care is required when these are embedded into optimisation problems. We use a *scenario set* to represent a random variable (two-stage), while a *scenario tree* is used to represent a stochastic process (multistage) where branching reflects development of information. Scenario representations should also be as small as possible since their size is directly proportional to the size of the resulting optimisation problem. Compared to other integral approximation techniques (like Quasi-Monte Carlo sampling and quadrature rules) special care is required since we optimise with respect to an approximation (also known as the *optimizer's curse*; Smith & Winkler, 2006) and the size of scenario approximations are typically much smaller.

Scenario generation can be particularly challenging since, by the nature of common applications of stochastic programming (Wallace & Ziemba, 2005), the number of random parameters is typically very high while special distributional forms (like heavy tails and mixtures) can be challenging to approximate. Paper I describes this issue further.

We distinguish between two main approaches to scenario generation: (i) distribution-based, and (ii) problem-based. The distribution-based approaches mainly look to the uncertain phenomenon to approximate some of its important characteristics. On the other hand, problem-based scenario generation aims to incorporate insights about the mathematical model into its representation of uncertainty. Fundamentally, the problem-based approach accepts that, in the context of decision-making, some realisations of uncertainty are more relevant to consider than others.

## 2 Problem-based representations of uncertainty

The aim of our problem-based scenario generation approach is to incorporate information about a specific problem without making strong assumptions about it. We refer to this as being *problem agnostic*. A problem-based scenario generation method that is agnostic to the problem could be applied to any new problem without extensive knowledge, while still making effective scenario generation more easily available.

Decision-making under uncertainty considers how current decisions affect future results, across different realisations of uncertainty. When making problem-based representations of uncertainty, our approach considers which *modes of change* decisions can cause in the *distribution of results*. We identify the primary modes of change and prioritise these to ensure the representation reflects the most important characteristics of the problem. Furthermore, there is a correspondence between the number of important modes of change and the required size of the representation, which reflects the need to account for multiple realisations of uncertainty. This further implies that problem-based representation of uncertainty is tightly linked to the *reason why* uncertainty matters in decision-making.

This is the topic of Paper I. Here, we elaborate a bit on the methodology to provide an overview, but note that some technical details are omitted.

## 2.1 Sparsity in problem structure

Mathematically, we represent uncertainty through *random variables*. The value of random variable  $Y(\omega)$  represents the realisation of some quantity  $Y$  in an outcome  $\omega$ , and the outcome space  $\omega \in \Omega$  represents the set of possible realisations where  $P$  assigns probabilities to outcomes. We use expectations

$$\mathbb{E}^P [Y] = \int_{\Omega} Y(\omega)P(d\omega), \quad (3)$$

with respect to probabilities  $P$  to summarise characteristics of random variables. To account for decisions, we assign a random variable

$$Y^x(\omega) = F(x, \xi(\omega)), \quad (4)$$

to each decision  $x$ , where  $\xi(\omega)$  denotes the stochastic parameters, and  $F(x, \xi)$  is a recourse function. A set of feasible decisions  $\mathcal{X}$  generates a collection

$$\mathcal{Y} = \{Y^x(\omega) : x \in \mathcal{X}\}, \quad (5)$$

of corresponding random variables. To optimise  $x$ , we must find the most preferable random variable  $Y^x(\omega) \in \mathcal{Y}$  as evaluated by its expected value but, in practice, the expected value can only be evaluated in terms of an alternative distribution  $R$ .

Fundamentally, we may think of  $Y(\omega) \in \mathcal{Y}$  as mappings from outcomes  $\omega \in \Omega$  to objective values, which is constructive since probabilities are assigned to outcomes. The stochastic parameters represented by  $\xi$  can be challenging to represent due to high dimensionality, and we argue that emphasising  $\xi$  leads to unneeded redundancy. Instead, we emphasise  $\mathcal{Y}$  directly for making an alternative distribution  $R$ .

To extract the key characteristics of  $\mathcal{Y}$ , we want to represent it in terms of a sparse basis.<sup>2</sup> If we assume  $\mathcal{Y}$  lies in a decomposable space, we may represent each of its elements in terms of a basis  $\{u_i(\omega)\}_{i \in \mathcal{I}}$  such that

$$Y^x(\omega) = \sum_{i \in \mathcal{I}} c_i(x)u_i(\omega), \quad \forall Y^x(\omega) \in \mathcal{Y}. \quad (6)$$

Observe in particular that this decomposition decouples decisions  $x$  from random variables (functions of  $\omega$ ). The next step is to pick a sparse basis  $\{u_i(\omega)\}_{i \in \mathcal{B}}$  where  $\mathcal{B} \subset \mathcal{I}$  and where the size of  $\mathcal{B}$  is much smaller than  $\mathcal{I}$ . An additional detail here is that we prioritise better decisions when choosing the sparse basis, and rely on evaluation in terms of candidate decisions. The basis is used to enforce consistent expectations

$$\mathbb{E}^R [u_i] = \mathbb{E}^P [u_i], \quad \forall i \in \mathcal{B}, \quad (7)$$

where  $R$  represents a scenario set. Basic linear algebra shows that by enforcing consistent expectations on a basis, this consistency is preserved across all of its linear combinations.

---

<sup>2</sup>Sparsity is the concept that seemingly much detail can be described using small representations without sacrificing much on precision. This has received increasing attention, particularly within the domains of statistics and machine learning where the aim is to find patterns in vast amounts of data (Hastie et al., 2009).

Furthermore, approximation errors can be quantified in terms of the projection distance onto the linear sub-space generated by the basis.

Whether  $\mathcal{Y}$  can be represented by a sparse basis is for all practical purposes an empirical question. This is why we support the methodology by extensive numerical experiments. The interpretation of sparsity is that there is strong structure to the problem, meaning the random variables in  $\mathcal{Y}$  share similarities. Formally, this can be analysed through the stability of

$$x \mapsto Y^x, \tag{8}$$

to quantify how sensitive the random variables  $Y^x$  are to changes in decisions. However, the mapping  $x \mapsto Y^x$  would also be very complicated, meaning empirical examination is ultimately more practical.

### 3 Aquaculture Operations

Harvest planning in Norwegian salmon aquaculture faces a multitude of uncertain factors and operational limitations, coupled with large downside risk due to large capital binding in the fish stock and risk of mortality. Farmers manage a portfolio of highly heterogeneous sites subject to company-wide limiting constraints which greatly increase the complexity of the decision problem. The source of biological risk is the immediate environment of the fish, and these are density-driven phenomena (parasites and disease) that develop in both space and time. Combined, these considerations make the *Aquaculture Harvest Planning* (AHP) problem well suited for application of quantitative decision support tools that account for uncertainty. The major challenges can be summarised as:

- Parasitic salmon lice and associated treatment actions
- Large short- and medium-term price fluctuations
- Fish health and mortality
- Regulatory restrictions on production capacity
- Harvest operations and well-boat logistics

In essence, harvest planning consists of deciding *when and where to harvest* within a *portfolio of sites* subject to a *dynamic* and *stochastic* environment. The decision problem compares the current value of biomass against the alternative value of waiting, with respect to stochastic risk and future operational flexibility. Due to portfolio effects, both *sequencing* and *timing* are important considerations. We solve the decision problem using multistage stochastic programming.

To hedge overall operational risk, it is essential to determine heterogeneity in risk exposure among sites to determine the most effective harvesting sequence. The major source of risk heterogeneity is salmon lice, whose development is strongly affected by ocean currents (transporting lice between sites) and temperature (governing lice growth and reproduction). Paper III addresses joint forecasting of salmon lice and treatment interventions among all aquaculture sites along the Norwegian coastline where we combine modelling of on-site lice dynamics with hydrodynamic simulation of stream patterns to account for spatial dependence. Treatments are modelled as an exogenous stochastic process which interacts with the stochastic process for lice development. This modelling assumption avoids decision-dependent uncertainty, and we argue this is reasonable based

on properties of the decision problem. The paper concludes there is large heterogeneity among sites at very significant levels of risk exposure.

The decision model is introduced in Paper IV together with modelling assumptions, techniques for representing uncertainty, and validation of solutions against the underlying stochastic process. The decision model uses the forecasting model from Paper III together with other (less extensive) forecasting models for the other risk factors. The formulation is a large-scale mixed-integer multistage stochastic program. The main techniques that enable solving this decision problem are precise modelling assumptions and parsimonious representations of uncertainty. The paper concludes that the decision model prescribes effective solutions to a large variety of situations within a complex environment. The overall risk exposure is large, and we find effective hedging strategies that improve results to a considerable degree.

## References

- Birge, J. R., & Louveaux, F. (2011). *Introduction to stochastic programming* (2nd ed.). Springer New York, NY. <https://doi.org/10.1007/978-1-4614-0237-4>
- Dantzig, G. B. (1955). Linear Programming under Uncertainty. *Management Science*, 1(3/4), 197–206. Retrieved May 10, 2023, from <https://www.jstor.org/stable/2627159>
- Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning*. Springer New York. <https://doi.org/10.1007/978-0-387-84858-7>
- Kahneman, D., Slovic, P., & Tversky, A. (Eds.). (1982). *Judgment under uncertainty: Heuristics and biases*. Cambridge University Press.
- Kall, P., & Wallace, S. W. (1994). *Stochastic Programming*. John Wiley & Sons.
- King, A. J., & Wallace, S. W. (2012, June). *Modeling with Stochastic Programming*. Springer New York, NY. <https://doi.org/10.1007/978-0-387-87817-1>
- Rockafellar, R. T., & Wets, R. J.-B. (1991). Scenarios and Policy Aggregation in Optimization Under Uncertainty. *Mathematics of Operations Research*, 16(1), 119–147. <https://doi.org/10.1287/moor.16.1.119>
- Shapiro, A., Dentcheva, D., & Ruszczyński, A. P. (2014). *Lectures on stochastic programming: Modeling and theory* (Second edition). Society for Industrial and Applied Mathematics : Mathematical Optimization Society. <https://doi.org/10.1137/1.9781611973433>
- Smith, J. E., & Winkler, R. L. (2006). The Optimizer’s Curse: Skepticism and Postdecision Surprise in Decision Analysis. *Management Science*, 52(3), 311–322. <https://doi.org/10.1287/mnsc.1050.0451>
- Wallace, S. W. (2010). Stochastic programming and the option of doing it differently. *Annals of Operations Research*, 177(1), 3–8. <https://doi.org/10.1007/s10479-009-0600-x>
- Wallace, S. W., & Ziemba, W. T. (Eds.). (2005). *Applications of stochastic programming*. Society for Industrial and Applied Mathematics.



# Papers



# Paper I

# Problem-based Scenario Generation by Decomposing Output Distributions

Benjamin S. Narum, Jamie F. Fairbrother and Stein W. Wallace

## Abstract

Scenario generation is required for most applications of stochastic programming to evaluate the expected effect of decisions made under uncertainty. We propose a novel and effective problem-based scenario generation method for two-stage stochastic programming that is agnostic to the specific stochastic program and kind of distribution. Our contribution lies in studying how an output distribution may change across decisions and exploit this for scenario generation. From a collection of output distributions, we find a few components that largely compose these, and such components are used directly for scenario generation. Computationally, the procedure relies on evaluating the recourse function over a large discrete distribution across a set of candidate decisions, while the scenario set itself is found using standard and efficient linear algebra algorithms that scale well. The method's effectiveness is demonstrated on four case study problems from typical applications of stochastic programming to show it is more effective than its distribution-based alternatives. Due to its generality, the method is especially well suited to address scenario generation for distributions that are particularly challenging.

## 1 Introduction

Stochastic programming is a useful tool for decision-making when there is uncertainty in the effects of decisions. It allows us to explicitly account for this uncertainty by building flexibility into decisions in such a way that the decision-maker is prepared for multiple future outcomes. As opposed to deterministic approaches, it achieves this by accounting for multiple outcomes *simultaneously* (Wallace, 2010). In particular, one models uncertain parameters as random variables and optimises the expectation of some utility or cost function (see Birge & Louveaux, 2011; Kall & Wallace, 1994; Shapiro et al., 2014).

Applications of stochastic programming include logistics, finance and engineering (King & Wallace, 2012; Wallace & Ziemba, 2005). A common characteristic of such problems is that quantifying the effect of a decision conditional on a future outcome involves modelling and optimising future decisions that explicitly exploit the *gain of additional information* within the limitations set by the current decision. This means calculating the expected future effect of decisions involves evaluating an integral over a function defined by an optimisation problem, which is usually analytically intractable and for which standard numerical integration may also be impractical due to high dimensionality. Scenario generation is the often used viable alternative. This consists of finding a discrete set of scenarios that approximately represents the distribution, which is a requirement for finding solutions to a majority of such models. A fundamental concern is getting the most parsimonious scenario set possible while still obtaining good solution with respect to the original formulation.

Stochastic programming problems are *distribution optimisation problems* (Wets, 1996). For every possible decision, what we get is a distribution of the potential effects of that decision. This perspective is fundamental to the current paper. Our contribution lies in detecting the structure of how such distributions may change as the decision changes and exploiting this for the purpose of scenario generation. For this to be effective there must be degeneracy (i.e., sparsity) in a problem’s output distributions; namely, it must be possible to explain large collections of output distributions by relatively few components. Our problem-based scenario generation can be applied to very general forms of two-stage stochastic programming problem for a wide variety of input distributions. We illustrate its effectiveness on four case study problems and show these have a natural degeneracy well suited for scenario generation. Furthermore, a minimum number of scenarios required for a given level of accuracy is suggested by the method itself. The method requires, as input, numerical evaluations of the recourse function for a set of candidate decisions over a discrete distribution, while the scenario reduction program itself relies on standard linear algebra and linear programming; hence, it scales well.

The paper is structured as follows: Section 2 gives more background and motivation for scenario generation, and highlights our contributions relative to previous literature; Section 3 presents our method alongside its mathematical underpinnings; Section 4 illustrates the method’s effectiveness on four case study problems; Section 5 gives some additional discussion; finally, we conclude the paper.

## 2 Background

Two-stage stochastic programs with recourse take the form

$$\min_{x \in \mathcal{X}} \mathbb{E} [f(x) + Q(x, \xi)]$$

where  $x$  is a (constrained) decision made under uncertainty,  $f(x)$  a deterministic cost, and  $Q(x, \xi)$  the *recourse function* that determines the future effect of the decision. We refer to the integrand (in the expectation) as the *output distribution*, whose expectation is to be approximated using a scenario set, and the distribution of the random vector  $\xi$  is referred to as the *input distribution*.

Stochastic programming models, by the nature of their applications, often have an input distribution of high dimensionality and this dimensionality also often grows with the level of detail or scope of such a model. Fundamentally, this means scenario generation aims to approximate integrals over high dimensional distributions which, in principle,

necessitates many scenarios to adequately represent the distribution. However, having few scenarios is desirable since each scenario directly adds to the size of a stochastic program by duplicating variables and constraints, which can greatly impair the tractability of solving these within a reasonable time.

Generally, we cannot know *a priori* if a given scenario set is more appropriate than others. Instead, we benchmark scenario sets by comparing the performance of their corresponding optimal solutions within the original formulation. Kaut and Wallace (2007) refers to this as *out-of-sample evaluation*. That is, scenario generation methods should be assessed by the quality of candidate decisions they provide. Using this criterion, it turns out relatively small scenario sets can be effective for very large problems.

Methods for scenario generation can be classed as either *distribution-based* or *problem-based*. Distribution-based methods aim to approximate the input distribution without explicit consideration of the problem at hand, while problem-based methods explicitly exploit problem-specific knowledge to make the scenario set more compact. Problem-based approaches are most useful for problems that have distributions that are especially difficult for scenario generation or where the stochastic program is especially computationally demanding to solve.

## 2.1 Difficult distributions

Some problems have input distributions that are particularly challenging to address with scenario generation. Typically, this occurs when the distribution has complicated patterns, with potentially high but rare impact. These are especially relevant for problem-based scenario generation since distribution-based alternatives may require a prohibitively large number of scenarios to reach similar precision. Examples of difficult distributions include binary (0/1 valued marginals), multi-modal, those having high-impact tails, and distributions where the stochastic variables are qualitatively different. Importantly, the ability to easily deal with complicated distributions widens the scope of what *kinds of problems* can be explored in the context of decision-making under uncertainty, as emphasised by Vaagen and Wallace (2008).

The fundamental issue encapsulated by binary and multi-modal distributions is how to pick which “peaks” in the distribution to incorporate into a scenario set while not knowing each of their impacts. Omitting some peaks could result in large misrepresentations of the objective. Binary distributions are particularly relevant since they can be used to encode stochastic on-off behaviour in decision problems, which can correspondingly lead to large differences in the objective value. High-impact tails are difficult in the sense that, with many dimensions, there is an exponential number of tails and picking the relevant ones without redundancy is hard.

Problems with qualitatively different sources of uncertainty are challenging for distribution-based scenario generation because the sensitivity to their respective outcomes can be very different. Re-scaling need not help either since a problem’s sensitivity on a stochastic parameter does not necessarily relate to the scale of its distribution. Examples include problems characterised by having operational risk (drivers of costs) combined with market risk (price, supply and demand), rates (in  $[0, 1]$ ) combined with free variables (in  $(-\infty, \infty)$ ) or combinations of discrete (binary in particular) and continuous random variables (i.e. mixed). Especially difficult is the case where their combined impact may be particularly large within certain ranges of the distribution support.

To illustrate the issue with difficult distributions, consider the extreme case where some stochastic variables used during scenario generation do not actually affect the objective in any way. They still compromise the quality of a distribution-based method, while an

effective problem-based method would simply ignore them. Distribution-based scenario generation is blind to the kinds of issues described by difficult distributions since it cannot know where the objective is sensitive to its random parameters.

Applications having binary distributions include: stochastic customers (Bent & Van Hentenryck, 2004), batch sizing (Lulli & Sen, 2004), link failure in networks (i.e. in telecommunications and transportation) (Ball et al., 1995), vehicle routing with back-hauls (Farahani et al., 2011), good-or-bad weather (Wang & Jacquillat, 2020), or their use as a modelling tool to encode known qualitative aspects of the uncertainty (Ni et al., 2017). Applications having mixture distributions include: hospital planning with both stochastic occurrence and operating time (Caunhye & Nie, 2018), environmental outcomes (Emmerling & Tavoni, 2018), and mixtures of heavy-tailed distributions and normal distributions in wind power (Tewari et al., 2011). Applications using multi-modal distributions include: fashion considering both the trend and demand (Vaagen & Wallace, 2008) and various case problems by Parpas et al. (2015).

## 2.2 Literature on problem-based scenario generation

It has been recognised that explicitly accounting for problem structure in scenario generation leads to more concise scenario sets than distribution-based approaches. Recent examples include the works by Bertsimas and Mundru (2022), Fairbrother et al. (2022), Guo et al. (2019), Hewitt et al. (2021), Keutchayan et al. (2023), Narum (2020) and Prochazka and Wallace (2020).

The approach by Zhao and Wallace (2016) is a very clear-cut example of incorporating problem knowledge into scenario generation; the scenarios were simply hand-picked based on understanding the problem. The caveat with such an approach is that other problems are much harder to understand well enough to pick good scenarios. Fairbrother et al. (2022) showed that for some problems with tail risk measures, such as conditional value-at-risk, the support of the input distribution has regions that cannot affect the objective function, and are thus irrelevant to include in a scenario set. Similar aims were pursued by Arpón et al. (2018), while Prochazka and Wallace (2018) gave an analogous result for binary distributions. Prochazka and Wallace (2020) develop a heuristic that fits a scenario set to minimise the difference between the in-sample and out-of-sample evaluations for a set of candidate decisions. They obtained highly effective scenarios and, importantly, observed similar tendencies in the scenario sets as implied analytically by Fairbrother et al. (2022). However, a caveat of their approach is that the fitting heuristic must be redesigned for each specific problem. Henrion and Römisich (2022) expanded the use of the Fortet-Mourier metric as a theoretical bound on the stability of stochastic programs for scenario generation that takes explicit account of the problem. Originally, this metric motivated minimum transportation approaches to scenario generation by its bound on approximation errors, but in such a way that all explicit information about the problem is lost before scenario generation (Dupačová et al., 2003; Pflug, 2001). The tractability of the approach by Henrion and Römisich (2022) on application-relevant problems is, however, unclear.

Recently, three problem-based clustering approaches have appeared (Bertsimas & Mundru, 2022; Hewitt et al., 2021; Keutchayan et al., 2023) that are also distribution and problem agnostic. The methods in these papers require finding every single-scenario solution (found by solving the problem with a single scenario) and for each of these, evaluating the objective over every outcome of the (discrete) input distribution. Scenario reduction is then done by clustering the outcomes according to a metric based on these objective evaluations. Independently, the authors of this paper proposed a similar approach using

candidate decisions *not* required to be single-scenario solutions (Narum, 2020). The issue with using single-scenario solutions is that they are often of low quality, and thus do not represent well the structure of a problem. Indeed, Wallace (2010) discusses in detail how stochastic programs solved using a single scenario leads to *option-free decisions*, meaning that if the flexibility to effectively deal with a collection of potential outcomes comes at a cost (which it often does) it will never be contained in a single-scenario optimal decision. That is, single-scenario solutions are *qualitatively different* from the ones using multiple scenarios. We advocate, in contrast to these other approaches, the use of finding candidate solutions by solving problems with *multiple* scenarios (as was done by Narum (2020)) by the argument that these are closer to the optimal solution(s) and thus provides more appropriate approximations (see Section 3.1). The computational experiments in this paper also demonstrate that our method is effective using relatively few candidate decisions and has low sensitivity on the number of such candidate decisions used (see Section 4.5).

Like with these problem-based clustering methods, the scenario reduction method proposed in this paper relies on the evaluation of all outcomes over a set of candidate solutions, but can be applied using significantly fewer (and more appropriate) candidate solutions than those using only single scenario solutions. Instead of clustering, our scenario reduction works with the vector space of output distributions and attempts to reduce scenarios while preserving the expectation on this space in as efficient a manner as possible. This explicitly ensures low overall bias between in-sample and out-of-sample objective evaluations, which is not necessarily the case for clustering.

### 3 Method

We are interested in solving problems of the following form:

$$\begin{aligned} & \underset{x}{\text{minimize}} && f(x) + \mathbb{E}_p [Q(x, \xi)] \\ & \text{subject to} && x \in \mathcal{X} \end{aligned} \tag{I.1}$$

where  $x$  represents some *decision* constrained to the deterministic set  $\mathcal{X}$ , and  $f : \mathcal{X} \rightarrow \mathbb{R}$  its deterministic cost. Uncertainty is represented by the random vector  $\xi$ . The *recourse function*  $Q : \mathcal{X} \times \Xi \rightarrow \mathbb{R}$  represents the effect of the decision given some outcome  $\xi$ , and is assumed throughout to be finite for any  $x \in \mathcal{X}$  and  $\xi \in \Xi$ . We assume the distribution of  $\xi$  is discrete with outcomes  $\Xi = \{\xi^s : s \in \mathcal{S}\}$  and corresponding vector of probabilities  $p = (p_s)_{s \in \mathcal{S}}$  where  $\mathcal{S} = \{1, \dots, S\}$ . We use the subscript  $p$  on expectations  $\mathbb{E}_p[\cdot]$  to emphasise its reliance on the probabilities of the mass points.

We refer to this discrete distribution as the *observed distribution*. This terminology is used since there may be some underlying input distribution which truly represents the uncertainty but which cannot be used with our method, for example, because it is a continuous analytical distribution or because a full distribution is unavailable. In the former case the observed distribution may be constructed by sampling from the analytical distribution, and in the latter case we may use an empirical distribution based on historical data. In either case, we assume that the observed distribution is large enough to accurately evaluate the expectation of the recourse function.

Ideally, we would solve the problem (I.1) as stated with the observed distribution, but this is often computationally intractable due to a large number of outcomes. The purpose of scenario generation is to construct a discrete distribution with relatively few outcomes with which to approximate  $\mathbb{E}_p [Q(x, \xi)]$ , and in such a way that still leads to good decisions of the original optimisation problem. For this paper, we are particularly

interested in constructing scenario sets whose outcomes are a subset of those of the observed distribution, a distinction usually referred to as *scenario reduction*. This can be viewed as finding a new vector of probabilities  $r = (r_s)_{s \in \mathcal{S}}$  with few non-zero elements.

The methodology we propose works with output distributions, limited to the part described by the recourse function. That is, for different decisions  $x$  we are interested in the *recourse vector* which is defined as follows:

$$Q^x = (Q(x, \xi^s))_{s \in \mathcal{S}}. \quad (\text{I.2})$$

For a given set of *candidate decisions*, the corresponding collection of recourse vectors can be decomposed using singular value decomposition; consequently, the most important components (as inferred by their singular values) may be used for scenario generation to fulfil  $\mathbb{E}_r [Q(x, \xi)] \approx \mathbb{E}_p [Q(x, \xi)]$  parsimoniously within this candidate set.

If recourse vectors of other decisions are largely composed of the same components used for scenario generation, this approximation will remain good and yield high-quality solutions when used to solve (I.1). The required mathematical developments are presented in Section 3.1 and the numerical implementation is explained in more detail in Section 3.2.

### 3.1 Mathematical developments

We formulate the scenario reduction problem within the framework of inner product spaces weighted by probability. This enables us to analyse recourse vectors and to give results about the errors of our approximations. Consider the inner product space  $W$  defined on the vector space  $\mathbb{R}^S$  equipped with weighted inner product

$$\langle d, h \rangle_W = \sum_{s \in \mathcal{S}} p_s d_s h_s, \quad d, h \in W, \quad (\text{I.3})$$

where  $p$  is the probability vector of the observed distribution. This induces the norm  $\|d\|_W = \langle d, d \rangle_W^{1/2}$ . By using an inner product weighted by probability vector  $p$ , we take into account which outcomes contribute more to the calculation of the expectation. Note in particular that the recourse vectors  $Q^x$  defined above are elements in  $W$ .

The elements of  $W$  have the interpretation that they are finite discrete random variables with  $S$  outcomes. With this interpretation in mind, and a slight abuse of notation, we can express their expectation with respect to  $p$  as

$$\mathbb{E}_p [w] = \sum_{s \in \mathcal{S}} p_s w_s = \langle \mathbb{1}, w \rangle_W \quad (\text{I.4})$$

for any  $w \in W$  where  $\mathbb{1} = (1, \dots, 1)$ .

We now consider the evaluation of expectations of elements in  $W$  using probability vector  $r$  instead of  $p$ . The following result, which holds by the linearity of expectations, says that preserving expectations on a subspace of  $W$ , by using  $\mathbb{E}_r$  in place of  $\mathbb{E}_p$ , is equivalent to preserving expectations of any basis on that subspace.

**Lemma 1.** Suppose  $\{w^i\}_{i \in \mathcal{B}}$  where  $\mathcal{B} := \{0, \dots, B\}$  is a collection of linearly independent vectors in  $W$ , and let  $W_{\mathcal{B}} = \text{span}\{w^0, w^1, \dots, w^B\}$ . Let also  $p$  and  $r$  be two probability vectors. Then,  $\mathbb{E}_p[w] = \mathbb{E}_r[w]$  for all  $w \in W_{\mathcal{B}}$  if and only if  $\mathbb{E}_p[w^i] = \mathbb{E}_r[w^i]$  for all  $i \in \mathcal{B}$ .

Due to the linear independence, the probability vector  $r$  in this lemma requires at most  $B + 1$  non-zero elements to fulfil the condition of consistent expectations on  $W_{\mathcal{B}}$ . That is, the reduced scenario set would contain  $B + 1$  scenarios.

Our aim for scenario generation is to find an appropriate subspace  $W_{\mathcal{B}}$  that effectively approximates expectations of recourse vectors, given that we enforce consistent expectations between  $p$  and  $r$  on its basis. By a slight abuse of notation, let

$$W_{\mathcal{X}} = \text{span}\{Q^x : x \in \mathcal{X}\} \subseteq W, \quad (\text{I.5})$$

denote the span of all recourse vectors defined by decisions  $x \in \mathcal{X}$ . An interesting observation is that enforcing consistent expectations on the vector space spanned by recourse vectors of feasible decisions  $W_{\mathcal{X}}$  is sufficient to get exact results to (I.1). This is primarily interesting if  $W_{\mathcal{X}}$  has a smaller basis than  $W$  (to obtain a sparse  $r$ ) and is easily obtainable. In some special cases  $W_{\mathcal{X}}$  may easily be found but, generally, we consider this to be unavailable.

For a decision  $x \in \mathcal{X}$  and alternative probability vector  $r$  we define the *scenario approximation error* to be

$$\mathbb{E}_{p-r}[Q^x] := \mathbb{E}_p[Q^x] - \mathbb{E}_r[Q^x] = \sum_{s \in \mathcal{S}} (p_s - r_s) Q(x, \xi^s). \quad (\text{I.6})$$

To quantify the approximation error given a space  $W_{\mathcal{B}}$  on which we have consistent expectations, let  $P_{\mathcal{B}}$  be the projection operator onto  $W_{\mathcal{B}}$ . Denoting by  $I$  the identity operator, we then have that  $I - P_{\mathcal{B}}$  is the projection operator onto its orthogonal complement  $W_{\mathcal{B}}^{\perp}$ . Any recourse vector  $Q^x$  then has the decomposition

$$Q^x = P_{\mathcal{B}}Q^x + (I - P_{\mathcal{B}})Q^x. \quad (\text{I.7})$$

These operators allow us to bound the scenario approximation error as follows:

**Theorem 1.** Suppose  $W_{\mathcal{B}} \subseteq W$  is a subspace, and let  $r$  be an alternative probability vector with consistent expectations on  $W_{\mathcal{B}}$  with respect to probability vector  $p$ . Then, for any  $x \in \mathcal{X}$  we have the following bound on the (absolute) scenario approximation error:

$$|\mathbb{E}_{p-r}[Q^x]| \leq \|(I - P_{\mathcal{B}})Q^x\|_W \phi(p, r), \quad (\text{I.8})$$

where  $\phi(p, r) = (\sum_{s \in \mathcal{S}} (p_s - r_s)^2 p_s^{-1})^{1/2}$  is the square root of the  $\chi^2$ -distance from  $p$  to  $r$ .

*Proof.* Observe that

$$|\mathbb{E}_{p-r}[Q^x]| = |\mathbb{E}_{p-r}[P_{\mathcal{B}}Q^x + (I - P_{\mathcal{B}})Q^x]| = |\mathbb{E}_{p-r}[(I - P_{\mathcal{B}})Q^x]| \quad (\text{I.9a})$$

$$= \left| \sum_{s \in \mathcal{S}} ((I - P_{\mathcal{B}})Q^x)_s p_s^{1/2} p_s^{-1/2} (p_s - r_s) \right| \quad (\text{I.9b})$$

$$\leq \left( \sum_{s \in \mathcal{S}} p_s ((I - P_{\mathcal{B}})Q^x)_s^2 \right)^{1/2} \left( \sum_{s \in \mathcal{S}} (p_s - r_s)^2 p_s^{-1} \right)^{1/2} \quad (\text{I.9c})$$

$$= \|(I - P_{\mathcal{B}})Q^x\|_W \phi(p, r) \quad (\text{I.9d})$$

where (I.9a) follows from consistent expectations on  $W_{\mathcal{B}}$ , (I.9b) is multiplication by  $1 = p_s^{1/2} p_s^{-1/2}$ , and (I.9c) follows by the Cauchy-Schwartz inequality.  $\square$

Theorem 1 has an important interpretation. The factor  $\phi(p, r)$  is the square root of the  $\chi^2$ -distance (a  $\phi$ -divergence (Bayraksan & Love, 2015)) from the observed distribution  $p$  to

the scenario set defined by  $r$ . Very parsimonious scenario sets generally give higher values of  $\phi(p, r)$  and a worse bound on the approximation error; however, parsimonious scenario sets are also good for tractably finding solutions to stochastic programs. The factor  $\|(I - P_{\mathcal{B}})Q^x\|_W$ , on the other hand, is the projection distance onto  $W_{\mathcal{B}}$  which serves as a *problem-based measure* of how effective  $W_{\mathcal{B}}$  is in approximating expectations of recourse vectors for the given problem. An appropriate  $W_{\mathcal{B}}$  gives lower projection distance and a tighter bound on the approximation error. This is an explicit statement of the notion that *more effective representation of problem structure allows more parsimonious scenario sets*.

The aim of our problem-based scenario generation procedure may be stated as

$$\min_{W_{\mathcal{B}}} \|(I - P_{\mathcal{B}})Q^x\|_W^2, \quad (\text{I.10})$$

for a selection of decisions  $x$  relevant for approximation, where minimisation consists of choosing the basis for  $W_{\mathcal{B}}$ . Passing from  $W_{\mathcal{B}}$  to a scenario set is done by enforcing consistent expectations on its basis. We proceed to solve (I.10) heuristically by finding candidate decisions whose associated recourse vectors are assumed to be representative of other relevant recourse vectors and choosing their most important components.

### Candidate Decision Sets for problem-based scenario generation

To generate problem-based scenario sets, we consider a finite set of candidate decisions (with corresponding recourse vectors) of reasonably good quality. The motivation is that these can provide relevant components of recourse vectors that may be used for scenario generation. We justify the use of candidate decisions by a perturbation argument.

Denote the *candidate decision set* (CDS) as  $\mathcal{C} = \{x^1, \dots, x^K\} \subset \mathcal{X}$ , and the vector space spanned by its recourse vectors  $W_{\mathcal{C}} = \text{span}\{Q^x : x \in \mathcal{C}\} \subseteq W$ . Based on Theorem 1, showing that approximation errors are proportional to projection distance, we suggest finding candidate decisions of high quality and use their associated recourse vectors to make problem-based scenario sets. This builds on the assertion that a low perturbation in the decision  $\|x - \tilde{x}\|$  gives a low correspondingly perturbation in the recourse vector  $\|Q^x - Q^{\tilde{x}}\|_W$ .

Consider that there exists a non-decreasing function  $\psi : \mathbb{R}^+ \rightarrow \mathbb{R}^+$  such that

$$\|Q^x - Q^{\tilde{x}}\|_W \leq \psi(\|x - \tilde{x}\|), \quad (\text{I.11})$$

where  $\psi$  preferably takes as low values as possible (such a  $\psi$  always exists by the assumption that  $Q(x, \xi)$  is finite). Let the expressions

$$\text{dist}(x, \mathcal{A}) = \min_{\tilde{x} \in \mathcal{A}} \|x - \tilde{x}\|, \quad \mathcal{X}^* = \arg \min_{x \in \mathcal{X}} \{f(x) + \mathbb{E}_p [Q(x, \xi)]\}, \quad (\text{I.12})$$

denote the distance from a point to a set, and  $\mathcal{X}^*$  the optimal solution set. We primarily emphasise obtaining good approximations within some vicinity of the optimal solution set, and denote the set of relevant decisions as  $\tilde{\mathcal{X}} \supseteq \mathcal{X}^*$ . Assuming expectations are consistent on  $W_{\mathcal{C}}$ , we have that

$$|\mathbb{E}_{p-r} [Q^x]| \leq \psi(\text{dist}(x, \mathcal{C}))\phi(p, r), \quad \forall x \in \tilde{\mathcal{X}}, \quad (\text{I.13})$$

by (I.11) and Theorem 1. Hence, the bound on the approximation error (I.8) for decisions in  $\tilde{\mathcal{X}}$  is improved by having the candidate decisions  $\mathcal{C}$  be close to it. A tighter bounding function  $\psi$  naturally makes the approach using candidate decisions for scenario generation

more effective. In particular, if  $Q(x, \xi)$  is a second-stage linear program under fixed and complete recourse,  $\psi$  takes the form  $\psi(\|x - \tilde{x}\|) = A \|x - \tilde{x}\|$ , while if we additionally have integers and right-hand side uncertainty, it takes the form  $\psi(\|x - \tilde{x}\|) = A \|x - \tilde{x}\| + B$  (Schultz, 2000, Propositions 2.2 and 2.4). Prochazka and Wallace (2020) inspired this approach to problem-based scenario generation using candidate decisions.

The assumption that candidate recourse vectors are representative of new ones can be validated after-the-fact for a new decision  $x^*$  by evaluating  $\|(I - P_B)Q^{x^*}\|_W$ . If this projection distance is high, we may reconsider if the representation was good enough and proceed to find a more relevant set of candidate decisions. In Section 3.2, we discuss in more detail how candidate decisions can be found but assert for now that they should be of reasonable quality and not too similar.

### Decomposition over Candidate Recourse Vectors

We now address how the most important components of candidate recourse vectors can be extracted based on an excessively large set of candidate decisions. The aim is to get sparsity in the alternative probability vector  $r$  when enforcing expectations on these, while still getting a low approximation error. The recourse vectors of the CDS give rise to a *data matrix*  $M$  defined as

$$M = [Q^{x^1}, \dots, Q^{x^K}] \in \mathbb{R}^{S \times K}, \quad \{x^1, \dots, x^K\} = \mathcal{C} \quad (\text{I.14})$$

where  $Q^{x^k}$  for  $x^k \in \mathcal{C}$  are referred to as *candidate recourse vectors*. Equivalently,  $M_{sk} = Q(x^k, \xi^s)$ . The data matrix  $M$  is input for the scenario reduction method and is found by evaluating the recourse function over all scenarios of the observed distribution for each candidate decision.

To find the most relevant components of candidate recourse vectors, we work on linear combinations of these. Let  $\mathbb{R}^K$  be the space of possible weights used in linear combinations, equipped with the dot product and associated norm  $\|z\| = \langle z, z \rangle^{1/2}$ , where  $M(z)$  denotes a linear combination of candidate recourse vectors using weights  $z$ . We let  $z^k \in \mathbb{R}^K$  be defined such that  $Q^{x^k} = M(z^k)$  for  $k = 1, \dots, K$ , and note that  $\{z^1, \dots, z^K\}$  forms an orthonormal basis on  $\mathbb{R}^K$ . In what follows, we consider the collection of normalised linear combinations of recourse vectors

$$\{M(z) : \|z\| \leq 1\}, \quad (\text{I.15})$$

which also contains  $Q^{x^k}$  for every  $x^k \in \mathcal{C}$ . Geometrically, the set (I.15) forms an ellipse in  $W$  whose size represents the variety of recourse vectors that arise from the candidate decision set.

The most relevant components of candidate recourse vectors are found by decomposition. The singular value theorem (SVT) (Friedberg et al., 2002, Theorem 6.26) states that there exist an orthonormal basis  $\{v^1, \dots, v^K\}$  of  $\mathbb{R}^K$ , orthonormal basis  $\{u^1, \dots, u^S\}$  of  $W$  (with respect to inner product  $\langle \cdot, \cdot \rangle_W$ ), and unique *singular values*  $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_J > 0$  such that

$$M(v^i) = \begin{cases} \sigma_i u^i & \text{if } 1 \leq i \leq J \\ 0 & \text{if } i > J. \end{cases} \quad (\text{I.16})$$

where  $J$  is the rank of  $M$ . The vectors  $\{v^1, \dots, v^J\}$  and  $\{u^1, \dots, u^J\}$  are referred to as *right- and left-singular vectors*, and are unique up to the sign. This is referred to as the

singular value decomposition (SVD) of  $M$ . Observe that this SVD of  $M$  is *not* the same as the standard matrix SVD (which assumes the standard unweighted inner product).

In addition to preserving expectations for recourse vectors, we also ensure that the elements of  $r$  sum to one. An elegant way of doing this is to fix the first basis component to be  $\mathbb{1}$  since the condition  $\mathbb{E}_r[\mathbb{1}] = \mathbb{E}_p[\mathbb{1}]$  also implies  $\sum_{s \in \mathcal{S}} r_s = 1$ . To ensure the resulting singular vectors from SVD are orthogonal to  $\mathbb{1}$ , we therefore subtract the expectation with respect to  $p$  from the recourse vectors. Let  $\hat{M}$  denote the data matrix  $M$  whose columns are orthogonalised in this way.

For our purposes, the singular values essentially tell us how important each component  $u^i$  is in reconstructing the candidate recourse vectors. This suggests an effective way of preserving the expectation of these recourse vectors is to preserve the expectation of the singular vectors  $u^i$  corresponding to the largest singular values. Theorem 2 restates the bound on the approximation error from Theorem 1, limited to decisions in the CDS, when using left-singular vectors of  $\hat{M}$  to define the basis of  $W_{\mathcal{B}}$ .

**Theorem 2.** Let  $W_{\mathcal{B}} = \text{span}\{\mathbb{1}, u^1, \dots, u^B\}$  where  $u^i$  and  $\sigma_i$  are the left-singular vectors and values of  $\hat{M}$ , and let  $r$  fulfil  $\mathbb{E}_r[\tilde{w}] = \mathbb{E}_p[\tilde{w}]$  for all  $\tilde{w} \in W_{\mathcal{B}}$ . For any normalised linear combination of candidate recourse vectors  $w \in \{M(z) : \|z\| \leq 1\}$ , we then have that

$$|\mathbb{E}_{p-r}[w]| \leq \sigma_{B+1} \phi(p, r). \quad (\text{I.17})$$

*Proof.* By Theorem 1, we have that

$$\mathbb{E}_{p-r}[w] \leq \|(I - P_{\mathcal{B}})w\|_W \phi(p, r),$$

and by (I.16), we see that  $(I - P_{\mathcal{B}})w$  is spanned by a normalised linear combination of  $\{\sigma_{B+1}u^{B+1}, \dots, \sigma_J u^J\}$  whose norm must be bounded by  $\sigma_{B+1}$ .  $\square$

Adding consecutive left-singular vectors to  $W_{\mathcal{B}}$  of largest singular values implies  $\sigma_{B+1}$  gets lower and this bound tighter. Note that  $\phi(p, r)$  in (I.17) has conservative bound  $(\sup_s \{p_s^{-1}\} - 1)^{1/2} < \infty$ , which is tight only if  $r$  takes value one at a single element (where  $p$  is smallest) and zero otherwise. However, this does not account for the fact that  $r$  must also fulfil the constraints  $\mathbb{E}_r[u^i] = \mathbb{E}_p[u^i]$  for  $i \in \mathcal{B}$ . By finding  $r$  first, we may instead compute the bound (I.17) directly to obtain a less conservative value of  $\phi(p, r)$ . Furthermore, consider that the step (I.9c) in the proof of Theorem 1 conservatively decouples probabilities from the left singular vectors  $u^i$  when these are actually available. Proposition 1 utilises these instead to give an alternative bound on the approximation error that may be tighter than (I.17).

**Proposition 1.** With the same assumptions as in Theorem 2, we also have bound

$$|\mathbb{E}_{p-r}[w]| \leq \left( \sum_{i=B+1}^J \sigma_i^2 \mathbb{E}_p^p[-] r u^{i^2} \right)^{1/2} \quad (\text{I.18})$$

on the (absolute) approximation error for any normalised linear combination of recourse vectors  $w \in \{M(z) : \|z\| \leq 1\}$ .

*Proof.* Observe that

$$|\mathbb{E}_{p-r}[w]| = \left| \sum_{i=B+1}^J \sigma_i \mathbb{E}_{p-r}[u^i] \langle v^i, z \rangle \right| \quad (\text{I.19a})$$

$$\leq \left( \sum_{i=B+1}^J \sigma_i^2 \mathbb{E}_{p-r}[u^i]^2 \right)^{1/2} \left( \sum_{i=B+1}^J \langle v^i, z \rangle^2 \right)^{1/2} \quad (\text{I.19b})$$

$$\leq \left( \sum_{i=B+1}^J \sigma_i^2 \mathbb{E}_{p-r}[u^i]^2 \right)^{1/2} \quad (\text{I.19c})$$

where (I.19a) again follows by decomposition and enforcement of expectations on  $W_B$ , (I.19b) by Cauchy-Schwartz, and (I.19c) by the condition that  $\|z\| \leq 1$ .  $\square$

Using bounds (I.17) and (I.18), one can find the minimal number of components  $B$  required to limit the scenario approximation error to a specified level. As previously mentioned, one would need  $B+1$  scenarios in the reduced scenario set in order to enforce consistent expectations for the basis  $\{\mathbb{1}, u^1, \dots, u^B\}$ ; hence, these bounds can give a required number of scenarios to achieve a given level of accuracy within the CDS. The scenario approximation error could of course be larger outside the CDS, so this required number of scenarios is best interpreted as a *minimum* number of scenarios required to solve the problem with a given accuracy.

### 3.2 Scenario reduction program

This section lays out the computational details of the proposed scenario reduction method. Simply stated, it consists of computing the singular vectors of candidate recourse vectors (represented by data matrix  $M$ ) using a weighted version of singular value decomposition on matrices, and finding a new probability vector  $r$  that preserves expectations of those with the largest singular values. Preserving these expectations amounts to solving a set of linear equations, with non-negativity on  $r$ , and this can be achieved by linear programming techniques. Sparsity in  $r$  arises from the fact that the simplex algorithm will yield a basic solution, and that number of non-zero basic variables  $r_s$  will be small due to relatively few constraints.

#### Choosing candidate decisions

Before we can calculate the data matrix, we need to generate a set of solutions for the CDS. Following the discussion in Section 3.1, we propose to do this by solving the stochastic program with small candidate scenario sets *having more than one outcome* to get candidate decisions of reasonable quality.

The size of such candidate scenario sets is a trade-off since the more scenarios we use, the better solutions we will have, but the more computational effort is required to solve the problems. The number of candidate decisions to use is also a trade-off since the more candidate decisions we use, the more likely we are to obtain components of relevant recourse vectors, but the more computational effort is required to find all of them. Another factor to bear in mind is that the maximum size of scenario set we can generate using this method is limited by the number of decisions in the CDS. We should therefore use *at least* as many candidate decisions as the maximum size of scenario sets we want to find.

We use sampled scenario sets for our numerical tests to generate candidate decisions while, in principle, other kinds of candidate scenario sets could be used instead. Sampled scenario sets have the property that their expected distance to the optimal solution set (over many sampled sets) is monotonously decreasing in the size of the sampled set (Mak et al., 1999). In our numerical tests, we find that solving problems with rather few (3 or 5) sampled scenarios to generate the candidate solutions is sufficient for our method to work well.

### Weighted SVD on matrices

We now consider the calculation of singular values and vectors of  $M$  with respect to a weighted inner product. In the literature, this kind of extension is referred to as Generalised SVD (Van Loan, 1976), with details relevant for our purpose explained by Jolliffe (2002, Section 14.2.1). As implied by Theorem 2, before doing SVD, we subtract expectations column-wise from the data matrix  $M$  so that each column has zero expectation, and denote this by  $\hat{M}$ . To compute the weighted SVD on matrix  $\hat{M}$ , define the weight-scaled matrix

$$\check{M} = \text{Diag}(p)^{\frac{1}{2}} \hat{M} \quad (\text{I.20})$$

to account for weights in the inner product on  $W$ . Applying standard SVD to matrix  $\check{M}$  yields left singular vectors  $\check{u}^1, \dots, \check{u}^J$  and singular values  $\check{\sigma}_1, \dots, \check{\sigma}_J$ . These are also the singular values of  $\hat{M}$  with respect to weighted SVD, and its corresponding left-singular vectors are given by  $u^i = \text{Diag}(p)^{-\frac{1}{2}} \check{u}^i$ .

### Formulation

After computing the singular vectors, the next step of the method is to construct a new probability vector  $r$  which preserves the expectations of the first  $B$  of these (with the largest singular values). Finding such  $r$  can be done by solving the following linear program:

$$\min_r \quad a^T r \quad (\text{I.21a})$$

$$\text{s.t.} \quad \sum_{s=1}^S u_s^i r_s = \sum_{s=1}^S u_s^i p_s, \quad \forall i : \sigma_i \geq \sigma_B, \quad (\text{I.21b})$$

$$\sum_{s=1}^S r_s = 1, \quad r \geq 0, \quad (\text{I.21c})$$

where (I.21b) is the constraint to enforce expectations of basis components, and (I.21c) makes the reduced probabilities sum to one and be non-negative. The objective (I.21a) uses a randomly generated weight vector  $a$  of non-negative numbers between 0 and 1. Although an objective isn't really necessary as we only seek to find a vector satisfying the constraints, having a random objective is useful as it allows us to generate multiple different scenarios sets of the same size, which can be used for stability testing (Kaut & Wallace, 2007) or for simply generating multiple solutions.

Using the simplex algorithm to solve this problem will yield a basic solution for  $r$  which has at most  $B + 1$  non-zero elements. The reduced scenario set then consist of those outcomes with non-zero probabilities according to  $r$ . The value  $B$  should thus be

chosen according to how large a scenario set one wants to generate, and the scenario approximation error one is willing to tolerate. The overall scenario generation procedure is summarised in Algorithm 1.

---

**Algorithm 1:** Recourse Decomposition
 

---

**Data:** Observed outcomes  $\Xi$  with probabilities  $p$  having  $S$  elements, number of candidates  $K$ , candidate scenario set size  $S'$ , and truncation index  $B$

$\mathcal{U} \leftarrow K$  sampled scenario sets of size  $S'$  from  $\Xi$ ;

**for**  $k \leftarrow 1$  **to**  $K$  **do**

$\mathcal{C}[k] \leftarrow$  solution of stochastic program solved with scenario set  $\mathcal{U}[k]$  ;

**for**  $s \leftarrow 1$  **to**  $S$  **do**

$M[s, k] \leftarrow$  objective value of second-stage program with decision  $\mathcal{C}[k]$  and observed outcome  $\Xi[s]$ ;

**end**

**end**

$\check{M} \leftarrow \text{Diag}(p)^{1/2}(M - \mathbb{1}p^\top M)$  (mean subtracted and probability weighted matrix);

$\check{U}, \check{\sigma} \leftarrow$  matrix of left singular vectors and singular values of  $\check{M}$ , ordered high to low (from SVD);

$U \leftarrow \text{Diag}(p)^{-1/2}\check{U}$  (non-probability-weighted basis vectors);

$r \leftarrow$  solve the scenario reduction problem (I.21) with the  $B$  first column vectors of  $U$ ;

**Result:** Scenario set using outcomes and weights  $\{(\Xi[s], r_s) : r_s > 0\}$

---

### Scalability

The main parameters of the overall procedure are: the size of the observed distribution  $S$ , the number of candidate decisions  $K$ , the size of candidate scenario sets  $S'$ , and the number of included components  $B$ . Generally, we want  $S$  to be quite large so that the observed distribution accurately represents the underlying distribution, and that  $S'$  is small since we must be able to solve the problem  $K$  times using scenario sets of this size. Generally, we assume  $K \ll S$  is a sufficient number of candidate decisions due to degeneracy in problem structure. It is required that  $B \leq K$  since we cannot get more components than candidates. Lastly, scenario generation is done in the hopes that  $B \ll S$ , meaning the size of the distribution can be greatly reduced to be able to solve the problem. Obtaining the data matrix  $M$  requires solving  $K$  versions of the problem with size  $S'$  scenario sets, and evaluating them on the observed distribution with a total of  $K \times S$  evaluations of the recourse function. Weighted SVD must be done on a  $S \times K$  matrix, and once the  $B$  components are chosen, a LP of  $S$  variables and  $B + 1$  constraints is solved. Run-times for each of these steps were reported in Section 4.5. Realistically, the LP can be solved with  $S$  in the millions, while SVD on matrices (to get singular values and left-singular vectors of  $\check{M}$ ) has run-time complexity in the order of  $O(S^2K + K^3)$  (Golub & Van Loan, 2013) and solves within reasonable time for relevant values of  $S$  and  $K$  (see Table I.3).

## 4 Computational results

This section presents computational experiments related to our proposed scenario reduction method, which we refer to as *Recourse Decomposition* (RD). To test our method we use four stochastic two-stage problems that represent relevant applications and different

challenges in scenario generation. Results are reported in §4.5 but, first, the setup is explained.

## 4.1 Case study problems

The four selected case study problems reflect various applications of stochastic programming and have been selected to be unstable (see §4.3), to have difficult distributions, and to represent computationally challenging problems. These problems are referred to as

- Telecommunications Network Planning (TNP), from Sen et al. (1994)
- Multidimensional News-vendor with Substitution (MNV), from Vaagen et al. (2011)
- Aircraft Operations and Scheduling (AOS), from Midler and Wollmer (1969)
- Storage Layout and Routing (SLR)

where all but MNV minimise the objective. Each problem has its own characteristics, and a high-level overview is given in Table I.1. TNP has a high dimensional distribution and is highly unstable (also confirmed by Linderoth et al., 2006), MNV has a multi-modal distribution with complicated dependence patterns, AOS is chosen for instability while SLR represents a very computationally complex integer program with a binary distribution. The SLR problem could only be solved by exact methods for very small instances, while the others were reasonably tractable for the sake of conducting computational studies within a reasonable time. Further details about problem applications, mathematical formulations, distributions and problem instance generation are provided in the Supplemental Material. The observed distribution was made from sampling  $S = 5000$  outcomes from a parametric distribution defined for each problem. For SLR, there are maximally  $S = 256$  outcomes (by full enumeration of its binary distribution) whose probabilities are determined by 5000 samples.

**Table I.1:** Characteristics of case study problems.

Problem	Marginals	Dependence	Distr. dimension	Formulation
TNP	Gamma	Independence	82	LP
MNV	Gaussian mixture	Gaussian copula	10	MILP
AOS	Gamma	Gaussian copula	34	LP
SLR	Bernoulli	Gaussian copula	8	IP

## 4.2 Scenario generation benchmark methods

We compare Recourse Decomposition against Monte Carlo sampling and Minimum Transportation Distance (MTD) methods since these are the most versatile distribution-based methods to apply to various kinds of distributions. Our own Recourse Clustering (Narum, 2020) is also compared as an alternative problem-based method.

Monte Carlo sampling is the simplest implementation of scenario generation where scenario sets are sampled from the observed distribution. Shapiro et al. (2014, Chapter 5) explain more detailed properties of this as an approach to scenario generation. Given that we sample from a discrete distribution in our experiments, we draw samples with replacements until we have a specified number of unique outcomes.

Minimum Transportation Distance methods (Dupačová et al., 2003; Pflug, 2001) are based on minimising the transportation distance between the observed distribution and the scenario set based on some specified distance metric. This amounts to a partitioning problem that can be solved by centroid-based clustering methods (Rujeerapaiboon et al., 2022). We use k-medoids clustering (referred to as MTDMedoids) to get a scenario set limited to the outcomes in the observed distribution, and k-means clustering (referred to as MTDMeans) to get a scenario set not necessarily among the original outcomes. MTDMeans is omitted in the SLR problem due to its discrete distribution support. We use the squared Euclidean distance as a distance metric between outcomes. Multiple scenario sets of each size are obtained from random initialisation in the clustering heuristics.

To obtain scenario sets with Recourse Decomposition, we must specify how candidate decisions are found. This specification is determined by the *number of candidate decisions* ( $K$ ) in the CDS, and the scenario set size ( $S'$ ) used to find each candidate as described in Section 3.2. Generally, the larger the value of  $S'$  the higher the quality of the resulting candidate decisions. We refer to a CDS having fewer candidates or of lower quality decisions (or both) as *simpler*, and to a CDS having more candidates or of higher quality decisions (or both) as *richer*. For our experiments, candidates are found by solving the problem with sampled scenario sets of size  $S' = 3$  or  $5$ , repeated  $K = 100$  or  $200$  times. Each solution is then evaluated using the  $S = 5000$  observed outcomes to obtain the data matrix (of size  $5000 \times 100\text{--}200$ ). The richer specification is used for the especially unstable problem TNP.

Lastly, we compare to the method referred to as Recourse Clustering by Narum (2020). This uses the same data matrix  $M$  but instead of decomposition, it does k-medoids clustering on the rows of  $M$  using the  $L_1$ -norm as a distance metric. This method is similar to other clustering-based methods that utilise problem-specific properties (described in the literature review) but is simpler to implement for direct comparison.<sup>1</sup>

### 4.3 Empirical evaluation of scenario generation

To compare scenario generation methods, we use empirical evaluation metrics (see Kaut & Wallace, 2007, for details). *Stability* of a scenario generation method refers to whether we get similar results by applying the same scenario generation method multiple times using the same specification. In our case, we get different results due to randomness in the scenario generation methods themselves. *Quality* of a scenario generation method refers to the quality of decisions it is able to provide when these are evaluated using the underlying full distribution. Such evaluation often relies on large sampling estimates, and we use the observed distribution directly for this purpose.

Empirical evaluation of scenario generation requires generating multiple scenario sets by the same specification to derive measures of quality and stability. Let  $\mathcal{M}$  denote a scenario generation method and  $N$  the size of a generated scenario set. We repeatedly generate  $L = 20$  scenario sets of size  $N$  using method  $\mathcal{M}$  and apply these to find a set of solutions  $\mathcal{X}(N \mid \mathcal{M}, L)$  to the stochastic program. Any solution,  $x$ , can be evaluated *out-of-sample* as

$$F_p(x) := f(x) + \mathbb{E}_p [Q(x, \xi)], \quad (\text{I.22})$$

where  $p$  denotes the observed distribution. Correspondingly,  $F_r(x)$  denotes *in-sample* evaluation where  $r$  represents a scenario set (even if some scenario sets cannot be expressed

<sup>1</sup>The other mentioned contenders (Bertsimas & Mundru, 2022; Hewitt et al., 2021; Keutchayan et al., 2023) require evaluation over all 5000 single scenario solutions, and either require unavailable custom clustering heuristics or solve it exactly by MIPs which would not finish within reasonable time using  $S = 5000$  outcomes in the observed distribution.

this way). As a *measure of quality* we use:

$$\text{Quality}(N | \mathcal{M}, L) = \text{Median}_{x \in \mathcal{X}(N | \mathcal{M}, L)} \{F_p(x)\}. \quad (\text{I.23})$$

Namely, we summarise quality of the solutions  $\mathcal{X}(N | \mathcal{M}, L)$  by the median out-of-sample objective evaluation. Ideally, quality should be measured as the best out-of-sample objective evaluation but sampling inference on extrema quantities can be sensitive, thus, we use the median. As a *measure of stability* we use:

$$\text{Stability}(N | \mathcal{M}, L) = \sqrt{\text{Var}_{x \in \mathcal{X}(N | \mathcal{M}, L)} \{F_p(x)\} + \text{Avg}_{x \in \mathcal{X}(N | \mathcal{M}, L)} \left\{ (F_p(x) - F_r(x))^2 \right\}}. \quad (\text{I.24})$$

This stability measure combines the variance of out-of-sample evaluations and the average of the discrepancy (bias) between in-sample and out-of-sample evaluations. A low value of the former implies we get similar results each time, while a low value of the latter implies the in-sample evaluation is also a good approximation to the out-of-sample evaluation. By using the squared average bias, these two quantities are comparable. In practice, such a stability measure is often used as a stopping criterion to choose the scenario set size  $N$ .

Given these performance measures, we may now infer a *required number of scenarios* to fulfil a given level of performance. Assume we prefer low values of both measures  $\text{Quality}(N | \mathcal{M}, L)$  and  $\text{Stability}(N | \mathcal{M}, L)$ , and let  $\nu$  denote a threshold for either measure. Generally, the threshold  $\nu$  is chosen according to what constitutes sufficient performance for the given problem at hand. Then, the required number of scenarios using method  $\mathcal{M}$ , threshold  $\nu$ , and performance measure  $\mu \in \{\text{Quality}, \text{Stability}\}$ , is found as

$$N_\mu^*(\mathcal{M} | \nu) = \min\{N : \mu(N | \mathcal{M}, L) \leq \nu\}, \quad (\text{I.25})$$

where smaller values of  $N_\mu^*$  are preferable since stochastic programs using fewer scenarios are usually computationally easier to solve.

Scenario generation methods are benchmarked against each other by comparing each of their required number of scenarios when using the same performance threshold. Namely, scenario generation method  $\mathcal{M}_1$  is considered to be better than method  $\mathcal{M}_2$  whenever

$$N_\mu^*(\mathcal{M}_1 | \nu) < N_\mu^*(\mathcal{M}_2 | \nu),$$

using either measures of quality or stability. To quantify the improvement of  $\mathcal{M}_1$  over  $\mathcal{M}_2$ , we may find the required number of scenarios from method  $\mathcal{M}_1$  relative to what would have been required from method  $\mathcal{M}_2$ :

$$\frac{N_\mu^*(\mathcal{M}_1 | \nu)}{N_\mu^*(\mathcal{M}_2 | \nu)} \cdot 100\%. \quad (\text{I.26})$$

Here, a value below 100% means method  $\mathcal{M}_1$  requires less scenarios than method  $\mathcal{M}_2$  to reach the same performance threshold  $\nu$ .

In practice, these benchmarks are performed on a pre-selected range  $N \in \mathcal{N}$  and the performance threshold is set according to the best performance of the worst method within this range:

$$\nu \leftarrow \max \left\{ \min_{N \in \mathcal{N}} \mu(N | \mathcal{M}_1, L), \min_{N \in \mathcal{N}} \mu(N | \mathcal{M}_2, L) \right\}, \quad (\text{I.27})$$

where  $\mu$  denotes either performance measure. This ensures both methods  $\mathcal{M}_1$  and  $\mathcal{M}_2$  reach the threshold  $\nu$  within the range  $N \in \mathcal{N}$  so that they can be compared.

## 4.4 Experimental setup

The case study problems have been implemented in the Julia programming language (Bezanson et al., 2017) using the JuMP (Dunning et al., 2017) modelling package. Their linear programming (LP) and mixed-integer programming (MIP) formulations are solved using Gurobi v9.0 (Gurobi Optimization, LLC, 2022) with default settings unless stated otherwise. Computations are done on a computational cluster on nodes with 20 processing cores ( $2 \times$  Intel(R) Xeon(R) Gold 5115 CPU @ 2.40GHz) and 96 GB of RAM. For clustering heuristics, we use the `Clustering.jl` package in Julia.

## 4.5 Results

To illustrate the effectiveness of Recourse Decomposition, we first highlight its improved quality and stability properties compared to the other methods. We then explore the concept of degeneracy in problem structure by analysing singular values and projection distances, and report run-times for each step of the proposed scenario generation procedure. Lastly, we study the sensitivity of Recourse Decomposition on the specification of the CDS.

### Quality and stability

We compare the required number of scenarios to achieve similar quality and stability across methods using the measures described in §4.3. Convergence of the quality and stability measure for increasing scenario set size ( $N$ ) is shown in Figures I.1 and I.2, respectively, across methods and problems. Both figures show that all scenario generation methods tend to improve with increasing scenario set size, as expected.

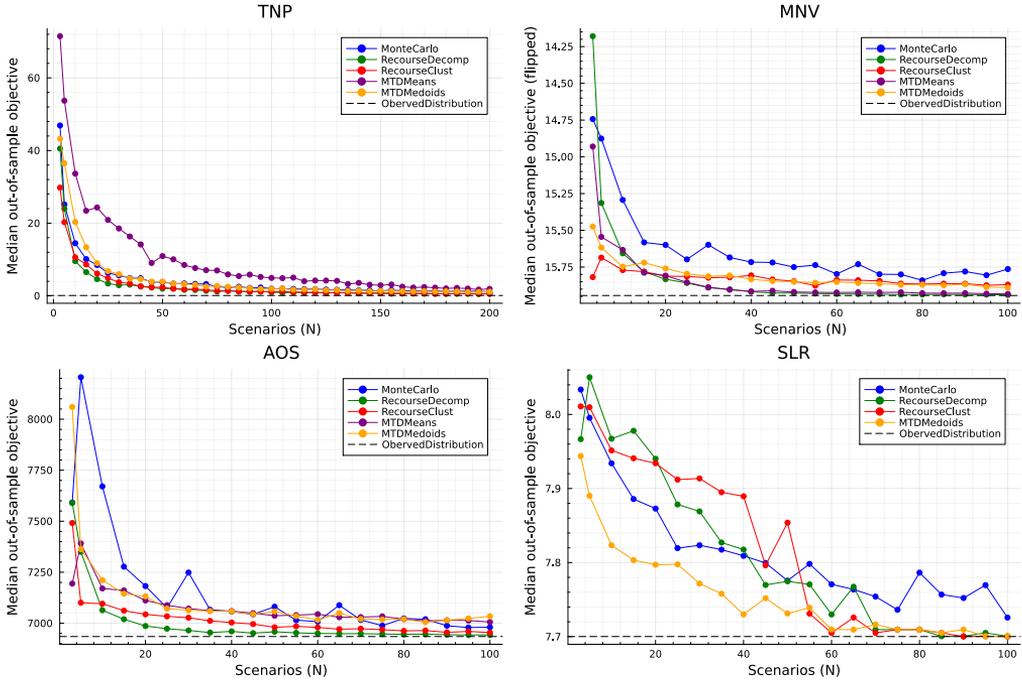
Furthermore, to more precisely quantify the improved convergence of Recourse Decomposition over other methods, we calculate whether smaller scenario sets can be used by Recourse Decomposition to obtain the same level of performance (quality or stability) as the other methods. Namely, in the notation of §4.3, we quantify the relative requirement for scenarios,

$$\frac{N_{\mu}^*(\mathcal{M}_{\text{RD}} \mid \nu)}{N_{\mu}^*(\mathcal{M} \mid \nu)} \cdot 100\%, \quad (\text{I.28})$$

where  $\mathcal{M}$  represents alternative methods and  $\mathcal{M}_{\text{RD}}$  represents Recourse Decomposition. The performance threshold  $\nu$  is determined by the best performance of the worst method (by the formula in Eq. I.27). Table I.2 reports RD’s relative requirement for scenarios to obtain the same level of quality or stability as each of the other methods.

We see from Figures I.1 and I.2 that Recourse Decomposition is consistently better than, or on par with, the other methods both on quality and stability, with the exception of SLR that we comment on below. While other methods sometimes start out better than RD for very small sizes (see MNV and AOS), we observe in Table I.2 that as they all eventually converge, RD consistently converges faster by reaching the same merits at smaller scenario set sizes. Compared to the distribution-based methods, RD obtains the same quality in the range of 20–90% as many scenarios and the same stability in the range of 10–74% as many scenarios.

The other problem-based method, Recourse Clustering (RC), is mostly on par with Recourse Decomposition. Interestingly, it seems RC performs a slightly better for low ranges of scenario set sizes for AOS and MNV, but is overtaken by slower convergence. In fact, RD achieves good quality results (Table I.2) using much fewer scenarios than Recourse Clustering on these two problems. It is reasonable to assume these problem-based



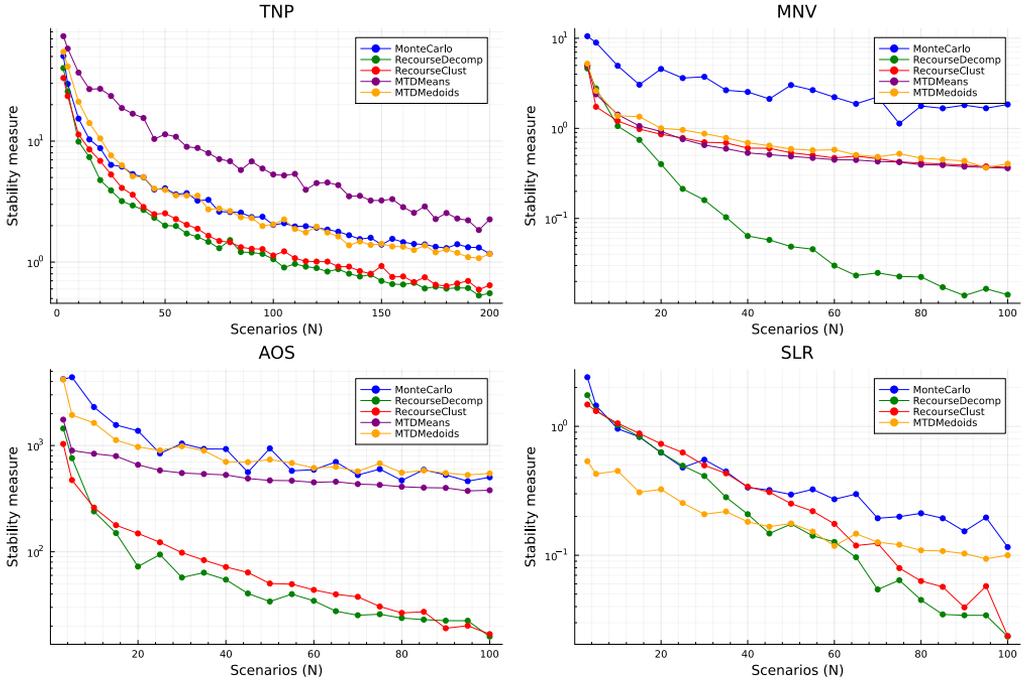
**Figure I.1:** Quality measure of scenario generation methods as a function of scenario set size ( $N$ ), across problems. MNV (maximisation) has a flipped second axis for comparability. Observed Distribution denotes the optimal solution to the full problem.

methods perform similarly since they use the same information about the problem, but we conclude RD more consistently performs well based on these results. The additional great advantage of RD is its accompanying analysis of singular values, error bounds, and detection of degeneracy in problem structure that can inform the modeller about properties of the specific problem at hand.

For the SLR problem, the MTD methods start out best and keeps being better until surpassed by RD in both quality and stability for higher ranges of scenarios set sizes by faster convergence (see Table I.2). Based on the nature of the specific problem (picking scenarios based on which products to collect simultaneously in a warehouse) clustering outcomes seems appropriate, but the problem instance is small ( $3 \times 3$  grid) and we suspect this problem could be more unstable for larger instances and require many more scenarios. Larger instances of this problem were intractable to solve within reasonable time using exact methods. We also see that RD is ultimately much more stable than the MTD methods (Table I.2 and Figure I.2), which is a great advantage when choosing scenario set sizes in practice by the criterion of sufficient stability. We think further exploration of problem-based scenario generation for this particular problem is an interesting avenue for further research, also using larger and more realistic instance sizes that require better solution procedures.

## Degeneracy in problem structure

We now investigate degeneracy of problem structure by which we mean the extent to which recourse vectors can be represented by few components. Singular values bound the



**Figure I.2:** Stability measure (lower is better) of scenario generation methods as a function of scenario set size ( $N$ ), across problems. Note that the second axis has a logarithmic scale.

approximation error *within* the CDS (Theorem 2), while the projection distance bounds it *outside* the CDS (Theorem 1). In practice, only the former is available; however, for this paper we have performed more extensive experiments (on reasonably tractable problems) for the sake of argument to compare projection distance to singular values, which infers generalisability.

In Figure I.3 we illustrate the singular values derived from a given large CDS. We also show the projection distance from an optimal recourse vector (found by solving the full problem) onto the space spanned by the corresponding left-singular vectors up to the same singular value. That is, if  $Q^{x^*}$  denotes the recourse vector of an optimal solution  $x^*$ , we compare the projection distance

$$\left\| (I - P_B)Q^{x^*} \right\|_W, \quad (\text{I.29})$$

to the singular value  $\sigma_B$  for increasing values of  $B$ . According to Theorem 1, this projection distance is proportional to the approximation error when using the components  $\{1, u^1, \dots, u^B\}$  to make a scenario set. For comparability, values are scaled by the order of magnitude of the objective values of each problem so that values in Figure I.3 can be interpreted as fractions of relevant objective values. The total number of candidates used for each problem is given by the maximum value on the first axis of the corresponding plot.

We see in Figure I.3 that the singular values decay rapidly in the beginning, which means there is degeneracy in problem structure within the CDS. The corresponding quick decay in the projection distance validates this degeneracy generalises to outside the CDS. Ideally, the projection distance would remain lower than singular values in all components,

**Table I.2:** Recourse Decomposition’s relative requirement for scenarios to reach the same level of quality and stability as other methods. See the formula in (I.28). Values below 100% means RD improves on the other method (highlighted in green). “—” means the method is not applied to the problem.

Quality	TNP	MNV	AOS	SLR
MonteCarlo	50.0%	31.2%	26.3%	70.0%
MTDMedoids	55.3%	30.0%	23.5%	89.5%
MTDMeans	30.8%	60.0%	20.0%	—
RecourseClust	100.0%	54.5%	35.0%	94.4%
Stability	TNP	MNV	AOS	SLR
MonteCarlo	47.5%	13.3%	10.5%	65.0%
MTDMedoids	51.3%	26.3%	10.5%	73.7%
MTDMeans	30.8%	25.0%	10.5%	—
RecourseClust	100.0%	25.0%	100.0%	100.0%

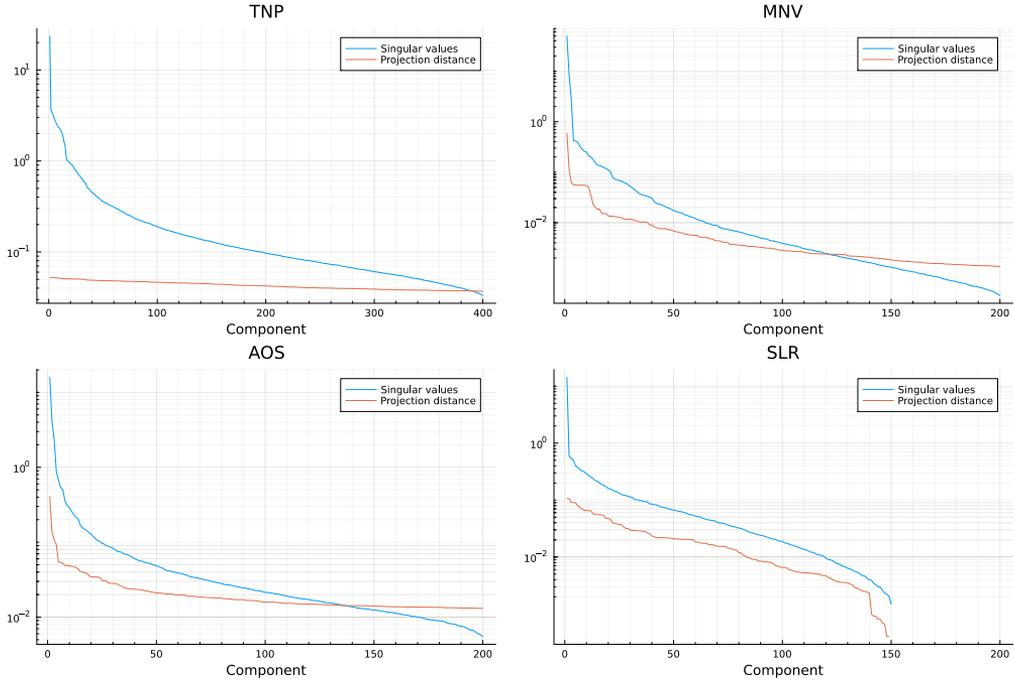
meaning the approximation error on new decisions (usually unavailable ex ante) is lower than the within-CDS error estimate (which is available). We see the projection distance surpasses the singular values for some of the latter components in three of the problems, which has the interpretation that candidate recourse vectors do not compose other relevant recourse vector completely. It is fair to assume this would be the case, and this is a potential weak-point of using candidate decisions for problem-based scenario generation. The important take-away is that their majority part is still covered, meaning we can obtain effective but not necessarily perfect approximations. Using more and higher-quality candidate decisions may also improve these approximations.

What is especially interesting about analysing singular values to determine degeneracy (and infer a minimum required number of scenarios) is that it only requires solving the problem using small scenario sets (in order to generate the candidate decisions). This can directly motivate the need to develop algorithms for a particular stochastic programming formulation (like a decomposition procedure or a heuristic), to be able to solve particular instances to sufficient precision. Validation of the final solution is always advised since conclusions made from candidate decisions are mainly valid within the CDS or if the true optimal decision(s) are very close to the CDS.

### Computational run-times

We report run-times for the computational steps of Recourse Decomposition in Table I.3. These steps consist of: (i) obtaining the data matrix  $M$  (solve to get candidate decisions, and evaluate these over the observed distribution), (ii) weighted SVD on  $M$ , (iii) solving the scenario reduction program (I.21), and (iv) solving the problem using the RD scenario set. To be conservative, we let steps (iii–iv) use the largest scenario set size  $N$  (as reported in Figure I.1) and report their average run-time over  $L = 20$  runs. All run-times are found using single-threading (for comparability), and the reported total run-time assumes steps (i–iv) are done in sequence. We also report the time to solve the full problem using the observed distribution instead of a scenario set. Lastly, we note that all comparing scenario generation methods spent less than one second to generate a scenario set, which is comparable to the scenario reduction program used for RD.

We observe from Table I.3 that Recourse Decomposition overall requires less computational time than the full problem. Consider also that step (i) where we obtain data matrix



**Figure I.3:** Singular values and projection distance to the optimal recourse vector, considering an incremental increase in the used components  $\mathcal{B}$ . See the formula in (I.29). Values are scaled by the order of magnitude of the respective case study problems (TNP: 30, MNV: 15, AOS: 7500, SLR: 8) for comparability. Zero values are omitted where it applies.

$M$  is highly relevant for parallelisation (composed of  $K + KS$  separate sub-tasks), which is also where the majority time is spent for most of the problems. The larger run-time comparison for SLR may be due to its lower number of observed outcomes ( $S = 256$ ) which makes the full problem smaller.

The primary advantage of generating parsimonious scenario sets comes from avoiding above-linear scaling of solution-time in the size of the problem (whose size roughly scales proportional to the number of scenarios), as well as memory limitations. We have intentionally kept instances of the respective problems small enough that the full problem could be solved, but with moderately larger instances for AOS and SLR, we quickly run into memory issues or un-reasonable solve times (more than a week). Keep in mind that larger such instances can still be solved using our proposed scenario generation method to get effective solutions.

### Sensitivity on candidate decision sets

We want to examine how sensitive the results of Recurse Decomposition are to the specification of the candidate decision set (CDS) by benchmarking against itself when using different specifications of the CDS. Namely, we construct many versions of the CDS having incrementally richer specifications of  $S'$  and  $K$  to make RD scenario sets. Intuitively, these represent different methods  $\mathcal{M}_{RD}(S', K)$  specified by the richness of their CDS. These are then compared in terms of the quality measure (I.23), where the simplest CDS is used as the baseline for comparison. In the notation of §4.3, we find the

**Table I.3:** Computational run-times (in seconds) for Recourse Decomposition (RD) and the full problem using single-threading. The total run-time for RD assumes all steps are done in sequence, and the full problem use the observed distribution instead of a scenario set. The right-most comparison reports the fraction of total run-time using RD compared to the full problem.

	Candidate dec.		Recourse eval.		WSVD	Scenario reduction	Solve	Total		Comp.
	Avg.	Tot.	Avg.	Tot.				RD	Full	
TNP	0.0927	18.54	0.0006	599.73	0.0812	0.8029	202.96	8.22e+2	4.90e+4	1.68%
MNV	0.0111	1.11	0.0001	41.35	0.0327	0.3415	0.30	4.31e+1	7.50e+2	5.75%
AOS	0.0947	9.47	0.0099	4949.82	0.0348	0.3542	66.30	5.03e+3	5.13e+5	0.98%
SLR	0.1001	10.01	0.0007	17.10	0.0105	0.1906	204.86	2.32e+2	8.79e+2	26.40%

relative requirement for scenarios

$$\frac{N_{\text{Quality}}^*(\mathcal{M}_{\text{RD}}(K, S') \mid \nu)}{N_{\text{Quality}}^*(\mathcal{M}_{\text{RD}}(K_{\min}, S'_{\min}) \mid \nu)} \cdot 100\%, \quad (\text{I.30})$$

for each specification  $K$  and  $S'$  of the CDS, where  $K_{\min}$  and  $S'_{\min}$  represent the simplest baseline specification. The threshold  $\nu$  is again specified by the formula (I.27). Table I.4 shows (I.30) for ranges of  $S'$  and  $K$ .

**Table I.4:** Relative requirement for scenarios using different specifications of the CDS when compared to a baseline specification. See the formula in (I.30). Vertically, we have the sample size of the scenario sets used to find candidate decisions ( $S'$ ) and, horizontally, we have the number of such candidates found ( $K$ ). The simplest CDS (highlighted in blue) is the baseline for comparison and gives no improvement on itself by definition. Values below 100% signify improvement (highlighted in green) and values above 100% signify a worsening (highlighted in red).

TNP	$K = 200$	$K = 250$	$K = 300$	$K = 350$	$K = 400$	MNV	$K = 100$	$K = 125$	$K = 150$	$K = 175$	$K = 200$
$S' = 3$	100.0 %	92.5 %	92.5 %	92.5 %	100.0 %	$S' = 3$	100.0 %	105.3 %	100.0 %	105.3 %	105.3 %
$S' = 6$	95.0 %	95.0 %	87.5 %	97.5 %	100.0 %	$S' = 6$	84.2 %	105.3 %	100.0 %	105.3 %	94.7 %
$S' = 9$	100.0 %	92.5 %	85.0 %	90.0 %	95.0 %	$S' = 9$	105.3 %	100.0 %	94.7 %	105.3 %	100.0 %
$S' = 12$	95.0 %	100.0 %	90.0 %	90.0 %	100.0 %	$S' = 12$	105.3 %	100.0 %	100.0 %	100.0 %	94.7 %
$S' = 15$	97.5 %	92.5 %	100.0 %	95.0 %	95.0 %	$S' = 15$	100.0 %	105.3 %	89.5 %	100.0 %	105.3 %
AOS	$K = 100$	$K = 125$	$K = 150$	$K = 175$	$K = 200$	SLR	$K = 100$	$K = 125$	$K = 150$	$K = 175$	$K = 200$
$S' = 3$	100.0 %	100.0 %	100.0 %	105.3 %	105.3 %	$S' = 3$	100.0 %	82.4 %	100.0 %	64.7 %	105.9 %
$S' = 6$	105.3 %	94.7 %	105.3 %	89.5 %	105.3 %	$S' = 6$	100.0 %	76.5 %	76.5 %	70.6 %	82.4 %
$S' = 9$	105.3 %	94.7 %	89.5 %	89.5 %	100.0 %	$S' = 9$	82.4 %	70.6 %	82.4 %	70.6 %	70.6 %
$S' = 12$	100.0 %	100.0 %	78.9 %	105.3 %	105.3 %	$S' = 12$	76.5 %	70.6 %	82.4 %	64.7 %	82.4 %
$S' = 15$	105.3 %	105.3 %	105.3 %	111.1 %	100.0 %	$S' = 15$	82.4 %	82.4 %	76.5 %	64.7 %	70.6 %

Primarily, we conclude from Table I.4 that sensitivity on the specification of the CDS is low. One exception is the SLR problem where it seems candidate scenario sets larger than  $S' = 6$  gives better results, while sensitivity on the number of such candidates ( $K$ ) still seems to be low. Note that the number of candidate solutions  $K$  cannot be lower than the largest possible scenario set size within the range of scenario set sizes  $N \in \mathcal{N}$  used in these tests.

## 5 Discussion

We have argued that the proposed problem-based scenario generation exploits degeneracy in problem structure, and that a few components can be found that compose the majority part of all attainable output distributions. For the purpose of scenario generation on any kind of recourse function this, in practice, is an *empirical claim*. Degeneracy must be validated on a given problem by analysing singular values on a candidate set of recourse vectors. We also suspect some problems have this property especially pronounced; namely, ones where the stochastic parameters interact in specific ways within the recourse program such that we get an overall low-dimensional span of output-distributions across relevant candidate decisions. In other words, problems with much structure. This is an observation based on the kinds of models commonly used in application.

Our proposed scenario reduction method is a natural and very lightweight extension to the typical way a modeller may approach a new stochastic programming problem. Once a formulation has been made, it should be tested using some reasonably sized scenario sets to check its computational tractability and stability, where solutions are evaluated out-of-sample. If this reveals there is sufficient stability, we would be done. Otherwise, further investigations are needed, and our problem-based scenario generation procedure can very naturally be applied using the computations already performed. Thus, obtaining more effective scenario sets from that point on requires almost negligible additional computations.

Since the approach can be applied when probabilities in the observed distribution are not equiprobable, we may also apply distribution-based scenario generation to obtain the observed distribution itself. By using a smaller and better constructed observed distribution, we may relieve some of the computations required to obtain the data matrix while giving more reliable estimates of the expected recourse.

## Conclusion

This paper provides a scaleable, effective and easily implementable approach to exploit problem structure for the purpose of scenario generation. We do this by directly studying the output distribution of the problem, which means the procedure is agnostic to the problem and the kind of input distribution. We show it is possible to find a few components that compose the majority part of a representative collection of output distributions, and that constructing a scenario set using such components gives better and more reliable results when used to solve the stochastic program. The provided bounds on the resulting approximation error can be used to approximately infer the required size of the scenario set for a given precision and, importantly, this only relies on solving the stochastic program using small scenario sets.

## References

- Arpón, S., Homem-de-Mello, T., & Pagnoncelli, B. (2018). Scenario reduction for stochastic programs with Conditional Value-at-Risk. *Mathematical Programming*, 170(1), 327–356. <https://doi.org/10.1007/s10107-018-1298-9>
- Ball, M. O., Colbourn, C. J., & Provan, J. S. (1995, January). Chapter 11 Network reliability. In *Handbooks in Operations Research and Management Science* (pp. 673–762, Vol. 7). Elsevier. [https://doi.org/10.1016/S0927-0507\(05\)80128-8](https://doi.org/10.1016/S0927-0507(05)80128-8)

- Bayraksan, G., & Love, D. K. (2015, September). Data-Driven Stochastic Programming Using Phi-Divergences. In D. Aleman, A. Thiele, J. C. Smith & H. J. Greenberg (Eds.), *The Operations Research Revolution* (pp. 1–19). INFORMS. <https://doi.org/10.1287/educ.2015.0134>
- Bent, R. W., & Van Hentenryck, P. (2004). Scenario-Based Planning for Partially Dynamic Vehicle Routing with Stochastic Customers. *Operations Research*, *52*(6), 977–987. <https://doi.org/10.1287/opre.1040.0124>
- Bertsimas, D., & Mundru, N. (2022). Optimization-Based Scenario Reduction for Data-Driven Two-Stage Stochastic Optimization. *Operations Research*. <https://doi.org/10.1287/opre.2022.2265>
- Bezanson, J., Edelman, A., Karpinski, S., & Shah, V. B. (2017). Julia: A Fresh Approach to Numerical Computing. *SIAM Review*, *59*(1), 65–98. <https://doi.org/10.1137/141000671>
- Birge, J. R., & Louveaux, F. (2011). *Introduction to stochastic programming* (2nd ed.). Springer New York, NY. <https://doi.org/10.1007/978-1-4614-0237-4>
- Caunhye, A. M., & Nie, X. (2018). A Stochastic Programming Model for Casualty Response Planning During Catastrophic Health Events. *Transportation Science*, *52*(2), 437–453. <https://doi.org/10.1287/trsc.2017.0777>
- Dunning, I., Huchette, J., & Lubin, M. (2017). JuMP: A Modeling Language for Mathematical Optimization. *SIAM Review*, *59*(2), 295–320. <https://doi.org/10.1137/15M1020575>
- Dupačová, J., Gröwe-Kuska, N., & Römisch, W. (2003). Scenario reduction in stochastic programming. *Mathematical Programming*, *95*(3), 493–511. <https://doi.org/10.1007/s10107-002-0331-0>
- Emmerling, J., & Tavoni, M. (2018). Climate Engineering and Abatement: A ‘flat’ Relationship Under Uncertainty. *Environmental and Resource Economics*, *69*(2), 395–415. <https://doi.org/10.1007/s10640-016-0104-5>
- Fairbrother, J., Turner, A., & Wallace, S. W. (2022). Problem-driven scenario generation: An analytical approach for stochastic programs with tail risk measure. *Mathematical Programming*, *191*(1), 141–182. <https://doi.org/10.1007/s10107-019-01451-7>
- Farahani, R., Rezapour, S., & Kardar, L. (Eds.). (2011, June). *Logistics Operations and Management: Concepts and Models* (1st edition). Elsevier.
- Friedberg, S. H., Insel, A. J., & Spence, L. E. (2002, November). *Linear Algebra, 4th Edition* (4th edition). Pearson.
- Golub, G. H., & Van Loan, C. F. (2013). *Matrix computations* (Fourth edition). The Johns Hopkins University Press. OCLC: ocn824733531.
- Guo, Z., Wallace, S. W., & Kaut, M. (2019). Vehicle Routing with Space- and Time-Correlated Stochastic Travel Times: Evaluating the Objective Function. *INFORMS Journal on Computing*, *31*(4), 654–670. <https://doi.org/10.1287/ijoc.2019.0906>
- Gurobi Optimization, LLC. (2022). Gurobi optimizer reference manual. <https://www.gurobi.com>
- Henrion, R., & Römisch, W. (2022). Problem-based optimal scenario generation and reduction in stochastic programming. *Mathematical Programming*, *191*(1), 183–205. <https://doi.org/10.1007/s10107-018-1337-6>
- Hewitt, M., Ortmann, J., & Rei, W. (2021). Decision-based scenario clustering for decision-making under uncertainty. *Annals of Operations Research*. <https://doi.org/10.1007/s10479-020-03843-x>
- Jolliffe, I. T. (2002, October). *Principal Component Analysis* (2nd edition). Springer.
- Kall, P., & Wallace, S. W. (1994). *Stochastic Programming*. John Wiley & Sons.
- Kaut, M., & Wallace, S. W. (2007). Evaluation of scenario-generation methods for stochastic programming. *Pacific Journal of Optimization*, *3*(2), 257–271.
- Keutchan, J., Ortmann, J., & Rei, W. (2023). Problem-driven scenario clustering in stochastic optimization. *Computational Management Science*, *20*(1), 13. <https://doi.org/10.1007/s10287-023-00446-2>
- King, A. J., & Wallace, S. W. (2012, June). *Modeling with Stochastic Programming*. Springer New York, NY. <https://doi.org/10.1007/978-0-387-87817-1>

- Linderoth, J., Shapiro, A., & Wright, S. (2006). The empirical behavior of sampling methods for stochastic programming. *Annals of Operations Research*, *142*(1), 215–241. <https://doi.org/10.1007/s10479-006-6169-8>
- Lulli, G., & Sen, S. (2004). A Branch-and-Price Algorithm for Multistage Stochastic Integer Programming with Application to Stochastic Batch-Sizing Problems. *Management Science*, *50*(6), 786–796. <https://doi.org/10.1287/mnsc.1030.0164>
- Mak, W.-K., Morton, D. P., & Wood, R. (1999). Monte Carlo bounding techniques for determining solution quality in stochastic programs. *Operations Research Letters*, *24*(1-2), 47–56. [https://doi.org/10.1016/S0167-6377\(98\)00054-6](https://doi.org/10.1016/S0167-6377(98)00054-6)
- Midler, J. L., & Wollmer, R. D. (1969). Stochastic programming models for scheduling airlift operations. *Naval Research Logistics Quarterly*, *16*(3), 315–330.
- Narum, B. (2020). *Problem-based scenario generation with binary distributions: Case Study in Air Traffic Flow Management* [Master's thesis, Norwegian University of Science and Technology]. <https://hdl.handle.net/11250/2777017>
- Ni, J., Chu, L. K., & Li, Q. (2017). Capacity decisions with debt financing: The effects of agency problem. *European Journal of Operational Research*, *261*(3), 1158–1169. <https://doi.org/10.1016/j.ejor.2017.02.042>
- Parpas, P., Ustun, B., Webster, M., & Tran, Q. K. (2015). Importance Sampling in Stochastic Programming: A Markov Chain Monte Carlo Approach. *INFORMS Journal on Computing*, *27*(2), 358–377. <https://doi.org/10.1287/ijoc.2014.0630>
- Pflug, G. (2001). Scenario tree generation for multiperiod financial optimization by optimal discretization. *Mathematical Programming*, *89*(2), 251–271. <https://doi.org/10.1007/PL00011398>
- Prochazka, V., & Wallace, S. W. (2018). Stochastic programs with binary distributions: Structural properties of scenario trees and algorithms. *Computational Management Science*, *15*(3), 397–410. <https://doi.org/10.1007/s10287-018-0312-2>
- Prochazka, V., & Wallace, S. W. (2020). Scenario tree construction driven by heuristic solutions of the optimization problem. *Computational Management Science*, *17*(2), 277–307. <https://doi.org/10.1007/s10287-020-00369-2>
- Rujeerapaiboon, N., Schindler, K., Kuhn, D., & Wiesemann, W. (2022). Scenario reduction revisited: Fundamental limits and guarantees. *Mathematical Programming*, *191*(1), 207–242. <https://doi.org/10.1007/s10107-018-1269-1>
- Schultz, R. (2000). Some Aspects of Stability in Stochastic Programming. *Annals of Operations Research*, *100*(1), 55–84. <https://doi.org/10.1023/A:1019258932012>
- Sen, S., Doverspike, R. D., & Cosares, S. (1994). Network planning with random demand. *Telecommunication Systems*, *3*(1), 11–30. <https://doi.org/10.1007/BF02110042>
- Shapiro, A., Dentcheva, D., & Ruszczyński, A. P. (2014). *Lectures on stochastic programming: Modeling and theory* (Second edition). Society for Industrial and Applied Mathematics : Mathematical Optimization Society. <https://doi.org/10.1137/1.9781611973433>
- Tewari, S., Geyer, C. J., & Mohan, N. (2011). A Statistical Model for Wind Power Forecast Error and its Application to the Estimation of Penalties in Liberalized Markets. *IEEE Transactions on Power Systems*, *26*(4), 2031–2039. <https://doi.org/10.1109/TPWRS.2011.2141159>
- Vaagen, H., & Wallace, S. W. (2008). Product variety arising from hedging in the fashion supply chains. *International Journal of Production Economics*, *114*(2), 431–455. <https://doi.org/10.1016/j.ijpe.2007.11.013>
- Vaagen, H., Wallace, S. W., & Kaut, M. (2011). Modelling consumer-directed substitution. *International Journal of Production Economics*, *134*(2), 388–397. <https://doi.org/10.1016/j.ijpe.2009.11.012>
- Van Loan, C. F. (1976). Generalizing the Singular Value Decomposition. *SIAM Journal on Numerical Analysis*, *13*(1), 76–83. <https://doi.org/10.1137/0713009>
- Wallace, S. W. (2010). Stochastic programming and the option of doing it differently. *Annals of Operations Research*, *177*(1), 3–8. <https://doi.org/10.1007/s10479-009-0600-x>

- Wallace, S. W., & Ziemba, W. T. (Eds.). (2005). *Applications of stochastic programming*. Society for Industrial and Applied Mathematics.
- Wang, K., & Jacquillat, A. (2020). A Stochastic Integer Programming Approach to Air Traffic Scheduling and Operations. *Operations Research*, 68(5), 1375–1402. <https://doi.org/10.1287/opre.2020.1985>
- Wets, R. J.-B. (1996). Challenges in stochastic programming. *Mathematical Programming*, 75(2), 115–135. <https://doi.org/10.1007/BF02592149>
- Zhao, Y., & Wallace, S. W. (2016). Appraising redundancy in facility layout. *International Journal of Production Research*, 54(3), 665–679. <https://doi.org/10.1080/00207543.2015.1030041>

## Paper II

# On the Safe Side of Stochastic Programming: Bounds and Approximations

Benjamin S. Narum, Francesca Maggioni and Stein W. Wallace

### Abstract

There will always be stochastic programs that are too large or complex to be solved in their basic form. In this article, we review, discuss, and compare different ways such stochastic programs can be handled using bounds and approximations, all based on manipulations of the random variables. We are particularly interested in how methods based on different underlying ideas can be combined or possibly are the same.

## 1 Introduction

Stochastic programming is an important tool for dealing with decision problems when the outcome of a decision is stochastic. This is qualitatively different from deterministic approaches and considers concepts such as options, hedging and risk that do not exist in a deterministic world. Optimal decisions to stochastic programs consider different future outcomes simultaneously and consequently allocate resources to obtain flexibility in dealing with these futures effectively if they occur.

The advantage of stochastic programming, as opposed to other approaches to decision-making under uncertainty, is its tight link to mathematical programming and its tools to solve large-scale constrained optimisation problems that may include combinatorial considerations by integers. The field of mathematical programming has seen immense improvements in the ability to get precise solutions to optimisation problems; however, stochastic programs also require *approximation of the stochastics*. As a result, the mathematical program solved may be very close to, or at times very far from, the underlying problem with its stochastics. It all depends on the quality of the approximation.

Stochastic programming problems are formulated as optimising the expectation (or some other risk measure) over stochastic variables that take the role of parameters in an optimisation problem. Exact solutions to these are, in general, fundamentally intractable.

This paper gives an overview of bounds and approximations with respect to the distributional aspects of stochastic programming. Our primary goal is to give an intuitive overview to provide an understanding of the underlying ideas of these bounds. While there are many associated technical details, we aim to simplify the exposition as much as possible while still conveying the principal ideas, distinctions and connections between different approaches. References are listed throughout for the interested reader to explore in more detail.

We refer readers to the book by King and Wallace (2012) for the modelling aspects of stochastic programming while Kall and Wallace (1994), Birge and Louveaux (2011) and Shapiro et al. (2014) explain the mathematical background, solution procedures and more detailed technical properties. This section discusses the uses of bounds and some of their implications for modelling.

**Motivation on bounds** Models in stochastic programming aim to optimise a current decision while additionally accounting for the limitations this puts on future decisions determined once the outcomes of uncertain parameters are revealed. Future decisions reflect reactive actions once more information is known and are usually determined by an optimisation model with no closed-form solution. This means the objective function of the current decision is to optimise the expectation of another optimisation problem. Needless to say, evaluation of this expectation poses a great challenge (especially if the distribution is continuous) and must be approximated in some way. Furthermore, we also aim to optimise this expected value (whose evaluation is intractable) with respect to the current decision. The literature on bounds aims to find tight interval estimates of these quantities. In this paper, we explore bounds both on the *evaluation of expected values* and on the *optimised expected values*. The distinction between these is important because they elicit different kinds of argumentation.

In a mathematical and algorithmic sense, bounds are important to determine if an error gap is sufficiently low so that the problem can be considered solved or that estimates of certain quantities are sufficiently accurate. The former relies on some version of the latter but has different conceptual purposes. Bounds on the evaluation of expectations can be used to reveal the current performance of a given solution in terms of the original (intractable) objective function but we are also interested in whether the solution could have been better by comparing it to an interval estimate of the optimal objective value. This distinction is especially interesting since finding a solution is subject to (an often different) approximation, and the effectiveness of this approximation can be benchmarked by first estimating its performance and then deciding if this is sufficiently close to being optimal.

There is also an application side of bounds. In the context of minimisation (which we use throughout), a *lower bound is optimistic* while an *upper bound is conservative*. This means we are generally more interested in upper bounds since these are more informative of our true exposure. Investment and pricing problems are especially interesting areas to apply the evaluation bounds in Section 3 because such problems consist of deciding whether to get exposed or not, and evaluating this exposure is essential. An overly optimistic account of the objective value may lead to the phenomenon known as the *winner's curse* (Thaler, 1988), where the bidder having the worst value estimate wins the bid. If we are already invested, the perspective changes, and we mostly care about making good decisions. The bounds in Section 4 are then of particular interest since these can be used to evaluate how close a solution is to the actual optimal objective value.

Generally, more uncertainty leads to a worse performance if we don't plan for it.

Ignoring uncertainty produces a decision whose real performance is an upper bound on the best attainable performance and gets better with a better account of uncertainty. This is worth remembering when deterministic models are applied in stochastic environments. A naive approximation of the underlying distribution may also lead to bounds instead of estimates, a surprisingly easily made mistake.

The structure of the paper is as follows: background and notation is described in Section 2. In Section 3, we explore bounds on the evaluation of expected values, while Section 4 emphasises bounds on the optimal value of such expected values. The argumentation of the approaches in these two sections is fundamentally different but has interesting connections. In Section 5, we conclude the paper.

## 2 Background and notation

We present the notation used throughout the paper for *minimisation* of stochastic programs under uncertainty. Such formulations may be stated as *two-stage* or *multistage* problems, but the same notation is used for both. Unless explicitly stated otherwise, we refer to both kinds of formulations. Risk aversion is omitted, which is why these two formulations are very similar in notation. We assume all relevant optimisation problems in the paper have finite optimal solutions.

We consider a finite-horizon sequential decision-making problem under uncertainty where decisions are made at discrete stages  $t \in \{0, \dots, T\}$  and  $T$  denotes the (finite) planning horizon, while  $[t] := \{0, \dots, t\}$  denotes all stages up to  $t$ . The uncertain data  $\xi := (\xi_1, \dots, \xi_T)$  is revealed gradually over time from stage  $t = 1$ , and our decisions should be adapted to this process. The decision process takes the form

$$\begin{array}{ccccccc} x_0 & \rightarrow & \xi_1 & \rightarrow & x_1 & \rightarrow \cdots \rightarrow & \xi_T & \rightarrow & x_T \\ \text{decision} & & \text{observation} & & \text{decision} & & \text{observation} & & \text{decision} \end{array},$$

where each  $\xi_t$  has support  $\Xi_t \in \mathbb{R}^{d_t}$ , and overall  $\xi$  has support  $\Xi := \Xi_1 \times \dots \times \Xi_T$ . The history up to stage  $t$  is denoted  $\xi_{[t]} := (\xi_1, \dots, \xi_t)$ . For consistency, we let  $\xi_0$  represent an initial deterministic state known in advance of the first decision. The decision process begins with initial decision  $x_0$  at stage  $t = 0$ , called the *first-stage* or *here-and-now* decision, followed by sequential decisions  $x_t$  at stages  $t = 1, \dots, T$ . The history of the decision process, at a given point in time  $t$ , is denoted by  $x_{[t]} := (x_0, x_1, \dots, x_t)$ .

An important concept associated with uncertainty is the corresponding *information structure*  $\mathcal{F} := (\mathcal{F}_0, \dots, \mathcal{F}_T)$  which describes the availability of information at each stage. We refer to outcomes by the underlying outcome space  $\Omega$ , and information is encoded by whether such outcomes can be distinguished at a given point in time. Here,  $\mathcal{F}_t$  denotes the available information at stage  $t$ , and we have the criterion that  $\mathcal{F}_t \subseteq \mathcal{F}_{t+1}$  which signifies that information is gained over time. Continuous outcome spaces require the notion of nested  $\sigma$ -algebras over  $\Omega$  to encode the revelation of information, while for discrete outcome spaces, this can also be represented by a tree structure (see Fig. II.1). Implicitly,  $\xi_t : \Omega \rightarrow \mathbb{R}^{d_t}$  is here a mapping from outcomes  $\omega$  to realisations of the uncertain data  $\xi_t(\omega)$ , and similarly for the decision process  $x_t : \Omega \rightarrow \mathbb{R}^{n_t}$ . We generally also refer to  $x(\omega) = (x_0, x_1(\omega), \dots, x_T(\omega))$  as a *decision policy*, which is said to be *feasible* if it satisfies the constraints  $x(\omega) \in \mathcal{X}(\omega) := \mathcal{X}_0(\omega) \times \dots \times \mathcal{X}_T(\omega)$  where  $\mathcal{X}_t(\omega) \subseteq \mathbb{R}^{n_t}$ . Whenever the specific value of the uncertain data process matters, we refer to it by  $\xi$ , while we generally use  $\omega$  as a generic referral to outcomes. Note that processes  $\xi(\omega)$  and  $x(\omega)$  must always adhere to the information structure  $\mathcal{F}$ , meaning they are defined with respect to it. This also means the decision policy  $x_t(\omega)$  at stage  $t$  may only depend on information  $\mathcal{F}_t$  available up to stage  $t$ , also known as *nonanticipativity*.

We let  $f_t(x_{[t]}(\omega), \xi_{[t]}(\omega))$  denote the cost incurred at stage  $t$  given the decisions  $x_{[t]}(\omega)$  and stochastic process  $\xi_{[t]}(\omega)$  up to stage  $t$ , given by outcome  $\omega$ . Let  $\mathbb{E}_t^P[\cdot] := \mathbb{E}_{\mathcal{F}_t}^P[\cdot]$  denote the conditional expectation with respect to all information up to stage  $t$  using distribution  $P$ , and let  $\mathbb{E}^P[\cdot]$  denote the unconditional expectation.

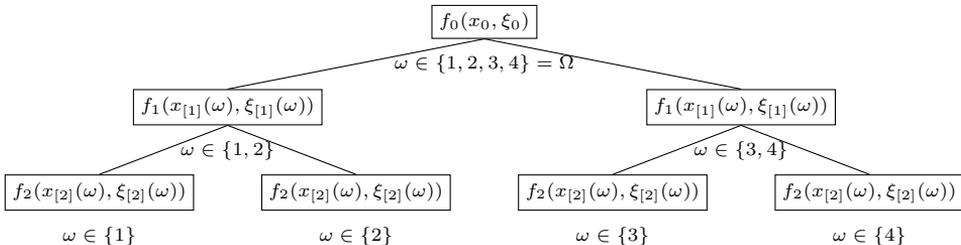
A  $(T + 1)$ -stage stochastic programming problem can be formulated as

$$v(P) := \min_{x_0 \in \mathcal{X}_0} f_0(x_0, \xi_0) + \mathbb{E}_0^P \left[ \min_{x_1(\omega) \in \mathcal{X}_1(\omega)} f_1(x_{[1]}, \xi_{[1]}) + \mathbb{E}_1^P \left[ \cdots \right. \right. \\ \left. \left. \cdots + \mathbb{E}_{T-1}^P \left[ \min_{x_T(\omega) \in \mathcal{X}_T(\omega)} f_T(x_{[T]}, \xi_{[T]}) \right] \right] \right] \quad (\text{II.1a})$$

$$= \min_{x(\omega) \in \mathcal{X}(\omega)} \mathbb{E}_P \left[ \sum_{t=0}^T f_t(x_{[t]}(\omega), \xi_{[t]}(\omega)) \right] = \min_{x(\omega) \in \mathcal{X}(\omega)} \mathbb{E}_P [f(x(\omega), \xi(\omega))], \quad (\text{II.1b})$$

where  $v(P)$  is the optimal objective value under distribution  $P$  and the shorthand notation  $f(x(\omega), \xi(\omega))$  in (II.1b) denotes the total incurred cost for the given process  $\xi$  and decision  $x$  in outcome  $\omega$ . Notice that the cost functions  $f_t$  can possibly be non-linear.

For discrete outcome spaces  $\Omega$ , it is useful to depict the principle of information structure as a tree where each  $\omega$  represents a path from the root to the leaves. This is illustrated in Fig. II.1 with the associated notation. The tree has nodes organised in levels corresponding to stages  $t \in \{0, \dots, T\}$ . At level  $t = 0$ , we have a single *root node* associated with the known quantity  $\xi_0$  and the first-stage decision  $x_0$ . At level  $t = 1$ , we have multiple nodes representing different realisations of  $\xi_1$  and associated decisions  $x_1$  that depend on this realisation. Generally, nodes correspond to possible realisations of uncertainty, and each node has an associated set of outcomes that describes which paths pass through it. Information consists of knowing this set of outcomes but without the ability to distinguish exactly which one occurs in the future. Hence, we must plan for all of them simultaneously (weighted by probability).



**Figure II.1:** A scenario tree representing the information structure and the objective value associated with different nodes.

By the theory of duality in optimisation, we may dualise this optimisation problem (II.1) with respect to its constraints  $\mathcal{X}(\omega)$ , and obtain

$$v(P) = \min_{x(\omega) \in \mathcal{X}(\omega)} \mathbb{E}_P [f(x(\omega), \xi(\omega))] \geq \max_{\lambda(\omega) \in \Lambda(\omega)} \mathbb{E}_P [f^*(\lambda(\omega), \xi(\omega))] = v^*(P), \quad (\text{II.2})$$

by weak duality, where  $f^*(\lambda(\omega), \xi(\omega))$  is the dual objective function,  $v^*(P)$  its optimal objective value,  $\lambda(\omega)$  a dual policy, and  $\Lambda(\omega)$  the dual constraints. All quantities in the dual problem also adhere to the same information structure  $\mathcal{F}$  and follow the same notation conventions.

## 2.1 Stochastic linear programs

Stochastic linear programs are the primary formulation used for applications of stochastic programming and may also be extended by including integer variables. These take a particular form with linear objective  $f(x(\omega), \xi(\omega)) = c(\omega)^\top x(\omega)$  and linear constraints  $\mathcal{X}(\omega) = \{x(\omega) : A(\omega)x(\omega) \geq b(\omega)\}$ . Properties of these are used as assumptions in Section 3.4 and Section 4.3, while its parametric shape is important for the discussion in Section 3.3. The stochastic linear program (SLP) has formulation

$$\begin{aligned} \min_{x(\omega)} \quad & \mathbb{E}_{\mathcal{P}} [c(\omega)^\top x(\omega)] \\ \text{s.t.} \quad & A(\omega)x(\omega) \geq b(\omega), \quad \forall \omega \in \Omega \\ & x(\omega) \geq 0 \end{aligned} \tag{SLP}$$

with dual

$$\begin{aligned} \max_{\lambda(\omega)} \quad & \mathbb{E}_{\mathcal{P}} [b(\omega)^\top \lambda(\omega)] \\ & A(\omega)^\top \lambda(\omega) \leq c(\omega), \quad \forall \omega \in \Omega \\ & \lambda(\omega) \geq 0, \end{aligned} \tag{D-SLP}$$

where  $\xi(\omega)$  is represented by the random parameters  $c(\omega)$ ,  $A(\omega)$  and  $b(\omega)$  while  $\lambda(\omega)$  denotes the dual multiplier of the constraints of (SLP). This general formulation of the constraints has a time structure such that  $\xi_t(\omega)$  denotes the parameters  $c_t(\omega)$ ,  $A_t(\omega) := (B_t(\omega), W_t(\omega))$  and  $b_t(\omega)$  specific to stage  $t$ . The constraint matrix  $A(\omega)$  has a block structure incorporating decisions at previous stages. In particular, the constraints in the primal take the more specific form

$$A(\omega)x(\omega) \geq b(\omega) \iff \begin{cases} A_0 x_0 & \geq b_0, \\ B_1(\omega)x_0 + W_1(\omega)x_1(\omega) & \geq b_1(\omega), \\ B_2(\omega)x_{[1]}(\omega) + W_2(\omega)x_2(\omega) & \geq b_1(\omega), \\ \vdots \\ B_T(\omega)x_{[T]}(\omega) + W_T(\omega)x_T(\omega) & \geq b_T(\omega), \end{cases}$$

where the first constraint is deterministic, and the consecutive ones are stochastic.

There are some specific shapes and properties of (SLP) to take into consideration:

- *Right-hand side uncertainty*: If only the right-hand side  $b(\omega)$  is random, (SLP) is convex and piecewise linear in the random parameters.
- *Objective uncertainty*: If only the objective coefficient  $c(\omega)$  is uncertain, (SLP) is concave and piecewise linear in the random parameters.
- *Convex-concave saddle*: If we have both right-hand side  $b(\omega)$  and objective  $c(\omega)$  uncertainty, (SLP) is piecewise bi-linear in the random parameters and is also a convex-concave saddle function.
- *Fixed recourse*: If  $W_t$  is deterministic, this is referred to as *fixed recourse*, while otherwise this it is called *random recourse*. The shape of (SLP) in  $W$  is generally non-convex.

- *Complete recourse*: The assumption that (SLP) is always feasible, i.e., taking finite values, is referred to as complete recourse. Relatively complete recourse means feasibility in stage  $t$  is ensured for all previous feasible decisions  $\mathcal{X}_{[t-1]}$ .

These shapes follow from the parametric shape of LPs (Bertsimas & Tsitsiklis, 1997). Notice that bounds based on convexity are generally not applicable to mixed-integer linear programs (MILPs).

### Penalty formulations and soft-constraints

Penalty formulations of (SLP) using soft-constraints are especially useful to consider in the context of bounds. These can be used to define finite growth conditions (Section 3.3), to construct majorising functions (Section 3.4), and to have well-defined behaviour for sub-optimal candidate policies (Section 4). Soft constraints are constructed by adding non-negative auxiliary variables  $z(\omega)$  to the constraints as

$$A(\omega)x(\omega) + z(\omega) \geq b(\omega), \quad (\text{II.3})$$

that have associated penalties  $\mu \geq 0$  in the objective. This special form of constraint allows us to determine the solution of  $z$  easily as

$$z(\omega) = (b(\omega) - A(\omega)x(\omega))^+ := \max\{0, b(\omega) - A(\omega)x(\omega)\}, \quad (\text{II.4})$$

whose form is similar to those of simple recourse problems (Birge & Louveaux, 2011, §3.1d). Observe also that this reduces projection onto the feasible set to a trivial max operation. The primal objective can then be restated as

$$c(\omega)^\top x(\omega) + \mu^\top (b(\omega) - A(\omega)x(\omega))^+, \quad (\text{II.5})$$

while relaxing all constraints but the variable bounds on  $x(\omega)$ . A direct consequence of soft constraints is that we have bounds  $0 \leq \lambda(\omega) \leq \mu$  on the dual variables. Intuitively, whenever the dual multiplier (the shadow price) reaches  $\mu$ , the penalty takes over instead by violating the constraint. Consequently, penalties should be chosen according to the true marginal cost of violating these constraints. Analogously, soft constraints in the dual formulation using penalties  $\nu$  give bounds  $0 \leq x(\omega) \leq \nu$  on the primal variable, and the dual formulation is always feasible. The dual objective can then be restated as

$$b(\omega)^\top \lambda(\omega) + \nu^\top (A(\omega)^\top \lambda(\omega) - c(\omega))^+, \quad (\text{II.6})$$

while relaxing all constraints but the variable bounds on  $\lambda(\omega)$ .

The geometric interpretation of these penalty formulations is that the optimal objective values of the primal and dual formulations, as functions of the stochastic parameters, are enclosed in pointed cones whose growth rate is determined by the penalties  $\mu$  and  $\nu$ . In the special case of simple recourse, the objective function is equal to this cone.

## 3 Bounds on the evaluation of expectations

When the underlying distribution  $P$  is known, for example, by an analytical expression or by specific information about its moments, there exist methods to get guaranteed (as opposed to statistical) bounds on the expected objective value without solving the intractable integral implied by the original problem formulation (II.1). An essential property of

the bounds reviewed in this section is that they require evaluating the objective function only on a finite set of outcomes, also when the underlying distribution is continuous.

In this section, we primarily address two-stage problems (i.e.,  $T = 1$ ) for a *fixed* first-stage decision  $x_0$  and uncertainty in the form of a *random vector*  $\xi$  on support  $\Xi \subseteq \mathbb{R}^d$ . For ease of notation in this section (Section 3), we let  $\xi_i$  denote component  $i$  of the random vector and omit the time index of  $\xi$ . The function to be approximated is then stated as

$$\phi(\xi) := \xi \mapsto \min_{x_1 \in \mathcal{X}_1(\omega)} f_1(x_{[1]}, \xi), \quad x_0 \text{ fixed}, \quad (\text{II.7})$$

which represents the optimisation problem solved once the uncertainty  $\xi$  is revealed. This can be time-consuming to evaluate, which motivates having few evaluation points to determine bounds on its expectation  $\mathbb{E}_P[\phi(\xi)]$ . Extension to multistage problems is discussed in Section 3.3.

Guaranteed bounds are mainly based on the shape of  $\phi$ . These bounds have two closely linked interpretations, but many of the bounds we discuss have been developed primarily with one perspective in mind. The two interpretations of guaranteed bounds are:

1. **Distribution approximation:** Find an alternative distribution  $Q$  whose evaluation of the expected objective function gives a bound.
2. **Function approximation:** Find a simpler function that is strictly equal or larger/smaller than  $\phi$  and whose expectation is easily evaluated.

Distribution approximation can intuitively be understood as moving probability mass to parts of the distribution support that combined has a larger/smaller objective value. Function approximation is intuitively understood by the fact that a function that is larger than another must also have a larger expectation.

The link between distribution approximation and function approximation and the more general framework of generalised moment problems are discussed in Section 3.1. An interesting generalising property of distributional bounds is discussed in Section 3.2. Bounding approaches primarily motivated by distribution approximation are discussed in Section 3.3 and those motivated by function approximations in Section 3.4. Lastly, Monte Carlo integration is an often used alternative approach to evaluate expectations, discussed in Section 3.5.

### 3.1 Guaranteed bounds as a Generalised Moment Problem

The *generalised moment problem* (GMP) has important implications for deriving bounds on the expectation of a function when certain limited information about the distribution is known. To some degree, this underlies all of the guaranteed bounds. The GMP is a semi-infinite program whose formulation aims to optimise the expectation of a function over an altered distribution that is subject to certain generalised moment constraints. While its formulation is very general and solutions are not always easily given, certain setups of the GMP give easily found or even analytical solutions. We primarily explore these easily found solutions in this paper and leave out details about semi-infinite programs.

The link to bounds in stochastic programming started with Madansky (1959), who used the moment problem to prove upper bounds on convex functions and was later tied to optimisation of the distribution to get upper and lower bounds by Dupačová (1987), Klein Haneveld (1986) and Kall (1988), originally pioneered by Dupačová (1966) with the minimax formulation of stochastic programs. The optimal distribution of the GMP has

also been referred to as an extreme measure which is particularly interesting because it has been shown that these are discrete and finite, even if the original distribution  $P$  is continuous (Karr, 1983; Kemperman, 1968). Duality further shows that the distribution approximation by the GMP has dual formulation with the direct interpretation of function approximation, which makes these two perspectives particularly tightly connected. The GMP can be solved directly to obtain bounds but is based on an iterative procedure that involves solving non-convex sub-problems (Birge & Wets, 1987), and we instead emphasise simpler procedures.

We now state the mathematical framework of guaranteed bounds starting at the interpretation of distribution approximation, function approximation and implications of their link by strong duality. Lastly, we give an illustrative example in Section 3.1.

**Distribution approximation** Consider that we want to find an alternative distribution  $Q$  on support  $\Xi$  to evaluate the expectation  $\mathbb{E}_Q[\phi(\xi)]$  such that it bounds the true objective  $\mathbb{E}_P[\phi(\xi)]$  from above or below. The first observation to make is that the expectation is linear in the distribution (namely, in the probability assigned to each outcome) irrespective of the functional shape of  $\phi(\xi)$ . This means we can pose the bounding problem as a (semi-infinite) linear program where the distribution  $Q$  is optimised with respect to the expectation  $Q \mapsto \mathbb{E}^Q[\phi(\xi)]$ , subject to constraints  $Q \in \mathcal{P}$  where  $\mathcal{P}$  is a specific class of probability distributions. We may then intuitively maximise to get an upper bound and minimise to get a lower bound. The formulation for obtaining a lower bound by distribution alteration (thereof the abbreviation D) is

$$\min_{Q \in \mathcal{P}} \mathbb{E}^Q[\phi(\xi)], \quad (\text{D-LB})$$

and

$$\max_{Q \in \mathcal{P}} \mathbb{E}^Q[\phi(\xi)], \quad (\text{D-UB})$$

for the upper bound. An optimal solution to such formulations is denoted  $Q^*$ . In general, we must assume the support  $\Xi$  is bounded for  $Q^*$  to be well defined (extensions to unbounded support are discussed in Section 3.3).

Conveniently, the optimisation problems (D-LB) and (D-UB) are optimal at discrete distributions  $Q^*$ , even if  $P$  is continuous. This is given that we have a finite number of constraints of the general form

$$\mathbb{E}^Q[g_i(\xi)] = m_i := \mathbb{E}^P[g_i(\xi)], \quad \forall i \in \mathcal{I} \quad (\text{II.8})$$

where  $\mathcal{I}$  is a finite index set of constraints (Kemperman, 1968, Theorem 1). The *generalised moment functions*  $g_i(\xi)$  for  $i \in \mathcal{I}$  can take various forms but should be linearly independent. Common expressions for these are linear functions, moment functions, or multi-linear functions. We also require that probabilities under  $Q$  sum to one, which can be stated in the same form,

$$\mathbb{E}^Q[1] = Q(\Xi) = 1 = P(\Xi) = \mathbb{E}_P[1].$$

The constraint set on  $Q$  is then summarised as

$$\mathcal{P} := \left\{ Q : \begin{array}{l} Q(\Xi) = 1, \\ \mathbb{E}^Q[g_i(\xi)] = m_i, \quad \forall i \in \mathcal{I} \end{array} \right\}. \quad (\text{II.9})$$

We assume that the prescribed values  $m_i$  are easily found and that they are consistent with each other.<sup>1</sup> According to the definition of  $m_i$  we always have that  $P \in \mathcal{P}$ , which

<sup>1</sup>If freely specified, it is possible to choose  $m_i$  so that  $\mathcal{P}$  is empty.

means  $P$  is always feasible to (D-LB) and (D-UB). Their respective optimal solutions  $Q^*$  must thus indeed provide bounds on  $\mathbb{E}^P[\phi(\xi)]$  by optimising over the same set  $\mathcal{P}$ . Furthermore, the optimal approximating distribution  $Q^*$  has at most  $|\mathcal{I}| + 1$  points in its support (Kemperman, 1968). The contributions on different bounds in the literature lie in choosing generalised moment functions  $g_i(\xi)$  such that  $Q^*$  is easily found, which is treated in more detail in Section 3.3.

**Function approximation** We now turn to the perspective of function approximation. Instead of looking for a new distribution  $Q$  to evaluate expectations over basis functions  $g_i(\xi)$ , we now aim to find an affine combination of them

$$\tilde{\phi}(\xi) := u_0 + \sum_{i \in \mathcal{I}} u_i g_i(\xi), \quad (\text{II.10})$$

that approximates  $\phi$  well from either above or below, where  $u$  is a vector of  $|\mathcal{I}| + 1$  elements that must be chosen appropriately. Recall that the true expectations of  $g_i(\xi)$  are known to be  $m_i$ , so

$$\mathbb{E}_P[\tilde{\phi}(\xi)] = u_0 + \sum_{i \in \mathcal{I}} u_i m_i. \quad (\text{II.11})$$

In the case of a lower bound, we constrain  $u$  such that  $\tilde{\phi}(\xi)$  approximates  $\phi(\xi)$  from below, while for the upper bound,  $u$  is constrained such that  $\tilde{\phi}(\xi)$  approximates  $\phi(\xi)$  from above. The approximating problem from the perspective of function approximation (thereof the abbreviations F) has the formulation

$$\max_u \left\{ \mathbb{E}_P[\tilde{\phi}(\xi)] = u_0 + \sum_{i \in \mathcal{I}} u_i m_i \quad \text{s.t.} \quad \tilde{\phi}(\xi) \leq \phi(\xi), \quad \forall \xi \in \Xi \right\}, \quad (\text{F-LB})$$

for a lower bound and

$$\min_u \left\{ \mathbb{E}_P[\tilde{\phi}(\xi)] = u_0 + \sum_{i \in \mathcal{I}} u_i m_i \quad \text{s.t.} \quad \tilde{\phi}(\xi) \geq \phi(\xi), \quad \forall \xi \in \Xi \right\}, \quad (\text{F-UB})$$

for an upper bound. An optimal solution to these formulations is denoted  $u^*$ .

**Duality** Distribution approximation and function approximation are really two sides of the same coin: they are linked by *strong duality*. This means the distribution alteration problem (D-LB) has dual formulation (F-LB), and these have respective solutions  $Q^*$  and  $u^*$  whose objective values are equal. Correspondingly for (D-UB) and (F-UB). Strong duality holds by the insightful property that the approximating function  $\tilde{\phi}(\xi)$  is *equal* to  $\phi(\xi)$  at values of  $\xi$  where  $Q$  assigns probability mass. This follows from complementarity slackness of the primal-dual pair, which states that

$$Q(\tilde{\phi}(\xi) = \phi(\xi)) = 1, \quad (\text{II.12})$$

namely, the probability under  $Q$  that the approximating function  $\tilde{\phi}(\xi)$  is equal to  $\phi(\xi)$  is one. Intuitively, this means the optimal function approximation meets the true function in the support of the optimal approximating distribution.

### Example: One dimensional bounded support

We present an example to illustrate these results on distribution alteration and function approximation to obtain bounds (see Fig. II.2). Assume we have a one dimensional uniform distribution  $P$  on support  $\Xi = [0, 1]$  with expectation 0.5. Given this distribution, we want to solve (D-LB) and (D-UB) where

$$\phi(\xi) = \xi^4, \quad g_i(\xi) = \xi^i, \quad \forall i \in \{0, 1, 2\},$$

are the objective and (generalised) moment functions, respectively, where  $i$  are the different moments we want to be consistent in the approximating distribution  $Q$ . The lower bounding problem is then formulated as

$$\min_Q \{ \mathbb{E}_Q [\phi(\xi)] \quad \text{s.t.} \quad \mathbb{E}_Q [\xi^i] = \mathbb{E}_P [\xi^i], \quad \forall i \in \{0, 1, 2\} \}, \quad (\text{II.13})$$

and maximising gives an upper bound instead. Denoting with  $u_i$  the dual variable of each constraint, the corresponding approximating functions are

$$\tilde{\phi}_M^{L/U}(\xi) = \sum_{i=0}^M u_i \xi^i.$$

where  $M + 1$  denotes the number of moments enforced, (L) refers to lower and (U) upper bounding functions, respectively. When solving the problem, we incrementally add each moment constraint to illustrate their differences. The lower-bounding approximating functions have been found to have expressions

$$\tilde{\phi}_0^L(\xi) = 0, \quad \tilde{\phi}_1^L(\xi) = -\frac{3}{16} + \frac{1}{2}\xi, \quad \tilde{\phi}_2^L(\xi) = -\frac{16}{27}\xi + \frac{4}{3}\xi^2$$

where the approximating distribution supports are  $\{0\}, \{1/2\}, \{0, 2/3\}$ , respectively. The upper-bounding approximating functions have expressions

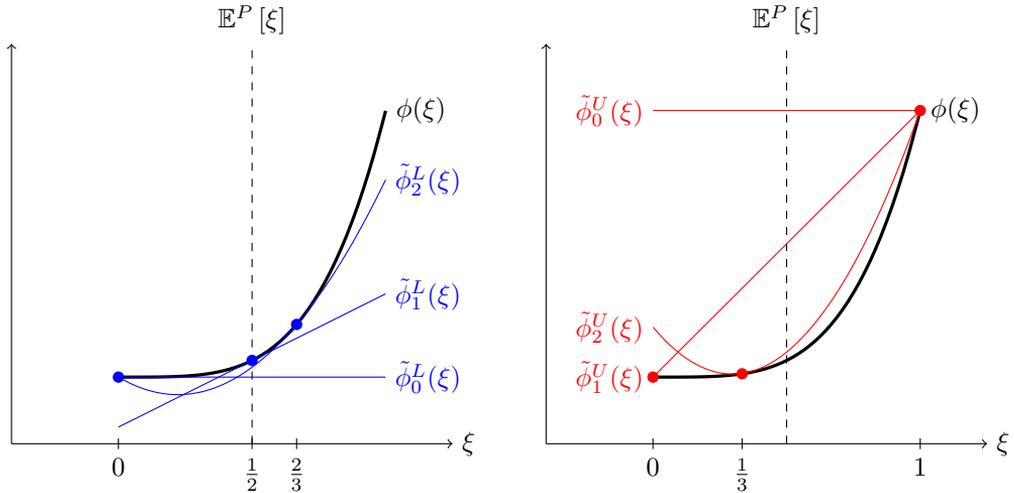
$$\tilde{\phi}_0^U(\xi) = 1, \quad \tilde{\phi}_1^U(\xi) = \xi, \quad \tilde{\phi}_2^U(\xi) = \frac{3}{16} - \frac{19}{16}\xi + 2\xi^2,$$

where the approximating distribution supports are  $\{1\}, \{0, 1\}, \{1/3, 1\}$ , respectively. These are plotted in Fig. II.2.

Observe (in Fig. II.2) that the zeroth moment approximation places all probability mass on the lowest and highest points of  $\phi(\xi)$  and the corresponding function approximation  $\tilde{\phi}$  is constant. This is an intuitive way of making conservative bounds but requires knowledge of the curve's lowest or highest point. When adding the first-moment basis function, we see that the lower bound places all probability mass in the expectation of  $\xi$  (not a coincidence) while the upper bound places all probability mass at the ends of the interval support of  $\xi$ . These are recoveries of the Jensen lower bound and the Edmund-Mandansky upper bound discussed in Section 3.3. We have also added the second-moment basis function as an illustration that this generalises for further basis functions. However, the support of the optimal  $Q^*$  is generally not as easily found, and there are fewer practical bounds based on this kind of information.

## 3.2 Generalisation to every decision and stochastic dominance

An important additional property of some of the bounds discussed in this section is that the approximating distribution  $Q$  may allow to bound  $\mathbb{E}_P [f(x, \xi)]$  for *any* decision  $x$ .



**Figure II.2:** Examples of lower (blue) and upper (red) approximations, with points at the position of the discrete distribution  $Q$  support where the approximations meet  $\phi(\xi)$ .

When this is the case, we may *solve* the stochastic program in terms of the objective  $\mathbb{E}_Q[f(x, \xi)]$  to obtain a tighter upper bound on the optimal value  $v(P)$  than possible when using only a single sub-optimal candidate solution.<sup>2</sup> Formally, this requires finding a  $Q$  such that

$$\mathbb{E}_P[f(x, \xi)] \leq \mathbb{E}_Q[f(x, \xi)], \quad \forall x \in \mathcal{X}, \quad (\text{II.14})$$

for an upper bound (or reversing the inequality for a lower bound). The relevance of this is limited to situations where a specific property of  $\xi \mapsto f(x, \xi)$  for all  $x \in \mathcal{X}$  is in itself sufficient to obtain the bounding distribution  $Q$  without explicit knowledge of the function. The property (II.14) is referred to as *stochastic dominance* (a wide area of research in itself), but for now, the most relevant property of  $\xi \mapsto f(x, \xi)$  is convexity, under which (II.14) is referred to as *convex stochastic dominance*. Under convexity, we may establish to a great extent where the approximating distribution  $Q$  should reside in its support  $\Xi$  without further information on the function itself. Finding  $Q$  then comes down to properties of the support  $\Xi$ , multivariate dependence in  $P$ , and the choice of generalised moment functions  $g(\xi)$ . Overall, the property of convex stochastic dominance greatly simplifies finding solutions to the bounding problems (D-UB) and (D-LB).

Majorising probability measures by ordering relations was explored first by Birge and Wets (1986) and pointed out by Wets (1984). Edirisinghe and Ziemba (1992) used the property of convex dominance to get optimised upper bounds, while Frauendorfer (1996), Kuhn (2005) and Frauendorfer et al. (2011) apply this to convex-concave multistage problems with linear time dependence. Maggioni and Pflug (2019) apply convex stochastic dominance to bound multistage problems with more general stochastic processes and risk-averse objective functions.

<sup>2</sup>More generally, the aim of minimising  $\max_{Q \in \mathcal{P}} \mathbb{E}_Q[f(x, \xi)]$  in terms of  $x$  is referred to as Distribution Robust Optimisation in the literature.

### 3.3 Distribution approximations

This section discusses approaches that are primarily motivated by distribution approximation to find bounds, where some are also solutions to the GMP. The bounds in this section make assumptions on the shape of  $\phi$ , multivariate dependence in  $P$ , the kind of support  $\Xi$ , and which generalised moment functions  $g(\xi)$  are used in the approximation. Table II.1 gives an overview of different approaches.

There are four relevant shapes of  $\phi$  to consider: (i) convex, (ii) concave, (iii) convex-concave, and (iv) monotonous. Convexity serves as a base case for our analysis. If it is instead concave, the analysis is the same, but lower and upper bounds must be switched. If it is instead convex-concave, these must be dealt with in an opposing manner, but the analysis is still largely the same (Section 3.3). Monotonous functions can be dealt with (although less effectively) by finding the highest or lowest evaluations in the support and may be considered a special case of the GMP using only zeroth-order conditions (see Example 3.1).

#### Jensen, Edmund-Madansky and hyper-planes on simplicial support

Jensen (1906) proved the (now classic) result that, for a convex function  $\phi$  of a random variable  $\xi$ , its expectation  $\mathbb{E}_P[\phi(\xi)]$  is bounded from below by the evaluation in the expectation of the random variable:

$$\phi(\mathbb{E}_P[\xi]) \leq \mathbb{E}_P[\phi(\xi)]. \quad (\text{II.15})$$

Furthermore, this only requires evaluation of  $\phi$  in a single point, which is of great practical importance when function evaluations can be time-consuming.

Edmundson (1957) showed how the expectation of a convex function of a one-dimensional random variable with a bounded interval support could be upper bounded by a linear function going through the end-points of the interval. Madansky (1959) posed this in the form of moment problems and showed it could be applied repeatedly for multivariate distributions with independence on rectangular support. Frauendorfer (1988) generalised these results to distributions with dependence by instead using multi-linear (generalised) moment functions

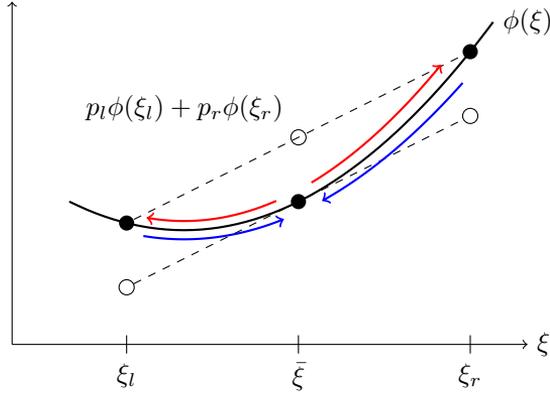
$$g_{\mathcal{I}}(\xi) = \prod_{i \in \mathcal{I}} \xi_i, \quad \forall \mathcal{I} \subseteq \{1, \dots, d\}, \quad (\text{II.16})$$

where  $\mathcal{I}$  are all combinations of each of the dependent random variables (i.e., in the power-set of  $\{1, \dots, d\}$ ). This gives an analytical expression for optimal probability mass to place in the extreme points on the rectangular support, while Kall (1987) showed this solution is unique. Furthermore, it was shown that the Edmund-Madansky result is a special case of the same expression where independence simplifies the expression. A disadvantage of the Edmund-Madansky (EM) type upper bound is that the number of evaluation points for the altered distribution scales exponentially as  $2^d$  in the dimensionality  $d$  of  $\Xi$ . Furthermore, it is often unrealistic to have accurate estimates of the expectations of the multi-linear generalised moment functions (II.16) in real-world settings.

These limitations of the EM type upper bound were mitigated by Birge and Wets (1986) and Gassmann and Ziemba (1986) who, effectively, used first-order moment functions

$$g_i(\xi) = \xi_i, \quad \forall i \in \{1, \dots, d\}, \quad (\text{II.17})$$

assuming simplicial support  $\Xi$  and general dependence structure. Birge and Wets (1986, Corollary 6.16) show that, under the assumption of convex  $\phi(\xi)$  and polytopal support  $\Xi$



**Figure II.3:** Moving probability mass on convex functions to obtain bounds with respect to linear approximating functions. Dashed lines illustrate the corresponding function approximations.

the bounding problem (D-UB) attains a solution on the extreme points of the polytope. When the polytope is *simplicial* (defined to have  $d + 1$  extreme points), the weights on the extreme points are uniquely determined by the set of  $d$  equations  $\mathbb{E}_P[\xi_i] = \mathbb{E}_Q[\xi_i]$  for all  $i \in \{1, \dots, d\}$ , as well as the conditions that probabilities sum to one. From the perspective of function approximation, this corresponds to finding an affine hyper-plane that meets  $\phi(\xi)$  in the extreme points of  $\Xi$ . Although this makes finding the upper-bounding distribution approximation much simpler, the EM type bound is still tighter (Kall, 1987).

Consider also that a bounded interval  $[a, b]$  is a simplicial set in one dimension, which means the EM bound is the one-dimensional counterpart of the hyper-plane approach. Independence and rectangular support mean the probabilities can be determined in each dimension separately, and the overall probability in each rectangular corner is determined as the product of the probability in each dimension.

The bounds based on zeroth- and first-order moments can intuitively be understood by directly applying the definition of convexity (see Fig. II.3). An affine line can be defined as a convex combination of two points. Placing these points closer to the expectation gives lower evaluations, and placing them further out gives higher evaluations. This observation is generalised by the concept of barycentric approximation for upper bounds.

### Barycentric approximation

More generally, we may state upper bounds as making convex combinations of the extreme points of a polytopal support (that is not simplicial) since the probability mass from upper bounding distribution approximations of convex functions always resides at the extreme points of the support (Birge & Wets, 1986). This representation also goes under the name of *barycentric approximation*, referring to the interpretation that a convex combination of the weighted extreme points is a barycenter (centre of mass). These developments are due to Birge and Wets (1986) and Frauendorfer (1992).

First, we express any point  $\xi$  within a polytopal set  $\Xi$  as a convex combination of its

extreme points  $\{\xi^{(1)}, \dots, \xi^{(K)}\} = \text{ext}(\Xi)$  with weights  $\{\rho_k(\xi)\}_{k=1, \dots, K}$ , subject to

$$\xi = \sum_{k=1}^K \rho_k(\xi) \xi^{(k)}, \quad \sum_{k=1}^K \rho_k(\xi) = 1, \quad \rho_k(\xi) \geq 0. \quad (\text{II.18})$$

This is also called a *barycentric coordinate* representation of  $\xi$ . The approximating function is then expressed as

$$\tilde{\phi}(\xi) = \sum_{k=1}^K \rho_k(\xi) \phi(\xi^{(k)}), \quad (\text{II.19})$$

which means

$$\phi(\xi) = \phi \left( \sum_{k=1}^K \rho_k(\xi) \xi^{(k)} \right) \leq \sum_{k=1}^K \rho_k(\xi) \phi(\xi^{(k)}) = \tilde{\phi}(\xi), \quad (\text{II.20})$$

by the definition that  $\phi$  is convex. Evaluating the expectation on both sides gives the upper bound

$$\mathbb{E}_P[\phi(\xi)] \leq \sum_{k=1}^K \bar{\rho}_k \phi(\xi^{(k)}), \quad (\text{II.21})$$

where  $\bar{\rho}_k = \mathbb{E}_P[\rho_k(\xi)]$  is the overall weight assigned to each extreme point given by its weight function  $\rho_k(\xi)$ . The possible caveat of barycentric approximation is obtaining  $\bar{\rho}_k$  since this involves a high dimensional integral over general functions  $\rho_k(\xi)$ . Preferably these are stated in a simple fashion. In particular, if  $\Xi$  is simplicial, we have that  $\rho_k(\xi)$  must be linear and  $\bar{\rho}_k$  are uniquely determined, which recovers the hyper-plane result in the previous section. If the support is rectangular, we can recover the dependent EM bound where  $\rho_k(\xi)$  are multi-linear and  $\bar{\rho}_k$  are again given by closed-form expressions.

### Higher-order bounds

Edirisinghe and Ziembra (1994a) and Edirisinghe (1996) extends the previous developments by a method using minorising affine functions to obtain lower bounds on convex functions. Originally, these were developed for convex-concave saddle functions (Section 3.3) where linearization in the convex components complements a barycentric approximation in the concave components but also led to new results on second-order lower bounds on convex functions. This lower bound is the first to improve on the Jensen bound by incorporating additional moment information.

Assume we have a given barycentric approximation  $\rho_k(\xi)$  on polyhedral support  $\Xi$ , potentially based on a special kind of support (simplicial or rectangular).<sup>3</sup> Since  $\phi$  is convex, we may create a minorising linearization

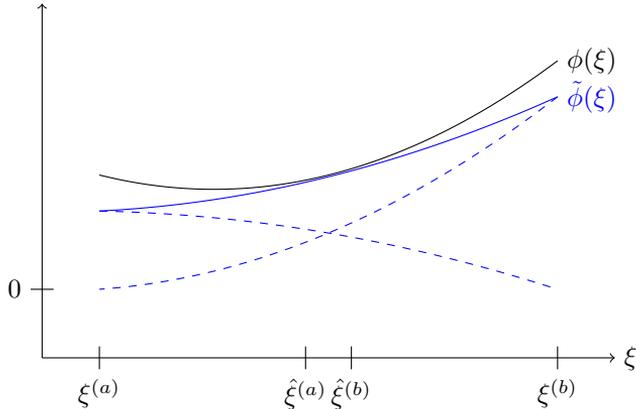
$$\phi(\xi) \geq \phi(\hat{\xi}) + \nabla_{\xi} \phi(\hat{\xi})^{\top} (\xi - \hat{\xi}) \quad (\text{II.22})$$

where  $\nabla_{\xi} \phi(\hat{\xi})$  is a sub-gradient of  $\phi$  with respect to  $\xi$  at a point  $\hat{\xi}$  (where the inequality is also tight). Consider that we may create such a linearization in certain points  $\hat{\xi}^{(k)}$  associated with each function  $\rho_k(\xi)$ . By the definition of barycentric coordinates we have that  $1 = \sum_{k=1}^K \rho_k(\xi)$ , which gives

$$\phi(\xi) = \sum_{k=1}^K \rho_k(\xi) \phi(\xi) \geq \sum_{k=1}^K \rho_k(\xi) \left[ \phi(\hat{\xi}^{(k)}) + \nabla_{\xi} \phi(\hat{\xi}^{(k)})^{\top} (\xi - \hat{\xi}^{(k)}) \right] = \tilde{\phi}(\xi), \quad (\text{II.23})$$

---

<sup>3</sup>If  $\rho_k(\xi)$  are degenerate (i.e., there exist multiple valid representations by the given moment information), the issue of finding the lower bound can instead be formulated as a non-linear convex optimisation problem (Edirisinghe & Ziembra, 1994a, 1996).



**Figure II.4:** Second-order lower bound by combined linearisation and barycentric approximation on an interval  $[\xi^{(a)}, \xi^{(b)}]$ . The evaluation points of the approximating distribution are  $\hat{\xi}^{(a)}$  and  $\hat{\xi}^{(b)}$ , while  $\tilde{\phi}$  is the separable approximating function.

where the linearization is specific to each  $k$ . Higher-order and cross-moment terms may be contained in the term  $\rho_k(\xi)\xi$ . By evaluating the expectation over (II.23), we find that

$$\mathbb{E}_P[\phi(\xi)] \geq \sum_{k=1}^K \bar{\rho}_k \left[ \phi(\hat{\xi}^{(k)}) + \nabla_{\xi} \phi(\hat{\xi}^{(k)})^{\top} \left( \frac{\mathbb{E}_P[\rho_k(\xi)\xi]}{\bar{\rho}_k} - \hat{\xi}^{(k)} \right) \right] = \sum_{k=1}^K \bar{\rho}_k \phi(\hat{\xi}^{(k)}), \quad (\text{II.24})$$

where the last equality follows by letting  $\hat{\xi}^{(k)} = \mathbb{E}_P[\rho_k(\xi)\xi] / \bar{\rho}_k$ , the point at which the  $k$ -specific linearization of  $\phi$  is tight. Separability of  $\tilde{\phi}$  is important as this allows finding each  $\hat{\xi}^{(k)}$  separately. The approximating distribution then has support  $\{\hat{\xi}^{(k)}\}_{k=1, \dots, K}$  with probabilities  $\hat{\rho}_k$ . An illustration for the one-dimensional case is given in Fig. II.4.

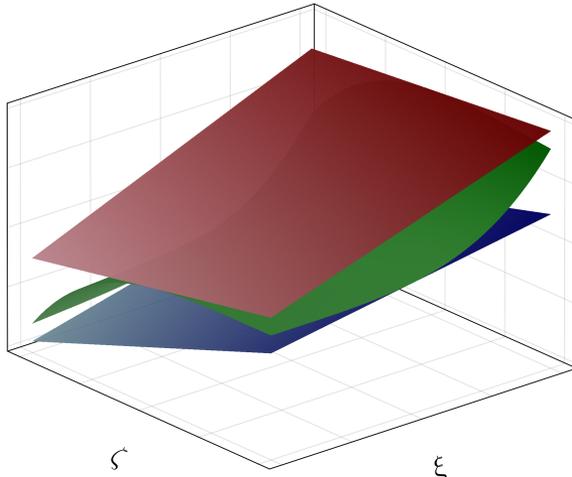
The points  $\hat{\xi}^{(k)}$  here have the interpretation of being the conditional expectation of  $\xi$  under the probability assignment  $\rho_k(\xi) / \bar{\rho}_k$ , which tends towards  $\xi^{(k)}$ . It also holds that  $\hat{\xi}^{(k)} \in \Xi$  as long as  $\bar{\rho}_k > 0$  while, otherwise,  $\hat{\xi}^{(k)}$  is irrelevant since  $\bar{\rho}_k$  is its associated probability mass. If  $\rho_k(\xi)\xi$  only contain cross moments, the bound coincides with the solution of (D-LB), while this is not the case for second-order bounds since the approximating distribution only preserves first-order and cross moments exactly. Fig. II.4 illustrates that the bounding function  $\tilde{\phi}(\xi)$  does not meet  $\phi(\xi)$ , which would be a requirement for this to hold. Dokov and Morton (2005) also provide a tighter second-order lower bound than (II.23), which requires more evaluation points, rectangular support, and independence, where the Edirisinghe (1996) bound is a special case. Neither this bound solves (D-LB) exactly.

Lastly, Dokov and Morton (2002) extend upper bounds on convex functions to arbitrary order using Bernstein polynomials, which gives a regular grid of evaluation points. As a special case, this also recovers the EM kind of bounds, with and without dependence.

### Convex-concave saddle functions

Convex-concave saddle functions, defined to be convex in some dimensions and concave in others, can be dealt with by treating these in an opposing manner. The primary concern for these is how to deal with dependence between the convex and concave components.

This section follows the exact developments of Section 3.3 and Section 3.3, while simplifying special cases are due to Frauentorfer (1992). From the perspective of function approximation, these bounds give bi-linear or multi-linear approximating functions on the convex-concave saddle function, as illustrated in Fig. II.5.



**Figure II.5:** Bi-linear upper (red) and lower (blue) bounds on a convex-concave function  $\phi(\xi, \zeta)$  (green).

Let  $\phi(\xi, \zeta)$  be a convex-concave saddle function that is convex in  $\xi$  and concave in  $\zeta$  and let the polytopal distribution support be  $\Xi \times Z$  where  $\xi \in \Xi$  and  $\zeta \in Z$ . This corresponds to appending  $\zeta$  to the multivariate random variable in the previous setup. Let  $\xi^{(k)}$  denote the extreme points of  $\Xi$  and  $\zeta^{(l)}$  the extreme points of  $Z$ . Again, we use barycentric approximations  $\rho_k(\xi)$  and  $\rho_l(\zeta)$  determined by these extreme points. By the saddle property of  $\phi(\xi, \zeta)$  and the barycentric approximation, we then have that

$$\sum_{l=1}^L \rho_l(\zeta) \phi(\xi, \zeta^{(l)}) \leq \phi(\xi, \zeta) \leq \sum_{k=1}^K \rho_k(\xi) \phi(\xi^{(k)}, \zeta), \quad (\text{II.25})$$

due to convexity in  $\xi$  and concavity in  $\zeta$ . Evaluating the expectation of (II.25) gives bounds on  $\mathbb{E}_P[\phi(\xi, \zeta)]$ , however, this cannot necessarily be easily evaluated and dependence between  $\xi$  and  $\zeta$  must be cared for when doing this. Under independence between  $\xi$  and  $\zeta$ , we may apply the Jensen inequality directly to (II.25) to obtain the bounds

$$\sum_{l=1}^L \bar{\rho}_l \phi_l(\bar{\xi}, \zeta^{(l)}) \leq \mathbb{E}_P[\phi(\xi, \zeta)] \leq \sum_{k=1}^K \bar{\rho}_k \phi(\xi^{(k)}, \bar{\zeta}), \quad (\text{II.26})$$

where  $\bar{\xi} = \mathbb{E}_P[\xi]$  and  $\bar{\zeta} = \mathbb{E}_P[\zeta]$ . Under dependence, however, we may proceed by applying the linearization procedure from Section 3.3 to expression (II.25), and obtain

$$\sum_{kl} \bar{\rho}_l \phi_l(\hat{\xi}^{(k)}, \zeta^{(l)}) \leq \mathbb{E}_P[\phi(\xi, \zeta)] \leq \sum_{kl} \bar{\rho}_k \phi(\xi^{(k)}, \hat{\zeta}^{(l)}), \quad (\text{II.27})$$

instead, where  $\hat{\xi}^{(k)}$  and  $\hat{\zeta}^{(l)}$  are determined by the procedures laid out in Section 3.3. Note that the linearization with respect to  $\xi$  uses the barycentric approximation  $\rho_l(\zeta)$  (as well as the converse), meaning the approximating functions have terms  $\rho_k(\xi)\zeta$  and  $\rho_l(\zeta)\xi$  expressed by first-order or cross moments only. As before, the bound using only first-order and cross moments are tight on the corresponding generalised moment problems (D-LB) and (D-UB). Under assumptions of simplicial or rectangular support, the barycentric approximations are unique, and the solutions are easily found in closed form as well as for the linearization points (Frauendorfer, 1992). Second-order moments can be added by using a single barycentric approximation defined in terms of the extreme points of  $\Xi \times Z$  instead (Edirisinghe, 1996).

### Subset refinement and unbounded support

The described bounds can be sharpened to arbitrary precision by refining the support  $\Xi$  into smaller subsets  $\{\Xi_k\}_{k=1,\dots,K}$  and computing the bounds on each of them. In the perspective of distribution alteration, we now use conditional expectations instead of total expectations while from the perspective of function approximation, we use piece-wise approximations within each subset instead of on the whole support.

The observation that these bounds could be refined to subsets started with Ben-Tal and Hochman (1972) who refined the Edmund-Madansky bound for positive linear functions, i.e.  $(\xi)^+ = \max\{0, \xi\}$ , and the implications of their result was generalised by Huang et al. (1977) to show that the bounds can be repeatedly applied to subsets of the support to refine them to arbitrary precision. Ben-Tal and Hochman (1972) showed that the bounds hold on intervals unbounded on one side if the convex function has a finite growth rate, which was extended to conic polyhedral sets by Gassmann and Ziemba (1986) and Birge and Wets (1986). Edirisinghe and Ziemba (1994b) extends the lower bounds by linearization (Section 3.3) to conic polyhedral support.

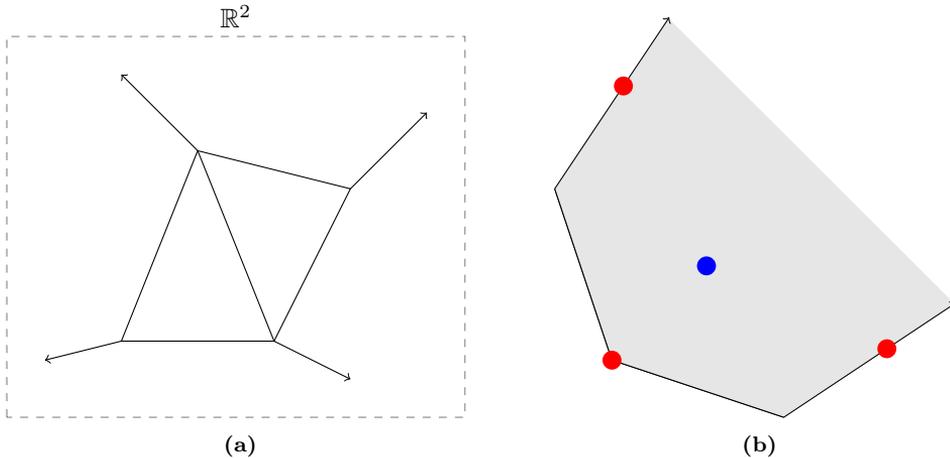
The subsets  $\Xi_k$  of  $\Xi$  must generally be polytopal or polyhedral cones, described by a finite number of extreme points and rays (see Fig. II.6a) and partition the support  $\Xi$ . A *polyhedral cone* is a polyhedron where for every unbounded direction it contains, it is bounded in the opposite direction. This avoids unbounded formulations of (D-LB) and (D-UB). Once a bound  $B(\Xi_k)$  is made on a subset  $\Xi_k$ , the overall bound  $B(\Xi)$  is found as  $B(\Xi) = \sum_{k=1}^K P(\Xi_k)B(\Xi_k)$ , where  $P(\Xi_k)$  denotes the probability of each subset.

Unbounded support requires the use of rays in its description. The fundamental requirement for unbounded support is that  $\phi$  has limited growth along any ray  $r$  starting at a point  $\xi \in \Xi$ . Namely,

$$\lim_{t \rightarrow \infty} \frac{|\phi(\xi + tr)|}{\|\xi + tr\|} \leq \mu_r < \infty, \quad \xi + tr \in \Xi, \quad (\text{II.28})$$

where  $\|\cdot\|$  is the Euclidean norm and  $t \in \mathbb{R}_+$ . In particular, (II.28) follows for linear recourse programs with complete recourse, and this can always be ensured by using penalty formulations (Section 2.1). For unbounded support, the optimal distribution approximation is situated on the extreme points of  $\Xi$ , possibly extended by the extreme rays (see Fig. II.6b). By the condition to preserve the first-order moment condition, we cannot extend along rays infinitely as long as the support is a polyhedral cone. The finite growth coefficients  $\mu_r$  determine the upper bound on the objective value with respect to the ray extensions.

As an example, the unbounded space  $\mathbb{R}^d$  can be described by conic polyhedral subsets in the form of every orthant (giving  $2^d$  subsets), which are polyhedral cones. While there



**Figure II.6:** (a) Partitioning a two-dimensional plane into polytopes and polyhedral cones. (b) Polyhedral cone with corresponding distribution approximations for an upper bound (red) at extreme points extended by rays and a lower bound (blue) in the interior.

are large differences in the number of evaluation points based on the kind of support for the bounds discussed, there is also a complexity to consider with respect to how a high-dimensional space  $\Xi$  is partitioned. This means there is a trade-off between the number of evaluation points for a subset and the number of partitioning subsets of the space.

### Multistage case

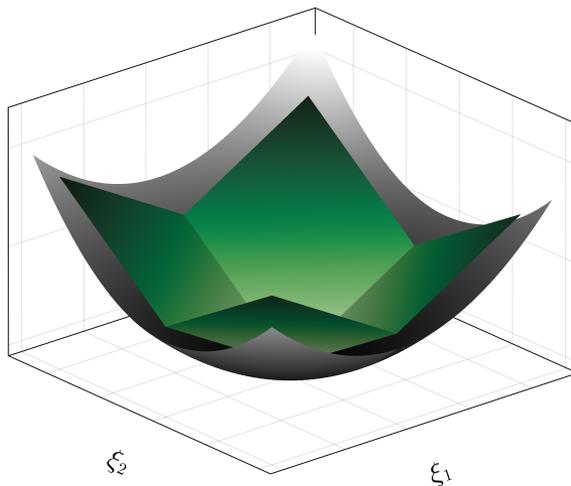
The main challenge with the multistage extension of distribution approximation is that a discrete approximating distribution  $Q$  also defines a simplified information structure  $\hat{\mathcal{F}}$  (a discrete tree) which interferes with the conditional evaluation of expectations. The fundamental question is whether we can apply the previously discussed bounds stage-wise: given a discretisation for the current stage, generate a discretisation for the next stage conditional on each discrete outcome. If possible, stage-wise discretisation significantly simplifies the discretisation, but this does not always suffice. The violation may occur when evaluating the conditional expectation of a stochastic quantity more than one stage ahead since this relies on the discretisation in intermediate stages that did not explicitly account for this. Under linear additive time-dependence with stage-wise independent noise, stage-wise discretisation is sufficient, while, in general, the entire process must be accounted for.

Distributional approximation on multistage stochastic programs has been explored by Edirisinghe and Ziemba (1992), Frauendorfer (1994, 1996), Frauendorfer et al. (2011) and Kuhn (2005). All of them use first-order and cross moment information, and assume stage-wise independence or linear additive time-dependence. More recently, Maggioni and Pflug (2019) have explored this for more general stochastic processes and show by counter-example that the entire process across all stages must then be accounted for during discretisation.

### 3.4 Function approximations

Function approximations are largely based on specific assumptions about the function at hand. The most general assumption is convexity which also holds for stochastic linear programs under right-hand side uncertainty, but some also deal with specific properties of these.

For function approximation to be useful, the expectation of the approximating function must also be easily evaluated. This is achieved by using simple function expressions (linear, piecewise linear or quadratic) and by ensuring separability. Separability means the approximating function over the high-dimensional random vector  $\xi$  can be expressed as a sum of functions of fewer (or single) elements of  $\xi$ . This reduces one high-dimensional integral to evaluating many low-dimensional integrals, which is a considerable simplification. Independence in the distribution would also ensure multiplicative separability.



**Figure II.7:** A linear, separable and pointed cone (green), akin to a simple recourse function, that majorises a convex function (grey).

For upper bounds, two kinds of function expressions have been considered to approximate  $\phi(\xi)$  whose expectation is easy to evaluate. The aim is to fit these as tightly as possible within  $\phi(\xi)$ , whose procedures vary. First, there are separable piecewise linear functions

$$\tilde{\phi}(\xi) = u_0 + \sum_{i=1}^d u_i (\xi_i - \hat{\xi}_i)^+ + \sum_{i=1}^d u_{d+i} (\xi_i - \hat{\xi}_i)^-, \quad (\text{II.29})$$

where  $\hat{\xi}$  is a predetermined value of  $\xi$  (the expectation, for example) that defines the tip of the cone and  $u \in \mathbb{R}^{2d+1}$  vector of coefficients. See Fig. II.7. This approximating function primarily addresses growth conditions in the limits of  $\xi$ , and (II.29) may be refined by inserting a piecewise linear function within bounded intervals  $[\xi_i^l, \xi_i^u]$  and letting the unbounded limit be determined by (II.29) outside of this interval (Birge & Wets, 1989). Second, there are second-order expressions

$$\tilde{\phi}(\xi) = u_0 + \sum_{i=1}^d u_i \xi_i + \sum_{i=1}^d u_{d+i} \xi_i^2, \quad (\text{II.30})$$

whose primary aim is to alleviate the requirement of finding conditional expectations required by separable piecewise linear approximations. Consider, for example, that these are estimated from empirical data and that the confidence on second-order statistical estimates can be higher than for estimates of conditional expectations on many small subsets of the support. In fact, a requirement to find (II.30) is a (not necessarily separable) piecewise linear function which may be found using the procedures to find (II.29).

Lastly, there are penalty formulations of a two-stage stochastic linear program (SLP) suggested by Morton and Wood (1999), which give an analogous shape to (II.29), expressed as

$$\tilde{\phi}(\xi) = c_1(\omega)\hat{x}_1 + \mu^\top (B_1(\omega)x_0 + W_1(\omega)\hat{x}_1 - b_1(\omega))^+ \quad (\text{II.31})$$

where  $\hat{x}_1$  is a fixed decision inserted into the objective of a penalty formulation (Section 2.1), where  $\hat{x}_1$  itself can be chosen such that it minimises the bound. This may not be separable to one-dimensional integrals (only lower-dimensional) but can be applied more generally to (SLP) with random recourse. Observe that if only  $b_1(\omega)$  is stochastic, (II.31) takes the same form as (II.29). Morton and Wood (1999) also showed that if we have dual formulations of the optimisation problem that defines  $\phi(\xi)$ , we may apply similar bounds on the dual objective function to obtain lower bounds by weak duality. Such dual bounds are discussed in more detail in Section 4.3.

### Piecewise linear approximating functions

The bound developed by Birge and Wallace (1988) (that generalises (Birge & Wets, 1989; Wallace, 1987)) assume right-hand side uncertainty and bounded primal variables for two-stage (SLP), and finds a linear conic upper bounding approximation of the form (II.29) defined over unbounded support. See Fig. II.7. The coefficients that determine the growth of this approximation are found by parametric evaluation of an altered formulation of the problem that requires a number of evaluations proportional to the dimensionality  $d$  of the support. It may not always be possible to find such finite slopes by the given procedure, in which case the upper bound evaluates to  $+\infty$ . Under penalty formulations of (SLP), the slopes are upper bounded by the penalty  $\mu$  of constraint violation and guaranteed to give a finite upper bound. This also means this bound is tighter than (II.31) under the assumption of right-hand side uncertainty.

Powell and Frantzeskakis (1994) uses a similar simplification as Wallace (1987) to get simplified upper bounding recourse functions of network problems (and discuss different strategies for doing so) but instead use these to optimise decisions. This allows for solving larger problems while still capturing the essence of the recourse problem at hand.

### Second-order approximating functions

Dulá (1992) and Dulá and Murthy (1992) derived a second-order upper bound (II.30) that majorises a *known* piece-wise linear convex function  $\phi$ . This known function is motivated by the piece-wise linear form of two-stage SLPs having either right-hand side or objective uncertainty. However, obtaining these explicitly requires extreme point enumeration of the (deterministic) dual or primal feasible sets. Other majorising approximations of this nature may also be used instead. Using *total* second-order information only (such that coefficients of second-order terms in (II.30) are equal), Dulá (1992) find a closed form expression to determine the coefficients. This has no assumptions on dependence or boundedness of the support. Kall (1991) proposed an alternative perspective on this problem that leads to a non-smooth optimisation problem. Dulá and Murthy (1992)

extends this to *marginal* second-order moments, which tightens the bound under the same assumptions. No closed-form solution is found, but the bound can be found by minimising a non-linear convex function in  $d + 1$  variables with linear constraints.

Birge and Dulá (1991) use second-order information to bound a separable convex function  $\phi$ , whose approximation  $\tilde{\phi}$  is found by a line search. This directly extends the approach of Birge and Wallace (1988) but can also be applied to more general separable functions using its first-order derivatives.

### 3.5 Monte Carlo integration

The statistical counterpart to guaranteed integral bounds is to use Monte Carlo integration. That is, to sample an alternative distribution  $\tilde{P}$  from the underlying distribution  $P$  to get a finite statistical estimator  $\mathbb{E}_{\tilde{P}} [f(\tilde{x}, \xi)]$  of the true expectation  $\mathbb{E}_P [f(\tilde{x}, \xi)]$  for a *fixed* candidate decision  $\tilde{x}$ . Confidence intervals on this estimator can then derive statistical upper and lower bounds. This makes no assumption on the shape of  $\xi \mapsto f(\tilde{x}, \xi)$ , but it must be evaluated in a potentially very large number of sampled outcomes. The statistical properties rely on having a fixed decision, while optimisation with respect to  $\mathbb{E}_{\tilde{P}} [f(x, \xi)]$  leads to bias in the estimator (see Section 4.2).

The average recourse value  $\mathbb{E}^{\tilde{P}} [f(\tilde{x}, \xi)]$  is an estimator of the expected recourse value  $\mathbb{E}^P [f(\tilde{x}, \xi)]$  which, by the central limit theorem, is asymptotically normally distributed, i.e.

$$\mathbb{E}^{\tilde{P}} [f(\tilde{x}, \xi)] \sim \text{N} \left( \mathbb{E}^P [f(\tilde{x}, \xi)], \text{Var}^P [f(\tilde{x}, \xi)] / \sqrt{|\tilde{P}|} \right), \quad (\text{II.32})$$

where  $|\tilde{P}|$  is the number of samples in  $\tilde{P}$ . Importantly, (II.32) holds for multistage problems where the samples are outcome paths (not necessarily trees) assuming the policy  $\tilde{x}(\omega)$  provides decisions for every possible outcome path (Shapiro et al., 2014, Section 5.8.1). We may also extend by sampling the points of  $\tilde{P}$  by more elaborate schemes to reduce variance in the estimator and get improved convergence rate, referred to as Quasi-Monte Carlo methods (Shapiro et al., 2014, Chapter 5.4).

## 4 Bounds on optimal expected values

We now turn to bounds on the optimal objective value  $v(P)$ . These are solely based on arguments from optimisation theory: primal-dual formulations, sub-optimality, relaxation and feasibility. Such bounds are particularly interesting in the context of determining if a given candidate decision is sufficiently close to the best obtainable decision. Given stochastic programs are fundamentally intractable to solve in their basic form, these are especially useful when pursuing approximation methods aimed at finding (close to) optimal decisions, and the bounds are used for validation.

This section discusses three approaches to bounding optimal objective values. In Section 4.1, we address candidate evaluation bounds based on sub-optimality, where the important concept is how to insert approximations into the original intractable formulation. In Section 4.2, we address how a relaxation of information structure can give simplified problems that provide bounds. Lastly, Section 4.3 reviews decision rules and aggregation bounds derived from these based on relaxation, tightening and sub-optimality. Using decision rules is also an alternative approximation approach to those based on discretising distributions to obtain scenario trees.

## 4.1 Bounds based on candidate policies

These bounds evaluate the objective function using candidate policies to get a bound on the optimal objective value  $v(P)$ . Namely, if  $\tilde{x}(\omega)$  is a candidate primal policy and  $\tilde{\lambda}(\omega)$  a candidate dual policy, we have that

$$\mathbb{E}^P [f(\tilde{x}(\omega), \xi(\omega))] \geq v(P) \geq v^*(P) \geq \mathbb{E}^P [f^*(\tilde{\lambda}(\omega), \xi(\omega))]. \quad (\text{II.33})$$

Should  $\tilde{x}(\omega)$  or  $\tilde{\lambda}(\omega)$  be infeasible, we use the convention that this gives a primal or dual objective value of  $\pm\infty$ , which means these still give bounds but not very interesting ones. Infeasibility can be mitigated effectively by penalty formulations of stochastic programs (see Section 2.1), or by defining candidate policies according to rules that redeem feasibility. For the lower bound in (II.33) to be practical, the duality gap should be relatively tight and may require strong duality. For example, this is not the case for MIPs, but the upper bound still applies to these.

Obtaining candidate policies usually relies on some strategy of approximation. This is either done by discretisation into a scenario tree represented by a simplified information structure  $\hat{\mathcal{F}}$  and alternative distribution  $Q$ , or by decision rules (Section 4.3). When using scenario trees, we must define an *extension rule* that determines the candidate policy for outcome paths  $\omega$  not contained in the tree. These may consist of simpler or more elaborate rules but must adhere to the underlying information structure  $\mathcal{F}$ . Essentially, the extension rule in a given stage can only use information known at that stage. The concept of distances between trees is valuable in this context (Pflug & Pichler, 2012, 2014).

The simplest extension rule would be a nearest neighbour rule according to the distance from the tree to any given path  $\tilde{\xi}_{[t]}$  up to its realisation at stage  $t$  where the policy must be determined. Regression procedures have been proposed by Keutchayan et al. (2017), while Stochastic Dual Dynamic Programming (SDDP) can be applied to obtain piece-wise linear policies for problems of a special kind (Pereira & Pinto, 1991). Alternatively, we may incrementally optimise the decision in each stage, conditional on decisions determined in past stages. This is mainly useful in the context of two-stage problems where the first-stage decision is fixed, and the second-stage problem is easily evaluated without the need for approximation of further stages. This is also the tightest possible candidate evaluation bound for two-stage problems where only the first-stage decision is sub-optimal. For multistage problems, however, the consecutive optimisation problems represent almost as difficult problems as the full problem, and we often resolve to simpler and less optimal extension rules, which means the quality of the extension procedure also interferes with the evaluation. Xu and Sen (2023) also introduce the concept of compromise policies that are determined by optimising a policy over an ensemble of value function approximations derived from many scenario trees (with extension rules) to exploit the principle of multiple replications instead of using larger scenario trees to obtain candidate policies.

The simplest scenario tree to construct a candidate policy from would be to use expected path, whose candidate evaluation bound is known as the *Expectation of the Expected Value* (EEV) solution (Madansky, 1960). Maggioni and Wallace (2012) and Crainic et al. (2018) also analyse how candidate decisions in two-stage problems can be obtained using partial information from the expected value solution while the remaining variables are optimised in the full problem (assuming finite number of outcomes).

### Out-of-sample evaluation bounds

Out-of-sample evaluation refers to the procedure of evaluating the performance of candidate policies found by some approximation procedure, like scenario generation, and then validating its performance in terms of the original objective by Monte Carlo integration (Section 3.5) to ease its evaluation of the original objective value. This means we replace  $P$  with a sample distribution  $\tilde{P}$  assumed to be sufficiently large so that it represents  $P$  well. In the context of two-stage stochastic programs, Kaut and Wallace (2007) proposed this as an approach to determine if scenario generation is effective by comparing the quality of candidate decisions obtained by different scenario generation methods. Out-of-sample evaluation essentially gauges different scenario generation procedures against each other by comparing which method can, by whichever means, produce the candidate decision that has the best evaluation bound. This has been a tool for further development of scenario generation methods. The concept of extension rules for out-of-sample evaluation in multistage problems proposed by Keutchan et al. (2017) serves as a multistage counterpart. In principle, other evaluation bounds from Section 3 could also have been used to compare candidate decisions.

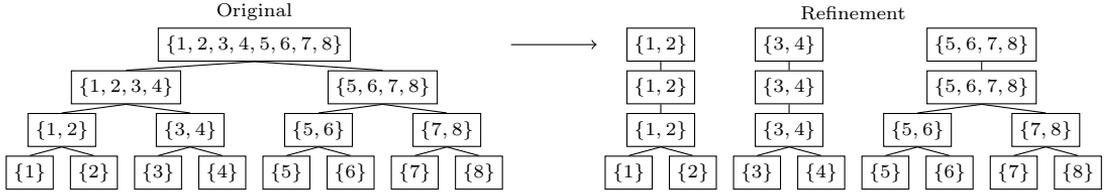
Evaluation of both the primal and dual objectives by candidate policies determined from scenario trees can gauge their representation of the underlying information structure in terms of the original objective function. While scenario set generation is mainly concerned with finding discrete outcomes, scenario trees additionally rely on effective representation of information structure. To our knowledge, this has not yet been explored in the context of scenario tree generation, but Kuhn (2008) used such primal-dual evaluation to gauge the effectiveness of decision rules (see Section 4.3). The advantage of primal-dual evaluation is that we get gap estimates of sub-optimality, providing a stopping criterion for sufficient approximation quality. In contrast, comparing upper bounds solely determines if one approximation approach is better than the other. Effective discrete representation of information structure tuned to the problem at hand is an interesting avenue of future research that currently has received little attention in the research literature on scenario tree generation.

## 4.2 Bounds on refinement of information structure

Refinement of information structure implies splitting a tree into smaller sub-trees and optimising decisions on these instead. This provides a collection of simpler optimisation problems whose solutions combine to provide a lower bound on the original problem. A conceptual illustration is given in Fig. II.8. This section presents a general setup that applies to various bounds in the literature derived from partitioning, group sub-problems, and sampling. However, note that some of these approaches involve combinatorial aspects not explained in detail. We also do not rely on the value of  $\xi$  to work out these bounds and, hence, refer to  $\omega$  instead.

Throughout this section, we assume a finite discrete distribution  $\tilde{P}$  with support  $\tilde{\Omega}$  such that the optimal objective value and solution set of  $\mathbb{E}_{\tilde{P}}[f(x(\omega), \xi(\omega))]$  is sufficiently close to those of  $\mathbb{E}_P[f(x(\omega), \xi(\omega))]$  (Shapiro et al., 2014, Section 7.2.5). We may, for example, interpret  $\tilde{P}$  as a sufficiently large sample from  $P$ . Sampling (in a stage-wise manner) or using some other discretisation procedure to obtain  $\tilde{P}$  is also the approach used in practice for computing these bounds.

The refinement is constructed by finding a (countable) collection of alternative distributions  $\{Q_k\}_{k \in \mathcal{K}}$  on subsets  $\{\Omega_k\}_{k \in \mathcal{K}}$  of  $\tilde{\Omega}$  and a convex combination of positive weights



**Figure II.8:** Example of refinement of information structure.

$\{\gamma_k\}_{k \in \mathcal{K}}$  (where  $\sum_{k \in \mathcal{K}} \gamma_k = 1$ ) such that

$$\tilde{P}(\omega) = \sum_{k \in \mathcal{K}} \gamma_k Q_k(\omega), \quad (\text{II.34})$$

where the alternative distributions compose the original distribution. Supports  $\Omega_k$  that are strictly smaller than  $\tilde{\Omega}$  is most interesting since this implies problems  $\mathbb{E}_{Q_k}[f(x, \xi)]$  are easier to solve than the original formulation using  $\tilde{P}$ . Probability assignment in terms of  $\{Q_k\}_{k \in \mathcal{K}}$  and subset selection in terms of  $\{\Omega_k\}_{k \in \mathcal{K}}$  are two alternative interpretations of how the distribution  $\tilde{P}$  is simplified, however, note that the relative assignment of probabilities within  $\Omega_k$  is solely determined by  $Q_k$ . By linearity of expectations in their probability assignment, we have that

$$\mathbb{E}_{\tilde{P}}[f(x(\omega), \xi(\omega))] = \sum_{k \in \mathcal{K}} \gamma_k \mathbb{E}_{Q_k}[f(x(\omega), \xi(\omega))], \quad (\text{II.35})$$

where the latter expression represents a convex combination of sub-problem evaluations. By minimising (II.35) with respect to decision  $x$  we obtain the lower bound

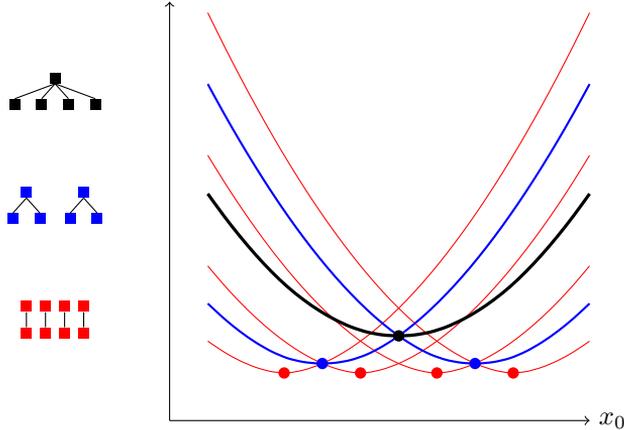
$$\min_{x(\omega) \in \mathcal{X}(\omega)} \{\mathbb{E}_{\tilde{P}}[f(x(\omega), \xi(\omega))]\} \geq \sum_{k \in \mathcal{K}} \gamma_k \min_{x(\omega) \in \mathcal{X}(\omega)} \{\mathbb{E}_{Q_k}[f(x(\omega), \xi(\omega))]\} \quad (\text{II.36})$$

by interchanging the order of optimisation and summation over  $\mathcal{K}$ . This also has the interpretation that the optimal value map  $\tilde{P} \mapsto v(\tilde{P})$  is concave in  $\tilde{P}$  (Maggioni & Pflug, 2016, Lemma 2.1). This is a simplification of the original formulation since we may solve smaller batches  $k \in \mathcal{K}$  of sub-problems instead of the full problem.

A consequence of solving the stochastic program in batches of sub-problems  $\mathbb{E}_{Q_k}[f(x(\omega), \xi(\omega))]$  is that this *relaxes the underlying information structure*, and leads to batch-specific optimal decisions that cannot be implemented in the original formulation. The fact that this leads to a lower bound can be illustrated conceptually (Fig. II.9) by considering how the evaluation of a convex combination of batch-specific objective functions  $\mathbb{E}_{Q_k}[f(x(\omega), \xi(\omega))]$  compares to an evaluation of each of them.

**Bounds based on partitioning** The first (and classic) example of refinement bounds is the expectation over all wait-and-see solutions (decisions are determined *after* uncertainty is revealed) by Madansky (1960). This follows from interchanging the order of minimisation and integration

$$\min_{x(\omega) \in \mathcal{X}(\omega)} \{\mathbb{E}_{\tilde{P}}[f(x(\omega), \xi(\omega))]\} \geq \mathbb{E}_{\tilde{P}} \left[ \min_{x(\omega) \in \mathcal{X}(\omega)} \{f(x(\omega), \xi(\omega))\} \right], \quad (\text{II.37})$$



**Figure II.9:** Illustration of different objective functions  $\mathbb{E}_{Q_k} [f(x(\omega), \xi(\omega))]$  in terms of first-stage decision  $x_0$  under refinement of information structure. Re-enforcing non-anticipativity in the first-stage decision corresponds to a convex combination of the refined objective functions. Observe that each objective function's minimiser (dots) changes according to the information structure.

where the sub-groups are considered to be the atoms of  $\tilde{\Omega}$ . Partitioning into larger subsets of  $\tilde{\Omega}$  than the atoms strictly strengthens this bound (Maggioni & Pflug, 2016, Section 2). We may also make a refinement chain of consecutive coarser partitions, where larger alternative distributions  $Q_k$  are convex combinations of smaller ones, to get a hierarchy of bounds that get progressively stronger.

**Bounds based on group sub-problems** We may instead define sub-groups of  $\Omega$  that can also intersect each other. The associated alternative distributions  $Q_k$  must still adhere to (II.34). Birge (1982) derived the first such result referred to as the sum of pairs sub-problems where a particular reference outcome  $\omega_r$  is paired with every other outcome such that  $\Omega_k = \{\omega_k, \omega_r\}$  for all  $k \neq r$ . The distributions  $Q_k(\omega_r)$  are adjusted such that the overall probability is consistently given as  $\tilde{P}(\omega_r) = \sum_k Q_k(\omega_r)$ . The reference outcome  $\omega_r$  here represents a particularly important outcome that the decision-maker always wants to account for. This grouping gives a lower bound on  $v(P)$  but is also an upper bound on the wait-and-see solution (II.37). The generalisation instead chooses  $\{\Omega_k^n\}_{k \in \mathcal{K}}$  to be all subsets of  $\tilde{\Omega}$  of a given size  $n$ , referred to as the *group sub-problems*. These give a lower bound for each size  $n$  that is monotonously non-decreasing in  $n$ , with edge cases that coincide with the wait-and-see problems and the original problem. The formulation such that the original problem is an edge case was derived by Sandıkçı et al. (2013) and Maggioni et al. (2014, 2016) for two-stage and multistage problems.

**Sampling based bounds** Monte Carlo sampling is commonly used as a viable option for solving intractable stochastic programs when the distribution is too large. Mak et al. (1999) showed that a sampled distribution  $Q_k^n$  of a given size  $n$  (which may represent historical data itself) provides a lower-bounding optimal objective value  $v(Q_k^n)$  in expectation over many samples of  $Q_k^n$ . This implies that solving a stochastic program on historical data gives, on average, a lower bounding objective value with respect to the underlying distribution. Furthermore, Mak et al. (1999, Theorem 2) show the lower bound is mono-

tonously increasing with the sample size  $n$ . In this setting, each alternative distribution  $Q_k^n$  represents a sampled empirical distribution of a given size, while the weighting  $\gamma_k$  represents the overall probability of obtaining this empirical distribution in particular. Over many sampled distributions  $Q_k^n$ , their expected relative occurrence is  $\gamma_k$ . Sample average approximation is analogous to group sub-problems where the relative weights  $\{\gamma_k\}_{k \in \mathcal{K}}$  have an additional sampling error, and potentially also that outcomes are defined to be equiprobable in each  $Q_k^n$  according to empirical distributions. There is much more to be said about solving stochastic programs with respect to sampled distributions, for which we refer to Bayraksan and Morton (2009) and Shapiro et al. (2014).

### 4.3 Bounds based on decision rules and aggregation

In their basic form, stochastic programs are infinite mathematical programs. Aggregation of variables and constraints were first introduced in finite (deterministic) linear programming as a simplification to make them smaller and easier to solve. In stochastic programming, however, aggregation turns an infinite linear program into a finite one by integrating (aggregating) the stochastic argument out of the problem, which is a considerable improvement (to say the least). In this context, we first choose a decision rule defined as a linear combination of basic rules that are predefined functions of the stochastic data process  $\xi$  (or  $\omega$ ). Inserting the decision rule in place of a general policy in the stochastic program allows evaluating expectations first and instead optimising over the finite number of coefficients in the linear combination. In contrast, a general policy must be optimised for all (infinite number of) outcomes.

Theory on aggregation bounds (that stem from deterministic linear programming) allows deriving bounds on the original infinite formulation of the stochastic program in terms of the solvable finite formulation. These bounds use arguments of how decision rules lead to relaxation, tightening or sub-optimality with respect to the original formulation. Primal-dual pairs give rise to upper and lower bounds according to weak duality. Constraint aggregation in the primal corresponds to variable aggregation in the dual, and conversely. Aggregation may also be done over stages in multistage formulations.

Aggregation bounds applied to stochastic programming started with Zipkin (1980a, 1980b) who analysed aggregation of variables and constraints for finite (deterministic) linear programs. Birge (1985) extended these results in the setting of stochastic programming by instead aggregating over uncertainty or stages assuming right-hand side uncertainty. Wright (1994) generalised this in a measure-theoretic framework using Lagrangian duality to general stochastic linear programs in the context of coarsened (simplified) information structures. More generally, this can be formulated by decision rules for which similar results apply. Decision rules have been successfully applied in the context of robust optimisation and were, for this reason, re-introduced to stochastic programming by Shapiro and Nemirovski (2005) as a means of tractable complexity reduction. Kuhn et al. (2011) have shown how to find solutions to stochastic linear programs where either the primal or the dual policy is replaced by decision rules (leading to semi-infinite programs), and aggregation bounds provide the optimality gap of these decision rules with respect to the original formulation.

#### Statement of decision rules

We state the formulation of decision rules and, for simplicity of exposition, we do this in the context of stochastic linear programs (SLP). Decision rules can be applied to replace the primal policy  $x(\omega)$ , the dual policy  $\lambda(\omega)$ , or both. Using decision rules in the

dual policy also has the alternative interpretation of constraint aggregation in the primal problem, and conversely.

Consider that we define decision rules  $\hat{x}(\omega)$  for primal decisions and  $\hat{\lambda}(\omega)$  for dual decisions. These are composed of linear combinations of simpler fixed policies

$$\hat{x}(\omega) = \sum_{k=1}^K w_k^x h_k^x(\omega), \quad \hat{\lambda}(\omega) = \sum_{l=1}^L w_l^\lambda h_l^\lambda(\omega), \quad (\text{II.38})$$

where  $h^x(\omega) := \{h_k^x(\omega)\}_{k=1,\dots,K}$  and  $h^\lambda(\omega) := \{h_l^\lambda(\omega)\}_{l=1,\dots,L}$  are predefined functions of  $\omega$ , while  $w^x$  and  $w^\lambda$  are vectors of  $K$  and  $L$  elements, respectively, that define the linear combinations. The functions  $h^x(\omega)$  and  $h^\lambda(\omega)$  must adhere to the same information structure  $\mathcal{F}$  and have the same dimensionality as the policies  $x$  and  $\lambda$  in the original formulation. Simplification from general policy to a decision rule effectively *limits its flexibility* since it is reduced to a linear combination of predefined functions. Furthermore, as noted by Kuhn (2008), inserting a decision rule for constraint multipliers corresponds to *more flexibility* in decisions.

To see how decision rules cause  $\omega$  to be aggregated out of the problem, consider the Lagrangian of (SLP)

$$L(x(\omega), \lambda(\omega)) = \mathbb{E}_P [c(\omega)^\top x(\omega) - \lambda(\omega)^\top (A(\omega)x(\omega) - b(\omega))], \quad (\text{II.39})$$

where inserting the decision rules  $\hat{x}(\omega), \hat{\lambda}(\omega)$  in place of the general policies  $x(\omega), \lambda(\omega)$  gives

$$L(\hat{x}(\omega), \hat{\lambda}(\omega)) = \hat{c}^\top w^x - (w^\lambda)^\top (\hat{A}w^x - \hat{b}), \quad (\text{II.40})$$

where

$$\hat{c}_k = \mathbb{E}_P [c(\omega)^\top h_k^x(\omega)], \quad \hat{A}_{lk} = \mathbb{E}_P [h_l^\lambda(\omega)^\top A(\omega) h_k^x(\omega)], \quad \hat{b}_l = \mathbb{E}_P [h_l^\lambda(\omega)^\top b(\omega)],$$

are the aggregated parameters. In the usual sense of linear programming, there are then finite primal and dual LPs associated with (II.40). Since the reliance on  $\omega$  is limited to predefined functions, evaluation of expectations can be performed immediately and aggregates  $\omega$  out of the problem. We may then proceed to optimise the decision rule with respect to the aggregated parameters and linear coefficients instead. Using decision rules either for variables or for constraint multipliers leads to aggregation bounds on the optimal value  $v(P)$  of the original formulation (see Section 4.3), which we illustrate in Section 4.3.

There is also a strong link between decision rules and function approximation (Section 3.4), where the only conceptual difference is a (sometimes trivial) evaluation of the objective function in terms of the decision rule. Consequently, decision rules and aggregation bounds have close analogies to those in Section 3.4.

### Example: Variable and constraint aggregation

Variable aggregation can be interpreted as replacing many variables with a single one. Consider the primal constraint set of (SLP),

$$A(\omega)x(\omega) \geq b(\omega), \quad \forall \omega \in \Omega,$$

where, for a single outcome, there is only a single constraint that the general policy  $x$  must fulfil in that outcome. If we instead use a decision rule  $\hat{x}(\omega) = w^x h^x(\omega) \geq 0$  and insert it into the constraint, we have that

$$w^x A(\omega) h^x(\omega) \geq b(\omega), \quad \forall \omega \in \Omega,$$

and the single coefficient  $w^x$  is subject to constraints for *all*  $\omega \in \Omega$ . Hence, the decision rule led to tightened constraints compared to the general policy because the flexibility to adjust to each outcome  $\omega$  has been limited. The decision rule also relies on the function  $h^x(\omega)$  that was chosen to decide how these constraints can be fulfilled.

Constraint aggregation can be interpreted as making a (weighted) sum of constraints. If we have two valid inequality constraints and add them together (possibly by positive weights), we obtain a new inequality constraint that must also be valid. If we only know that the summed constraint is valid, however, we cannot say anything about the validity of each constraint that composes it. This means the summation of constraints results in relaxation. To see why constraints are summed, consider that we use a decision rule  $\hat{\lambda}(\omega) = w^\lambda h^\lambda(\omega) \geq 0$  for the constraint multipliers in SLP. The restatement of SLP under Lagrangian relaxation of its constraints is

$$\min_{x(\omega) \geq 0} \mathbb{E}_P [c(\omega)^\top x(\omega) - \lambda(\omega)^\top (A(\omega)x(\omega) - b(\omega))],$$

and inserting the decision rule  $\hat{\lambda}(\omega)$  instead gives

$$\min_{x(\omega) \geq 0} \mathbb{E}_P [c(\omega)^\top x(\omega)] - w^\lambda \mathbb{E}_P [h^\lambda(\omega)^\top (A(\omega)x(\omega) - b(\omega))],$$

whose corresponding non-relaxed constraints are

$$\mathbb{E}_P [h^\lambda(\omega)^\top A(\omega)x(\omega)] \geq \mathbb{E}_P [h^\lambda(\omega)^\top b(\omega)],$$

with dual multiplier  $w^\lambda$ , and where the function  $h^\lambda(\omega)$  acts as weighting in the summation (integration) of constraints. This also means we replace many constraint multipliers with a single one, in analogy to variable aggregation. The relaxation follows from the fact that only the aggregated constraint must be fulfilled by  $x(\omega)$ , which gives it more flexibility.

### Aggregation bounds

We now consider using decision rules to replace either primal or dual policies to obtain bounds on the optimal objective value. Let  $\mathbb{P}(x, \lambda)$  denote the primal problem (SLP) with respect to the variables in the first argument and constraint multipliers in the second argument. General policies are denoted by  $x(\omega)$  and  $\lambda(\omega)$  while decision rules (II.38) inserted in their place are denoted by  $\hat{x}(\omega)$  and  $\hat{\lambda}(\omega)$ . For ease of readability, we omit their argument  $\omega$  in this section. Variable aggregation in the primal corresponds to constraint aggregation in the dual (as well as the converse), and so  $\mathbb{D}(\lambda, x)$  denotes the dual problem of  $\mathbb{P}(x, \lambda)$  with respect to the same arguments. Furthermore,

$$\inf_{x \in \mathcal{X}} \mathbb{P}(x, \lambda) \geq \sup_{\lambda \in \Lambda} \mathbb{D}(\lambda, x), \quad (\text{II.41})$$

by weak duality. Let  $\mathcal{X} = \text{feas}(\mathbb{P}(x, \lambda))$  and  $\Lambda = \text{feas}(\mathbb{D}(\lambda, x))$  denote the feasible region of decisions in  $\mathbb{P}(x, \lambda)$  and  $\mathbb{D}(\lambda, x)$ , respectively. Variable aggregation tightens the feasible region, while constraint aggregation relaxes it. We then have the relations,

$$\text{feas}(\mathbb{P}(\hat{x}, \lambda)) \subseteq \frac{\text{feas}(\mathbb{P}(x, \lambda))}{\text{feas}(\mathbb{P}(\hat{x}, \hat{\lambda}))} \subseteq \text{feas}(\mathbb{P}(x, \hat{\lambda})), \quad (\text{II.42})$$

and

$$\text{feas}(\mathbb{D}(\lambda, \hat{x})) \supseteq \frac{\text{feas}(\mathbb{D}(\lambda, x))}{\text{feas}(\mathbb{D}(\hat{\lambda}, \hat{x}))} \supseteq \text{feas}(\mathbb{D}(\hat{\lambda}, x)). \quad (\text{II.43})$$

Observe that inserting one kind of decision rule at a time gives simple relations between feasible regions, but there is no simple relation between the original and the fully simplified problems. We have the following relations between objective values

$$\mathbb{P}(\hat{x}, \lambda) \geq \inf \mathbb{P}(\hat{x}, \lambda) \geq \frac{\inf \mathbb{P}(x, \lambda) \geq \sup \mathbb{D}(\lambda, x)}{\inf \mathbb{P}(\hat{x}, \hat{\lambda}) \geq \sup \mathbb{D}(\hat{\lambda}, \hat{x})} \geq \sup \mathbb{D}(\hat{\lambda}, x) \geq \mathbb{D}(\hat{\lambda}, x), \quad (\text{II.44})$$

where  $\mathbb{P}(\hat{x}, \lambda)$  and  $\mathbb{D}(\hat{\lambda}, x)$  denote objective evaluation by a feasible decision policy.

The underlying formulations  $\mathbb{P}(x, \lambda)$  and  $\mathbb{D}(\lambda, x)$  are considered to be unsolvable, but bounds on these can be derived using decision rules according to the inequalities (II.44). Consider also that a low gap between these bounds means the predefined form of decision rules (II.38) are effective for the original formulation. Indeed, there exist more or less appropriate functions  $h^x(\omega)$  and  $h^\lambda(\omega)$  to obtain good results with low optimality gaps.

Tractable reformulations of  $\inf \mathbb{P}(\hat{x}, \lambda)$  and  $\sup \mathbb{D}(\hat{\lambda}, x)$  into linear and semi-definite programs of moderate sizes that can be solved directly have been laid out by Kuhn et al. (2011). Alternatively, we may solve the finite formulations  $\mathbb{P}(\hat{x}, \hat{\lambda})$  and  $\mathbb{D}(\hat{\lambda}, \hat{x})$  to obtain candidate primal  $\tilde{x}(\omega)$  and dual policies  $\tilde{\lambda}(\omega)$  and evaluate these within the objectives of  $\mathbb{P}(\hat{x}, \lambda)$  and  $\mathbb{D}(\hat{\lambda}, x)$  to get bounds on  $v(P)$ . This procedure is also described in Section 4.1 regarding scenario trees and extension rules.

## 5 Discussion and conclusions

Stochastic programming is applied to make effective decisions for real-world problems under uncertainty. While there is something fundamentally intractable in their formulations, bounds can give valuable insights into how well a given solution performs within the original formulation, or bounding distributions can be used as approximations in themselves.

The insight that function approximation and distribution approximation are tightly connected is a profound observation. This stems from the duality between functions and probability distributions, a connection seen throughout the paper. These perspectives were used for different approaches to evaluation bounds in Section 3, but also have close analogies to different ways of finding approximate solutions to stochastic programs either by scenario trees (distributions) or decision rules (functions) in Section 4.

The fundamental insight of evaluation bounds in Section 3 is that finite approximate distributions can be used to obtain bounds, even if the underlying distribution is continuous. This is a consequence of the generalised moment problem, solved by discrete distributions whose moments coincide with the underlying distribution. Furthermore, by the assumption of convexity in terms of the stochastic argument  $\xi$ , we may find analytical expressions for discrete distributions that provide guaranteed bounds. Bounds on optimal objective values in Section 4 also complement the evaluation bounds in Section 3. This is valuable both for applying and developing methods to solve such problems.

### Discussion: Bounds and modelling

Counter-intuitive mistakes may occur when one aims to make an approximation of uncertainty while actually getting bounds. These insights are especially relevant when applying models to decision problems under uncertainty, even if one does not solve a stochastic program.

An intuitive way to make a deterministic approximation to uncertainty is to insert the expected value of uncertain parameters. While the expectation of the stochastic phenomenon (i.e.  $\xi$ ) can be used as a best estimate for the phenomenon itself, stochastic programming deals with a non-linear transformation of the stochastic variables through the objective function, and what we really aim to find is the *expected objective value*. Hence, caution is due. For objective functions that are convex in the stochastic parameters, using the expected outcome of the phenomenon instead gives a lower bound (Section 3.3). Not only will this give a bound, but it will also be optimistic, the opposite of what we might want. More worryingly, an objective function that is convex-concave in uncertain parameters may lead to conservatism with respect to some parameters and optimism with respect to others, implicitly making a priority.

Consider a two-stage production planning problem as an illustration. The objective of the second-stage program may then be convex in its uncertain capacity and concave in the uncertain cost of materials, while additional operational considerations also limit flexibility in production. Using only the expected value means we overestimate the utility of additional capacity, given that operational considerations may cause other bottlenecks. We also underestimate how flexibility in production can counteract increased material costs by scaling down the production of products that turn out unprofitable. This means we are optimistic with respect to capacity and pessimistic with respect to the cost of materials.

Another intuitive way of approximating uncertainty is to plan for a restricted small set of outcomes instead of (approximately) every outcome, like sampling from the distribution. As explored in Section 4.2, considering only a restricted subset of outcomes at a time (but overall considering all of them) will, in combination, lead to a lower bound on the optimal objective value. Again, this is optimistic. One interpretation of this is that the optimal decision overestimates its availability of information and is less prepared for outcomes that were not explicitly accounted for. The recommended counter-measure in such situations is always to do candidate evaluation (Section 4.1) after a solution has been found to get a conservative estimate of how well a given decision or policy really performs in terms of the original formulation.

Scenario analysis (analysing decisions for one outcome at a time) is another case of apparent approximation of uncertainty that does not really build consideration of uncertainty into decisions. The issue is that considering a single outcome, we get the impression of knowing the future perfectly when it is actually unknowable (by definition of it being stochastic). Considering a single outcome at a time will *not* lead to decisions that buy options and enable flexibility to deal with multiple potential outcomes of the future, and are thus qualitatively different (Wallace, 2010). The gap to the wait-and-see lower bound in Section 4.2 quantifies the error that results from scenario analysis.

## Future research

A significant challenge for stochastic programming is how to solve multistage problems in a tractable manner. Using bounds to validate solutions can be valuable for further development in this domain. Approximations in the form of scenario trees or decision rules can potentially provide reasonable solutions by parsimonious representations but do not necessarily give indications of optimality or correct estimates of its optimal objective value. This is why bounds can help support their development.

Evaluation bounds (Section 3) have received much attention in the past while, more recently, the literature has shifted towards approximations in the form of scenario generation. Both are concerned with discretising distributions, but scenario generation has an

added layer of complication since it primarily concerns the quality of solutions obtained from *solving* a stochastic program with a given discrete distribution. The property of convex stochastic dominance (Section 3.2) conserves bounds for all decisions, which is an interesting link between these. Still, we believe there are synergies to be explored between bounds and scenario generation that can provide valuable insights. The concept of effective and ineffective scenarios developed in the setting of distributionally robust optimisation (Rahimian et al., 2019, 2022) may also provide further such connections.

There has been much recent research on risk aversion and aversion to ambiguity in the representation of uncertainty itself (not knowing the distribution). These are valuable tools to build conservatism into decisions and are tightly linked to bounds in the sense that they are often formulated to minimise an upper bound on the objective evaluation, subject to ambiguity (more general than moment conditions). These problems are challenging to solve (even more so for multistage problems), and quantifying interval estimates to their optimal solution is valuable to improve these approaches further and to give valuable insights for real applications. The literature is sparse in this domain but growing. Future work could also include using bounds from Section 4 within solution algorithms.

Table II.1: Distribution approximation bounds.

Name	Bound type	Multivariate dependency	Generalised Moment Function	Support	$\# \text{supp}(Q^*)$	Convex Dominance	Contributor(s)
Jensen's inequality	Lower	Yes	First-order	Unbounded	1	Yes	Jensen (1906)
Edmund-Madansky	Upper	No	First-order	Rectangular	$2^d$	Yes	Edmundson (1957) and Madansky (1959)
Edmund-Madansky with dependence	Upper	Yes	Multilinear	Rectangular	$2^d$	Yes	Frauendorfer (1988) and Kall (1987)
Hyper-plane	Upper	Yes	First-order	Polytopal	$\geq d + 1$	No	Birge and Wets (1986)
Hyper-plane	Upper	Yes	First-order	Polyhedral cone	$\geq d + 1$	No	Gassmann and Ziemba (1986)
Hyper-plane	Upper	Yes	First-order	Simplicial	$d + 1$	Yes	Birge and Wets (1986) and Gassmann and Ziemba (1986)
Barycentric approximation	Upper	Yes	Barycentric	Polytopal	$\geq d + 1$	Yes	Birge and Wets (1986) and Frauendorfer (1992)
Convex-concave saddle	Upper/Lower	No	First- and cross-order	Rectangular	$2^d$	Yes	Frauendorfer (1989) and Frauendorfer (1992)
Convex-concave saddle	Upper/Lower	Yes	First- and cross-order	Simplicial	$d + 1$	Yes	Frauendorfer (1989) and Frauendorfer (1992)
Convex-concave saddle	Upper/Lower	Yes	First- and cross-order	Polyhedral cone	$\geq d + 1$	No	Edirisinghe and Ziemba (1994a, 1994b, 1996)
Second-order lower bound	Lower	Yes	Second-order	Simplicial	$d + 1$	Yes	Edirisinghe (1996)
Second-order lower bound	Lower	Yes	Second-order	Rectangular	$2^d$	Yes	Edirisinghe (1996)
Second-order lower bound	Lower	No	Second-order	Rectangular	$\geq 2^d$	Yes	Dokov and Morton (2005)
Higher-order bound	Upper	Yes	$n$ th-order	Rectangular	$\binom{d+n}{n}$	Yes	Dokov and Morton (2002)

## References

- Bayraksan, G., & Morton, D. P. (2009, September). Assessing Solution Quality in Stochastic Programs via Sampling. In *Decision Technologies and Applications* (pp. 102–122). INFORMS. <https://doi.org/10.1287/educ.1090.0065>
- Ben-Tal, A., & Hochman, E. (1972). More Bounds on the Expectation of a Convex Function of a Random Variable. *Journal of Applied Probability*, 9(4), 803–812. <https://doi.org/10.2307/3212616>
- Bertsimas, D., & Tsitsiklis, J. (1997). *Introduction to Linear Programming*. Athena Scientific.
- Birge, J. R. (1982). The value of the stochastic solution in stochastic linear programs with fixed recourse. *Mathematical Programming*, 24(1), 314–325. <https://doi.org/10.1007/BF01585113>
- Birge, J. R. (1985). Aggregation bounds in stochastic linear programming. *Mathematical Programming*, 31(1), 25–41. <https://doi.org/10.1007/BF02591859>
- Birge, J. R., & Dulá, J. H. (1991). Bounding separable recourse functions with limited distribution information. *Annals of Operations Research*, 30(1), 277–298. <https://doi.org/10.1007/BF02204821>
- Birge, J. R., & Louveaux, F. (2011). *Introduction to stochastic programming* (2nd ed.). Springer New York, NY. <https://doi.org/10.1007/978-1-4614-0237-4>
- Birge, J. R., & Wallace, S. W. (1988). A Separable Piecewise Linear Upper Bound for Stochastic Linear Programs. *SIAM Journal on Control and Optimization*, 26(3), 725–739. <https://doi.org/10.1137/0326042>
- Birge, J. R., & Wets, R. J. -B. (1989). Sublinear upper bounds for stochastic programs with recourse. *Mathematical Programming*, 43(1), 131–149. <https://doi.org/10.1007/BF01582286>
- Birge, J. R., & Wets, R. J.-B. (1986). Designing approximation schemes for stochastic optimization problems, in particular for stochastic programs with recourse. In A. Prékopa & R. J. B. Wets (Eds.), *Stochastic Programming 84 Part I* (pp. 54–102). Springer. <https://doi.org/10.1007/BFb0121114>
- Birge, J. R., & Wets, R. J.-B. (1987). Computing Bounds for Stochastic Programming Problems by Means of a Generalized Moment Problem. *Mathematics of Operations Research*, 12(1), 149–162. <https://doi.org/10.1287/moor.12.1.149>
- Crainic, T. G., Maggioni, F., Perboli, G., & Rei, W. (2018). Reduced cost-based variable fixing in two-stage stochastic programming. *Annals of Operations Research*. <https://doi.org/10.1007/s10479-018-2942-8>
- Dokov, S. P., & Morton, D. P. (2002, May). *Higher-Order Upper Bounds on the Expectation of a Convex Function*. Humboldt-Universität zu Berlin, Mathematisch-Naturwissenschaftliche Fakultät II, Institut für Mathematik. <https://doi.org/10.18452/8272>  
Accepted: 2017-06-16T19:47:45Z.
- Dokov, S. P., & Morton, D. P. (2005). Second-Order Lower Bounds on the Expectation of a Convex Function. *Mathematics of Operations Research*, 30(3), 662–677. <https://doi.org/10.1287/moor.1040.0136>
- Dulá, J. H. (1992). An upper bound on the expectation of simplicial functions of multivariate random variables. *Mathematical Programming*, 55(1), 69–80. <https://doi.org/10.1007/BF01581191>
- Dulá, J. H., & Murthy, R. V. (1992). A Tchebysheff-Type Bound on the Expectation of Sublinear Polyhedral Functions. *Operations Research*, 40(5), 914–922. <https://doi.org/10.1287/opre.40.5.914>
- Dupačová, J. (1966). On minimax solutions of stochastic linear programming problems. *Časopis pro pěstování matematiky*, 091(4), 423–430. Retrieved November 7, 2022, from <https://eudml.org/doc/20949>
- Dupačová, J. (1987). The minimax approach to stochastic programming and an illustrative application. *Stochastics*, 20(1), 73–88. <https://doi.org/10.1080/17442508708833436>

- Edirisinghe, N. C. P. (1996). New Second-Order Bounds on the Expectation of Saddle Functions with Applications to Stochastic Linear Programming. *Operations Research*, 44(6), 909–922. <https://doi.org/10.1287/opre.44.6.909>
- Edirisinghe, N. C. P., & Ziemba, W. T. (1992). Tight Bounds for Stochastic Convex Programs. *Operations Research*, 40(4), 660–677. <https://doi.org/10.1287/opre.40.4.660>
- Edirisinghe, N. C. P., & Ziemba, W. T. (1994a). Bounds for Two-Stage Stochastic Programs with Fixed Recourse. *Mathematics of Operations Research*, 19(2), 292–313. Retrieved November 8, 2022, from <https://www.jstor.org/stable/3690222>
- Edirisinghe, N. C. P., & Ziemba, W. T. (1994b). Bounding the Expectation of a Saddle Function with Application to Stochastic Programming. *Mathematics of Operations Research*, 19(2), 314–340. <https://doi.org/10.1287/moor.19.2.314>
- Edirisinghe, N. C. P., & Ziemba, W. T. (1996). Implementing bounds-based approximations in convex-concave two-stage stochastic programming. *Mathematical Programming*, 75(2), 295–325. <https://doi.org/10.1007/BF02592157>
- Edmundson, H. P. (1957). *Bounds on the expectation of a convex function of a random variable* (tech. rep.). RAND Corporation. Santa Monica, CA.
- Frauentorfer, K. (1989). *A simplicial approximation scheme for convex two-stage stochastic programming problems* (tech. rep.). University of Zürich. Zürich.
- Frauentorfer, K. (1988). Solving SLP Recourse Problems with Arbitrary Multivariate Distributions—The Dependent Case. *Mathematics of Operations Research*, 13(3), 377–394. <https://doi.org/10.1287/moor.13.3.377>
- Frauentorfer, K. (1992). Barycentric Approximation. In K. Frauentorfer (Ed.), *Stochastic Two-Stage Programming* (pp. 67–123). Springer. [https://doi.org/10.1007/978-3-642-95696-6\\_4](https://doi.org/10.1007/978-3-642-95696-6_4)
- Frauentorfer, K. (1994). Multistage stochastic programming: Error analysis for the convex case. *Zeitschrift für Operations Research*, 39(1), 93–122. <https://doi.org/10.1007/BF01440737>
- Frauentorfer, K. (1996). Barycentric scenario trees in convex multistage stochastic programming. *Mathematical Programming*, 75(2), 277–293. <https://doi.org/10.1007/BF02592156>
- Frauentorfer, K., Kuhn, D., & Schürle, M. (2011). Barycentric Bounds in Stochastic Programming: Theory and Application. In G. Infanger (Ed.), *Stochastic Programming: The State of the Art In Honor of George B. Dantzig* (pp. 67–96). Springer. [https://doi.org/10.1007/978-1-4419-1642-6\\_5](https://doi.org/10.1007/978-1-4419-1642-6_5)
- Gassmann, H., & Ziemba, W. T. (1986). A tight upper bound for the expectation of a convex function of a multivariate random variable. In A. Prékopa & R. J. B. Wets (Eds.), *Stochastic Programming 84 Part I* (pp. 39–53). Springer. <https://doi.org/10.1007/BFb0121113>
- Huang, C. C., Ziemba, W. T., & Ben-Tal, A. (1977). Bounds on the Expectation of a Convex Function of a Random Variable: With Applications to Stochastic Programming. *Operations Research*, 25(2), 315–325. Retrieved November 15, 2022, from <https://www.jstor.org/stable/169834>
- Jensen, J. L. W. V. (1906). Sur les fonctions convexes et les inégalités entre les valeurs moyennes. *Acta Mathematica*, 30(none), 175–193. <https://doi.org/10.1007/BF02418571>
- Kall, P. (1987). Stochastic programs with recourse: An upper bound and the related moment problem. *Zeitschrift für Operations Research*, 31(3), A119–A141. <https://doi.org/10.1007/BF01259340>
- Kall, P. (1988). Stochastic programming with recourse: Upper bounds and moment problems - a review. In *Advances in Mathematical Optimization* (pp. 86–103). De Gruyter.
- Kall, P. (1991). An upper bound for SLP using first and total second moments. *Annals of Operations Research*, 30(1), 267–276. <https://doi.org/10.1007/BF02204820>
- Kall, P., & Wallace, S. W. (1994). *Stochastic Programming*. John Wiley & Sons.
- Karr, A. F. (1983). Extreme Points of Certain Sets of Probability Measures, with Applications. *Mathematics of Operations Research*, 8(1), 74–85. Retrieved November 24, 2022, from <https://www.jstor.org/stable/3689412>

- Kaut, M., & Wallace, S. W. (2007). Evaluation of scenario-generation methods for stochastic programming. *Pacific Journal of Optimization*, 3(2), 257–271.
- Kemperman, J. H. B. (1968). The General Moment Problem, A Geometric Approach. *The Annals of Mathematical Statistics*, 39(1), 93–122. Retrieved November 21, 2022, from <https://www.jstor.org/stable/2238913>
- Keutchan, J., Gendreau, M., & Saucier, A. (2017). Quality evaluation of scenario-tree generation methods for solving stochastic programming problems. *Computational Management Science*, 14(3), 333–365. <https://doi.org/10.1007/s10287-017-0279-4>
- King, A. J., & Wallace, S. W. (2012, June). *Modeling with Stochastic Programming*. Springer New York, NY. <https://doi.org/10.1007/978-0-387-87817-1>
- Klein Haneveld, W. K. (1986). Some Linear Programs in Probabilities and Their Duals. In W. K. Klein Haneveld (Ed.), *Duality in Stochastic Linear and Dynamic Programming* (pp. 49–111). Springer. [https://doi.org/10.1007/978-3-642-51697-9\\_4](https://doi.org/10.1007/978-3-642-51697-9_4)
- Kuhn, D. (2005). Barycentric Approximation Scheme. In D. Kuhn, M. Beckmann, H. P. Künzi, G. Fandel, W. Trockel, A. Basile, A. Drexl, H. Dawid, K. Inderfurth, W. Kürsten & U. Schittko (Eds.), *Generalized Bounds for Convex Multistage Stochastic Programs* (pp. 51–81). Springer. [https://doi.org/10.1007/3-540-26901-0\\_4](https://doi.org/10.1007/3-540-26901-0_4)
- Kuhn, D. (2008). Aggregation and discretization in multistage stochastic programming. *Mathematical Programming*, 113(1), 61–94. <https://doi.org/10.1007/s10107-006-0048-6>
- Kuhn, D., Wiesemann, W., & Georgiou, A. (2011). Primal and dual linear decision rules in stochastic and robust optimization. *Mathematical Programming*, 130(1), 177–209. <https://doi.org/10.1007/s10107-009-0331-4>
- Madansky, A. (1959). Bounds on the Expectation of a Convex Function of a Multivariate Random Variable. *The Annals of Mathematical Statistics*, 30(3), 743–746. <https://doi.org/10.1214/aoms/1177706203>
- Madansky, A. (1960). Inequalities for Stochastic Linear Programming Problems. *Management Science*, 6(2), 197–204. <https://doi.org/10.1287/mnsc.6.2.197>
- Maggioni, F., Allevi, E., & Bertocchi, M. (2014). Bounds in Multistage Linear Stochastic Programming. *Journal of Optimization Theory and Applications*, 163(1), 200–229. <https://doi.org/10.1007/s10957-013-0450-1>
- Maggioni, F., Allevi, E., & Bertocchi, M. (2016). Monotonic bounds in multistage mixed-integer stochastic programming. *Computational Management Science*, 13(3), 423–457. <https://doi.org/10.1007/s10287-016-0254-5>
- Maggioni, F., & Pflug, G. C. (2016). Bounds and Approximations for Multistage Stochastic Programs. *SIAM Journal on Optimization*, 26(1), 831–855. <https://doi.org/10.1137/140971889>
- Maggioni, F., & Pflug, G. C. (2019). Guaranteed Bounds for General Nondiscrete Multistage Risk-Averse Stochastic Optimization Programs. *SIAM Journal on Optimization*, 29(1), 454–483. <https://doi.org/10.1137/17M1140601>
- Maggioni, F., & Wallace, S. W. (2012). Analyzing the quality of the expected value solution in stochastic programming. *Annals of Operations Research*, 200(1), 37–54. <https://doi.org/10.1007/s10479-010-0807-x>
- Mak, W.-K., Morton, D. P., & Wood, R. (1999). Monte Carlo bounding techniques for determining solution quality in stochastic programs. *Operations Research Letters*, 24(1-2), 47–56. [https://doi.org/10.1016/S0167-6377\(98\)00054-6](https://doi.org/10.1016/S0167-6377(98)00054-6)
- Morton, D. P., & Wood, R. K. (1999). Restricted-Recourse Bounds for Stochastic Linear Programming. *Operations Research*, 47(6), 943–956. <https://doi.org/10.1287/opre.47.6.943>
- Pereira, M. V. F., & Pinto, L. M. V. G. (1991). Multi-stage stochastic optimization applied to energy planning. *Mathematical Programming*, 52(1-3), 359–375. <https://doi.org/10.1007/BF01582895>
- Pflug, G. C., & Pichler, A. (2012). A Distance For Multistage Stochastic Optimization Models. *SIAM Journal on Optimization*, 22(1), 1–23. <https://doi.org/10.1137/110825054>
- Pflug, G. C., & Pichler, A. (2014). *Multistage Stochastic Optimization*. Springer International Publishing. <https://doi.org/10.1007/978-3-319-08843-3>

- Powell, W. B., & Frantzeskakis, L. F. (1994). Restricted Recourse Strategies for Dynamic Networks with Random Arc Capacities. *Transportation Science*, 28(1), 3–23. <https://doi.org/10.1287/trsc.28.1.3>
- Rahimian, H., Bayraksan, G., & De-Mello, T. H. (2022). Effective Scenarios in Multistage Distributionally Robust Optimization with a Focus on Total Variation Distance. *SIAM Journal on Optimization*, 32(3), 1698–1727. <https://doi.org/10.1137/21M1446484>
- Rahimian, H., Bayraksan, G., & Homem-de-Mello, T. (2019). Identifying effective scenarios in distributionally robust stochastic programs with total variation distance. *Mathematical Programming*, 173(1-2), 393–430. <https://doi.org/10.1007/s10107-017-1224-6>
- Sandıkçı, B., Kong, N., & Schaefer, A. J. (2013). A hierarchy of bounds for stochastic mixed-integer programs. *Mathematical Programming*, 138(1-2), 253–272. <https://doi.org/10.1007/s10107-012-0526-y>
- Shapiro, A., Dentcheva, D., & Ruszczyński, A. P. (2014). *Lectures on stochastic programming: Modeling and theory* (Second edition). Society for Industrial and Applied Mathematics : Mathematical Optimization Society. <https://doi.org/10.1137/1.9781611973433>
- Shapiro, A., & Nemirovski, A. (2005). On Complexity of Stochastic Programming Problems. In V. Jeyakumar & A. Rubinov (Eds.), *Continuous Optimization: Current Trends and Modern Applications* (pp. 111–146). Springer US. [https://doi.org/10.1007/0-387-26771-9\\_4](https://doi.org/10.1007/0-387-26771-9_4)
- Thaler, R. H. (1988). Anomalies: The Winner’s Curse. *Journal of Economic Perspectives*, 2(1), 191–202. <https://doi.org/10.1257/jep.2.1.191>
- Wallace, S. W. (1987). A piecewise linear upper bound on the network recourse function. *Mathematical Programming*, 38(2), 133–146. <https://doi.org/10.1007/BF02604638>
- Wallace, S. W. (2010). Stochastic programming and the option of doing it differently. *Annals of Operations Research*, 177(1), 3–8. <https://doi.org/10.1007/s10479-009-0600-x>
- Wets, R. J. -. (1984). Modeling and solution strategies for unconstrained stochastic optimization problems. *Annals of Operations Research*, 1(1), 3–22. <https://doi.org/10.1007/BF01874449>
- Wright, S. E. (1994). Primal-Dual Aggregation and Disaggregation for Stochastic Linear Programs. *Mathematics of Operations Research*, 19(4), 893–908. Retrieved December 10, 2022, from <https://www.jstor.org/stable/3690318>
- Xu, J., & Sen, S. (2023). Compromise policy for multi-stage stochastic linear programming: Variance and bias reduction. *Computers & Operations Research*, 153, 106132. <https://doi.org/10.1016/j.cor.2022.106132>
- Zipkin, P. H. (1980a). Bounds for Row-Aggregation in Linear Programming. *Operations Research*, 28(4), 903–916. Retrieved February 7, 2023, from <https://www.jstor.org/stable/170330>
- Zipkin, P. H. (1980b). Bounds on the Effect of Aggregating Variables in Linear Programs. *Operations Research*, 28(2), 403–418. <https://doi.org/10.1287/opre.28.2.403>

# Paper III

## Joint Forecasting of Salmon Lice and Treatment Interventions

Benjamin S. Narum and Geir D. Berentsen

### Abstract

The need for joint forecasting of parasitic lice and associated preventative treatments stems from large monetary losses associated with such treatments, and the distribution of potential future treatments can be used in operational planning to hedge their associated risk. We present a spatio-temporal forecasting model that accounts for the joint dynamics between lice and treatments where spatial interaction between sites is derived from hydrodynamic transportation patterns. The model-derived forecasting distributions exhibit large heterogeneity between sites at significant levels of exposure which suggests the forecasting model can provide great value in assisting operational risk management.

## 1 Introduction

The salmon louse is a parasite that lives off Salmonids (salmon and trout) and spreads between hosts in the ocean waters. Salmonid aquaculture consists of breathing fish in pens in the ocean, and the large expansion of the industry has increased the density of hosts for the salmon louse. This, in turn, has resulted in higher levels of lice infection, causing concern about the welfare of both farmed and wild Salmonids. As a mitigation strategy, farmers apply removal treatments when they observe high levels of lice abundance, and such treatments contribute significantly to increased fish mortality (Bang Jensen et al., 2020; Kristoffersen et al., 2018; Walde et al., 2021). Conservative estimates suggest that lice treatments result in costs equivalent to approximately 9% of annual revenues in the Norwegian industry (Abolofia et al., 2017; Iversen et al., 2017), where the indirect costs of increased mortality and loss of growth (Walde et al., 2022) are important contributions. This paper addresses joint forecasting of lice development and associated treatment interventions for the purpose of assisting operational harvest planning.

Knowing the exposure to future lice treatments is important for farmers to schedule their harvest plans and manage the associated biological risks. While higher levels of lice abundance is the reason for performing treatments, most of the costs and biological risks

are directly associated to the treatments themselves. Since we aim to apply forecasts to operational risk management, the target forecast quantity is the *number of future treatments*. At the levels of lice abundance normally found at aquaculture sites, the biological risk from the parasite itself is comparably low due to the high frequency of treatments.

Motivated by the application, we model treatments as a stochastic process that interacts with another stochastic process for lice abundance. While treatment is a decision made by farmers, there are strong limitations in the flexibility of this decision since farmers are obliged to treat (by regulation) based on measured levels of lice abundance. Thus, we argue that the prospective frequency of future treatments (which infers risk to farmers) is mainly driven by its interaction with lice abundance, and only moderately by the specific timing that farmers decide. The time of treatment must lie within a short time-range ( $\pm 1$  week) once higher levels of lice abundance have been observed. Keep in mind that we aim to infer the long-term treatment count (in the order of 20 weeks ahead) and not necessarily the next treatment. If farmers have short-term plans for treatment (1 week ahead), these may be inserted as explanatory variables before making longer-term forecasts. Modelling treatments as a stochastic process is also motivated by multistage stochastic optimisation models used for operational risk management (see King & Wallace, 2012; Shapiro et al., 2014). These take a distributional forecast as input and aim to find plans that hedge operational risks effectively. These optimisation problems are potentially very computationally challenging to solve, and letting treatments be a decision introduces decision-dependent uncertainty (where decisions affect the distribution) (Hellemo et al., 2018; Jonsbråten et al., 1998). This makes them potentially intractable to solve within reasonable time. A way to mitigate this complexity is to exogenise treatment decisions by modelling these as a stochastic process instead.

Existing studies on the dynamics and spread of lice can broadly be divided into two categories. In the first category are statistical models used to draw inference about the dynamics of lice or to obtain site-specific forecasts (Aldrin et al., 2017, 2019; Aldrin et al., 2013; Elghafghuf et al., 2018, 2020). Some of these models also incorporate spatial dependence by considering seaway distance (along the ocean surface) between sites. In the second category is a physical model that uses simulation of hydrodynamic stream patterns combined with historical lice abundance data to infer a measure of "lice infestation pressure" along the coastline (Myksvoll et al., 2018). The caveat of the latter model is that it cannot forecast future lice counts since the link between infestation pressure and on-site dynamics of lice abundance has not been accounted for. Our approach to modelling lice abundance bridges the gap between these two approaches by introducing asymmetric spatial effects derived from stream simulation data (as opposed to the symmetric seaway distance) in a time-series model that links infestation pressure to the on-site dynamics of lice abundance. Moreover, while existing approaches consider lice treatments as an exogenous explanatory variable, we consider lice abundance and treatments as two stochastic processes that affect each other. To our knowledge, this paper is the first to incorporate stream patterns into on-site lice dynamics as well as its interaction with future treatments.

The paper is structured as follows: Section 2 presents all data sources and explains the basics of lice development; Section 3 presents the forecasting models for lice and treatments, with details on joint forecasting and estimation; Section 4 presents results and model validations; finally, Section 5 is a discussion with concluding remarks.

## 2 Data

We limit sites to those in the Norwegian production area 3, situated on the western coast of Norway, which is the largest and known for having the most activity of lice. Sites outside of this area are incorporated only by their contribution to neighbour infection. There are 137 sites in total that are active for 41572 site-weeks combined within the period June 2012 to September 2021. An overview of these sites is illustrated in Figure III.1.

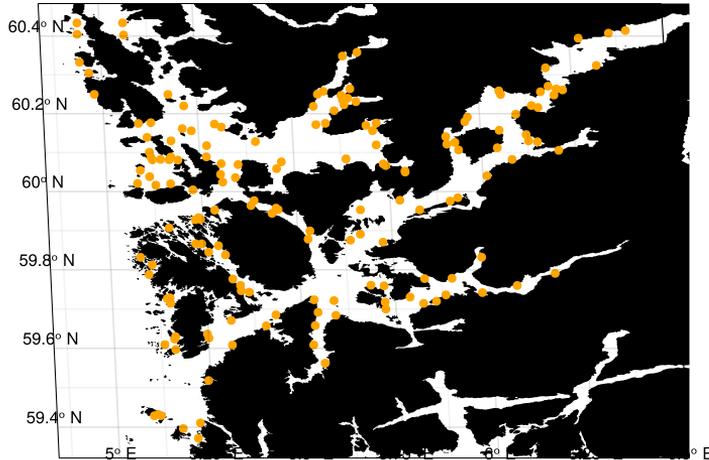


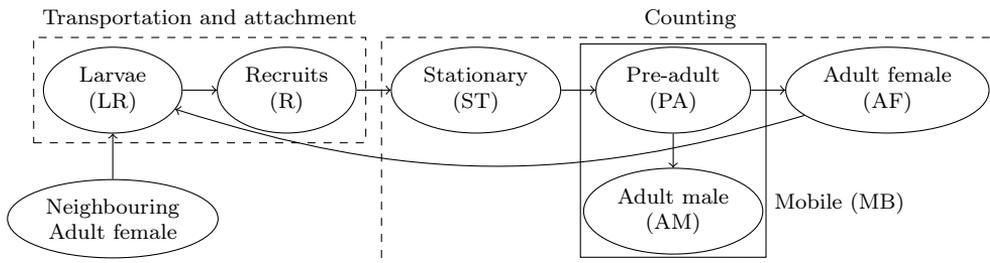
Figure III.1: Overview of aquaculture sites in production area 3.

### Lice and treatment data

Biologically, lice develop through multiple stages (Hamre et al., 2013) that we simplify into categories relevant to our model. See the illustration in Figure III.2. The development cycle starts by hatching larvae (LR) from egg strings attached to adult female (AF) lice. The larvae are then transported by ocean currents and may find a new host (a Salmonid). During transportation the louse may attach to a new host, then referred to as a *recruit* (R). If it cannot find a new host in time, it runs out of nutrition and dies. Note that the recruit may also originate from the same site. The recruit develops through three stages on the host: stationary (ST); pre-adult (PA), where it starts to move on the host; and lastly, adult female/male (AF/AM), where the female develops egg strings to hatch new larval lice (LR) that repeat the cycle. Keep in mind that this development cycle is highly temperature dependent and that males develop slightly faster than female lice (Hamre et al., 2019). Also note that pre-adult (PA) and adult male (AM) lice are visually indistinguishable and are thus only identified as mobile lice (MB).

By regulation, all Norwegian farmers are obliged to count lice in the three identifiable categories (ST, MB and AF) weekly on 10 or 20 fish (depending on the time of year and location) in each production cage (Lovdata, 2012). Typically, a single cage has approximately 200 000 fish in total and the lice count on each fish sample is averaged over all cages at a site (1–12 cages per site). This average is referred to as the *lice abundance* at a given site. These weekly reports are downloaded through Barentswatch<sup>1</sup>. Missing

<sup>1</sup><https://www.barentswatch.no/fiskehelse/>

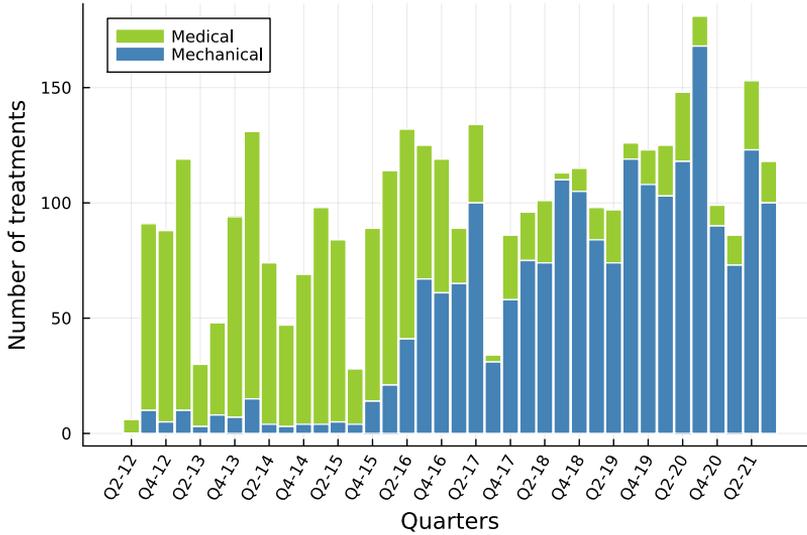


**Figure III.2:** Different stages of the lice development cycle used for the present model. Transportation happens in the two first stages (LR and R), and lice are counted in the latter three (ST, MB = PA + AM and AF) while attached to the fish.

lice counts are replaced by interpolation between the nearest foregoing and upcoming reports, reduced by 70% per week from the interpolation points (1597 interpolated points in total).

Farmers are obliged to commence lice treatment on the entire site when lice abundance is measured above 0.2 or 0.5 (depending on the time of year and location) (Lovdata, 2012). The lower limit was changed from 0.1 to 0.2 in 2017 (Lovdata, 2017). The weekly lice reports collected from Barentswatch also contain information on treatments, and these mainly distinguish between medical, mechanical, and cleaner fish treatments. Cleaner fish are other species of smaller fish that eat lice off Salmonids, which are deployed in the cage. Partial treatment of the site is allowed only by particularly good reason; hence, we do not distinguish between partial and complete treatments. Treatments are reported weekly but may last for more than one week. For medical treatments, farmers first report the start date and later report the end date of the treatment. We use the starting week as the occurrence of medical treatments. Medical treatments with missing start dates are placed at the week before the reported end of treatment. Treatments of the same kind (medical or mechanical) reported multiple times in consecutive weeks are joined together as a single treatment and placed on the first week of reporting. This prevents duplicates of the same treatment taking place over several weeks. If there are more than a certain number of treatments in consecutive weeks (3 for mechanical and 4 for medical), these are split as separate treatment periods, with commencement at the first week of each period. By visual inspection, we find that some sites have missing treatment reports. These sites are removed when fitting the models (4 sites in total), but still contribute to neighbour effects and are included in simulations. There are still a few minor active time periods where we suspect treatment reports are missing, but these are still included. Cleaner fish treatments are not necessarily reported after 2018 (Lovdata, 2018). For this reason, the effect of cleaner fish is not included when forecasting treatments but is included as a corrective term during model estimation. Figure III.3 illustrates the number of treatments in production area 3 within the relevant time range, and we see a clear shift from medical treatment to mechanical treatment. This is due to the development of resistance to commonly used medical substances, which caused a transition to mechanical treatments instead (Jensen et al., 2020). There are treatments in 8.4% of active weeks (3499 in total).

Salmonid farming in Norway requires two months of fallowing (closing the site) at the end of each production cycle as a preventive measure against lice, and there are no lice reports during this time (Lovdata, 2008). While a site is active, farmers are obliged to report lice every week, and we can know which sites are active based on the presence of



**Figure III.3:** Treatments in production area 3.

reports. However, some reports are missing in the data. We must distinguish between inactivity and missing reports since inactivity means lice abundance is zero while missing reports do not. If there are more than six weeks of consecutively missing reports, we assume the site is inactive, while otherwise, we assume reports are missing and interpolate the missing data. This is conservative by letting some (shorter) inactive periods potentially seem active instead, which prevents zeroing out the dynamics of lice abundance. We assume there are no missing treatments in weeks of missing reports.

Several sources of uncertainty are related to the lice count and treatment data. An extensive survey on this was conducted and summarized by Solberg et al. (2018) in 2017. First, lice counts are known to have considerable measurement error, mainly attributed to the small sample of fish collected to count lice. The small sample is motivated by negative health implications from handling the fish when counting lice. Second, there is uncertainty about whether the human counter can identify lice on the fish and correctly classify the stage of the louse. In particular, lice in the stationary stage (ST) are especially small in size and can be hard to observe (Thorvaldsen et al., 2019). Third, there are doubts as to whether the sample of fish collected for counting is representative of all fish in a cage since these are collected only from the sea surface. Fourth, the regulation allows counting lice multiple times in each cage and reporting only the last counting, which enables selective re-counting. Fifth, due to the high cost of treatments and the low treatment limit, farmers have clear incentives to under-report lice abundance and avoid unnecessary treatments, which can affect the quality of the data. The impact of such incentives on this dataset was explored by Jeong et al. (2023). Lastly, regulation allows treatments to be dismissed for the last three weeks before fallowing, which can explain some higher lice counts at the end of production cycles. Recently, technology to count lice by camera technology has been developed and is already in use at some sites. This means more reliable count data will become available over time.

## Temperature and transportation by coastal currents

Temperature estimates at three-meter depth are collected from the NorKyst800 model (Albretsen et al., 2011), which describes coastal conditions in the form of currents and temperature on a 800m by 800m grid. This model has been validated to give high-quality temperature estimates with a deviation of at most  $1^{\circ}\text{C}$  (Asplin et al., 2020). Occasional missing data within short time spans are filled by linear interpolation. We assume temperature forecasts can be obtained from other models, for example, from the European Centre for Medium-Range Weather Forecasts<sup>2</sup>, and consider it to be a given explanatory variable for this paper.

Transportation of lice between site pairs is mainly governed by stream patterns. We derive transportation patterns from a particle simulation model developed by the Norwegian Institute of Marine Research (see Myksvoll et al., 2018, for details), which uses stream patterns from the NorKyst800 model to simulate how particles drift. The particle simulation model is currently used in a national monitoring system to regulate the industry and has been validated against field experiment data to have high accuracy (Myksvoll et al., 2018). In detail, the simulation runs as follows: For every hour, three particles are released from every site and tracked as they follow stream patterns determined by NorKyst800. Whenever a particle is in proximity to another site, it is registered as an encounter with information about the source site, destination site, and age in hours. Particles whose age in degree-days surpasses 200 are removed. The simulation runs for nine months of stream data from February to October 2017. Sites that were not active during this simulation window have been removed (17 sites). From these simulations, we obtain a total transportation volume between pairs of sites, referred to as *connectivity* (in number of particles), as well as an empirical distribution of transportation times. Both connectivity and transportation times are asymmetric between site pairs, and Figure III.4 illustrates the relationship between connectivity and seaway distance used in previous literature (Aldrin et al., 2017, 2019; Jansen et al., 2012). Seaway distance is symmetric, and its discrepancy with connectivity can be large.

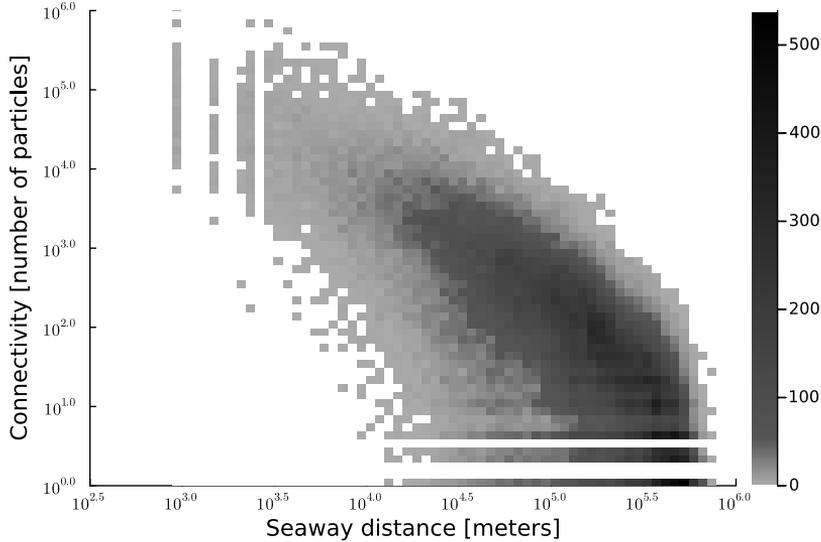
## 3 Methodology

To develop a model that can forecast treatments, we consider lice abundance and treatments as two processes that affect each other. Successful treatment should lead to lower lice abundance, while higher lice abundance should lead to increased probability of treatment. Furthermore, we assume the occurrence of treatment has a small delay so that it may only depend on past lice abundance; meanwhile, lice abundance may also depend on the occurrence of treatment within the same week. Figure III.5 illustrates this lead-lag relationship. This delay assumption on treatments allows constructing a joint forecasting model based on separate models for the conditional marginal distribution of lice abundance and treatment. These models are described in Section 3.1 and Section 3.2, respectively, while joint forecasting using both models is described in Section 3.3. The estimation procedure is described in Section 3.4.

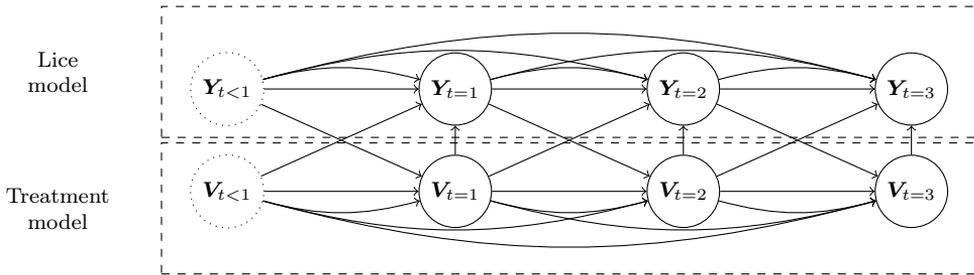
Each of the conditional models are designed in a two-level hierarchy having slightly different interpretations. First, we derive a *best-guess estimate* that incorporates all explanatory variables and domain knowledge about the respective phenomena. Second, we correct any remaining time-structure using Generalised Auto-Regressive Moving-Average (GARMA) models (Benjamin et al., 2003). For the lice model, the best-guess estimate

---

<sup>2</sup>[www.ecmwf.int/](http://www.ecmwf.int/)



**Figure III.4:** Comparison of seaway distance (in meters) to connectivity (number of particle encounters) among all site pairs.



**Figure III.5:** Dynamics between lice ( $Y$ ) and treatments ( $V$ ) at different time steps.

ensures consistent long-term forecasting ability based on explicit knowledge about the development-cycle of lice, while a moving-average component improves short-term predictions by correcting for more recent observations of actual lice abundance. For the treatment model, the best-guess estimate acts as a current assessment of the probability of treatment (mainly based on lice abundance) while an auto-regressive component corrects for relations to past treatments.

### 3.1 Lice abundance model

Let  $n_{it}Y_{it}^*$  denote the observed *lice count* within stages  $\star \in \{\text{ST}, \text{MB}, \text{AF}\}$  on  $n_{it}$  fish at site  $i \in \{1, \dots, N\}$  in week  $t$ . Correspondingly,  $Y_{it}^*$  denotes the observed *lice abundance*; namely, the average number of lice per fish. We represent the *occurrence of treatment* by a random variables  $V_{it}^\diamond$  where  $\diamond \in \{\text{mec}, \text{med}, \text{clf}\}$  denotes the different kinds of treatments: mechanical, medical or cleaner fish. Cleaner fish  $V_{it}^{\text{clf}}$  is a continuous rate, representing the number of cleaner fish deployed scaled by the capacity of the site. The binary variables

$V_{it}^\diamond$  for  $\diamond \in \{\text{mec}, \text{med}\}$  take value one if treatment  $\diamond$  is commenced at site  $i$  in week  $t$ , and zero otherwise. We let  $\mathcal{F}_t$  denote the information set generated by  $Y_{is}^\star$  and  $V_{is}^\diamond$  for all sites  $i = 1, \dots, N$ , weeks  $s = 1, \dots, t$ , stages  $\star \in \{\text{AF}, \text{MB}, \text{ST}\}$ , and treatment kinds  $\diamond \in \{\text{mec}, \text{med}, \text{clf}\}$ . For simplicity, we also let  $V_{it} = (V_{it}^{\text{mec}}, V_{it}^{\text{med}}, V_{it}^{\text{clf}})$  denote the concatenation of all treatment kinds.

The response variables in the lice model are the lice counts  $n_{it}Y_{it}^\star$  for all stages  $\star$  which we, conditional on  $\mathcal{F}_{t-1}$  and  $V_{it}$ , assume follow a negative binomial distribution

$$(n_{it}Y_{it}^\star \mid \mathcal{F}_{t-1}, V_{it}) \sim \text{NegBin}(n_{it}\mu_{it}^\star, n_{it}\nu^\star), \quad (\text{III.1})$$

parameterised by its expectation  $n_{it}\mu_{it}^\star$  and dispersion  $n_{it}\nu^\star$ . We assume a global dispersion parameter  $\nu^\star$  per stage, but scale by  $n_{it}$  to account for increased certainty in the abundance estimate when counting lice on more fish. The negative binomial distribution has been validated experimentally to be well suited for lice counts on farmed salmon (Jeong & Revie, 2020). For the response variable  $n_{it}Y_{it}^\star$ , we have  $n_{it} \in [10, 140]$  and  $Y_{it}^\star$  normally in the range  $[0, 1]$ .

The dynamics of lice abundance is modelled via the expected abundance  $\mu_{it}^\star$ . As an baseline explanatory term, we use a best-guess of lice abundance  $\bar{\mu}_{it}^\star$  derived from past recruits and their development into each stage  $\star$ . While the best-guess estimate has strong physical motivation that improves long-term forecasting ability, we also have regular observations of actual lice abundance in each stage that should correspond to  $\bar{\mu}_{it}^\star$ . Any discrepancy between the anticipated lice abundance  $\bar{\mu}_{it}^\star$  and observed abundance  $Y_{it}^\star$  is corrected for by a Generalised Moving-Average (GMA) model for  $\mu_{it}^\star$  to improve short-term predictions. Specifically, we let

$$\log(c + \mu_{it}^\star) = \log(c + \bar{\mu}_{it}^\star) + \sum_{l=1}^{20} A_{itl}\theta_l\zeta_{it}^\star, \quad (\text{III.2})$$

where

$$\zeta_{it}^\star = \log(c + Y_{it}^\star) - \log(c + \mu_{it}^\star), \quad (\text{III.3})$$

denotes the innovation of  $\mu_{it}^\star$  with respect to the observation  $Y_{it}^\star$ . The fitted parameter  $c > 0$  determines the zero-level of observations on the logarithmic scale to prevent values of  $-\infty$  whenever  $Y_{it}^\star, \bar{\mu}_{it}^\star, \mu_{it}^\star = 0$ . By design, we then have that  $\zeta_{it}^\star = 0$  whenever  $Y_{it}^\star = \mu_{it}^\star$ , meaning we default to the best-guess estimate whenever this discrepancy is low. The indicator  $A_{itl}$  takes value one if farm  $i$  is active from week  $t - l$  to week  $t$  and zero otherwise; this terminates the memory of the process whenever the site has been closed. Through model estimation, we find that the parameters  $\theta_l$  have a (close to) exponential decay in their values for increasing lags  $l$ ; hence, we let

$$\theta_l = \theta^{\text{scale}} \exp(\theta^{\text{rate}}(l - 1)), \quad (\text{III.4})$$

and estimate only the two parameters  $\theta^{\text{scale}}$  and  $\theta^{\text{rate}}$ . We use the same parameters  $\theta^{\text{scale}}$  and  $\theta^{\text{rate}}$  for this lag structure in all stages  $\star$ . The number of lags is determined by the physical model for  $\bar{\mu}^\star$ .

The primary aim of the lice model is to give a precise account of the dynamics of lice abundance while incorporating the effect of treatment, and the best-guess estimate of lice abundance  $\bar{\mu}^\star$  is motivated by the stage-structured development cycle of lice. For this purpose, we also consider the *unobserved* abundances  $Y^{\text{RN}}, Y^{\text{RS}}$  and  $Y^{\text{RU}}$  that decomposes all recruits (R) into their respective sources: neighbouring sites (RN), within-site

(RS), or unexplained sources (RU). The best-guess estimate then takes expression

$$\bar{\mu}_{it}^* = \kappa_{it0} \sum_{l=1}^{20} A_{itl} \kappa_{itl} r_{itl}^* \left( \hat{Y}_{i,t-l}^{\text{RS}} + \hat{Y}_{i,t-l}^{\text{RN}} + \hat{Y}_{i,t-l}^{\text{RU}} \right), \quad (\text{III.5})$$

which incorporates estimates of all relevant history of past recruits ( $\hat{Y}^{\text{RS}}$ ,  $\hat{Y}^{\text{RN}}$ ,  $\hat{Y}^{\text{RU}}$ ) while accounting for survivability ( $\kappa$ ) and temperature-dependent development times ( $r^*$ ). The effect of treatment is accounted for by survivability which is explained in Section 3.1, while the terms  $r^*$  that account for development time are described in Section 3.1. The estimates  $\hat{Y}^{\text{RN}}$ ,  $\hat{Y}^{\text{RS}}$  and  $\hat{Y}^{\text{RU}}$  for unobserved abundances are based on hydrodynamic simulations, biological relations, and past observations of adult female lice; and these are described in Section 3.1. We use a maximum lag of 20 weeks since most contributions beyond 20 weeks are small when accounting for both development time and mortality.

### Mortality

We let the factor  $\kappa_{its}$  represent the survivability of lice. Specifically, the log-survivability rate at site  $i$  between weeks  $t-s$  and  $t$  takes the expression

$$\log(\kappa_{its}) = s\rho + \sum_{l=0}^s \left( V_{i,t-l}^{\text{mec}} \sum_{k=0}^l \delta_k^{\text{mec}} + V_{i,t-l}^{\text{med}} \sum_{k=0}^l \delta_k^{\text{med}} + V_{i,t-l}^{\text{clf}} \sum_{k=0}^l \delta_k^{\text{clf}} \right), \quad (\text{III.6})$$

where the parameter  $\rho < 0$  represents a baseline weekly mortality. The parameters  $\delta_k^\circ < 0$  denote the additional log-effect of various treatments,  $k \in \{0, 1, \dots\}$  weeks after they were initiated. The total effect of a single treatment initiated  $l$  weeks ago then equals  $\exp\left(\sum_{k=0}^l \delta_k^\circ\right)$ . Hence,  $\kappa_{its}$  is the accumulated survival rate between weeks  $t-s$  and  $t$  which incorporates all natural and treatment-induced mortality. Observe also that  $\kappa_{it0}$  incorporates the effect of treatment in the current week but omits the baseline weekly mortality  $\rho$ .

### Development time

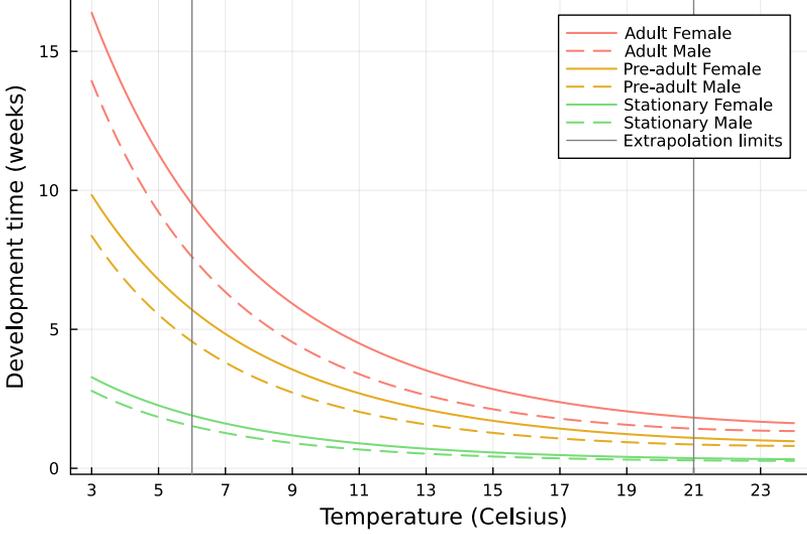
To account for development time, we derive auto-regressive coefficients to infer the abundance of counted lice (ST, MB and AF) from the abundance of recruits (RS, RN and RU) in the past. These auto-regressive coefficients are derived as an analytical expressions of the (temperature-dependent) development time, and the advantage of this analytical approach is that we may use existing knowledge about the dependence of development times on temperature. This avoids the need for re-estimating these effects. Only the potential variability in development time is estimated.

The development time going from a recruit into each consecutive development stage was examined by Hamre et al. (2019), who found that the development time (in weeks) between stages is independent of age but highly dependent on temperature as well as gender. Let  $T$  denote temperature in general, and  $T_{it}$  the temperature at site  $i$  in week  $t$ . The temperature-dependent development time for male and female lice, respectively, take expressions

$$D_M(T) = \exp(1.7216 - 0.2472T + 0.0050T^2), \quad (\text{III.7})$$

$$D_F(T) = \exp(1.8033 - 0.2172T + 0.0039T^2), \quad (\text{III.8})$$

which are estimated from the data in (Hamre et al., 2019). See details in the Supplemental Material. Available data covers temperatures in the range 6–21°C, and we extrapolate outside of these. The development time from recruit into stationary, pre-adult and adult lice are multiples 1, 3 and 5 of  $D_M(T)$  and  $D_F(T)$ , as illustrated in Figure III.6. We also assume lice die of old age after  $10D_M(T)$  and  $10D_F(T)$  weeks.



**Figure III.6:** Development time from recruit to the respective stages for both genders at different temperatures.

We use the expressions (III.7) and (III.8) to provide estimates for the average development time; however, to account for potential variability in development time, we assume a distribution for development time. We assume development time from recruit into stage  $\star$  is Weibull distributed, parameterised by its expectation  $\psi_\star$  and shape parameter  $\alpha > 0$ . Its cumulative distribution function (CDF) is then expressed as

$$F_\alpha(s | \psi_\star) = 1 - \exp\left(-\left(\frac{s\Gamma(1 + 1/\alpha)}{\psi_\star}\right)^\alpha\right), \quad (\text{III.9})$$

where  $\Gamma(\cdot)$  is the gamma function, and the parameter  $\alpha$  is subject to estimation. Using this CDF, we may now express the share of recruits that have developed into stage  $\star$  (with average development time  $\psi_\star > 0$ ) and *not* into the consecutive stage (with average development time  $\psi_{\star+1} \geq \psi_\star$ ) within  $s$  weeks as:

$$F_\alpha(s | \psi_\star) - F_\alpha(s | \psi_{\star+1}) \geq 0. \quad (\text{III.10})$$

This quantity is non-negative since these distributions have the same shape parameter  $\alpha$  while one expectation is larger than the other. The difference (III.10) specifies the contribution of recruits  $s$  weeks ago to each consecutive stage  $\star$  in the current week, and we may sum over all past weeks to find the total lice abundance. Due to asymmetries in the contribution of each gender to different stages, we get the following temperature-

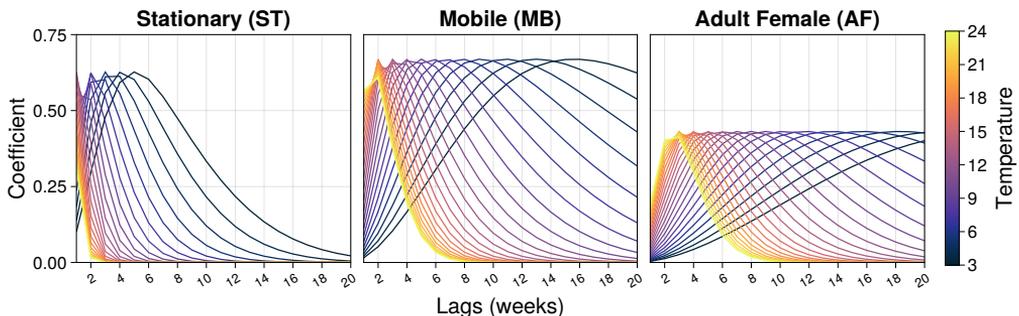
dependent coefficients to represent development:

$$r_{its}^{\text{AF}} = F_{\alpha}(s \mid 5D_F(T_{it})) - F_{\alpha}(s \mid 10D_F(T_{it})), \quad (\text{III.11a})$$

$$r_{its}^{\text{MB}} = F_{\alpha}(s \mid 3D_F(T_{it})) - F_{\alpha}(s \mid 5D_F(T_{it})) + F_{\alpha}(s \mid 3D_M(T_{it})) - F_{\alpha}(s \mid 10D_M(T_{it})), \quad (\text{III.11b})$$

$$r_{its}^{\text{ST}} = F_{\alpha}(s \mid 1D_F(T_{it})) - F_{\alpha}(s \mid 3D_F(T_{it})) + F_{\alpha}(s \mid 1D_M(T_{it})) - F_{\alpha}(s \mid 3D_M(T_{it})). \quad (\text{III.11c})$$

These have the interpretation that  $r_{its}^{\star}$  is the number of recruits from week  $t - s$  that contribute to lice abundance in stage  $\star$  and week  $t$ . The sources contributing to AF are only female lice in the adult stage, while the contribution to MB are pre-adult female lice as well as pre-adult and adult male lice. The contribution to ST is from male and female lice in the stationary stage. Lastly, we also consider mortality from old age for stages MB and AF. The inferred lead-lag relationship between recruits and each stage  $\star$  (while ignoring the effect of mortality) is illustrated for different temperatures in Figure III.7.



**Figure III.7:** Lead-lag relationship between recruits and the respective stages (ST, MB and AF) that infer temperature-dependent development times.

### Estimates of past recruits

Past recruits can be decomposed into different sources and estimated based on past abundance of adult female lice, transportation patterns and successful attachment. In the following, we assume transportation ends when a louse has successfully attached to a new host; then defined to be a recruit. The number of fish at location  $i$ , and thereby the total number of lice inferred by lice abundance, are generally not available since it is considered market-sensitive information. Instead, we use the capacity  $C_{it}$  of site  $i$  in week  $t$  (capacity will rarely change over time) as a proxy for fish count. The rate of egg production is temperature-dependent, whose expression is given by

$$H(T) = \exp(-0.869324 + 0.15615T - 0.007699T^2). \quad (\text{III.12})$$

This is estimated from data in (Samsing et al., 2016) and normalised to take values close to one. See the Supplemental Material.

The estimate of recruits from neighbouring sites  $Y_{it}^{\text{RN}}$  then takes expression

$$\hat{Y}_{it}^{\text{RN}} = \frac{l^{\text{RN}}}{C_{it}} \sum_{j \in J_i} \sum_{s=1}^5 w_{jits} [C_{j,t-s} H(T_{i,t-s}) Y_{j,t-s}^{\text{AF}}], \quad (\text{III.13})$$

where  $C_{j,t-s}H(T_{j,t-s})Y_{j,t-s}^{\text{AF}}$  is a proxy for the number of emitted larvae from site  $j$  in week  $t-s$ , and  $\iota^{\text{RN}} > 0$  is a scaling parameter. The weights  $w_{jits}$  represent the magnitude of transported lice that originate from site  $j$  in week  $t-s$  that successfully attaches to a new host at site  $i$  in week  $t$ . The set  $J_i$  contains all relevant neighbours of site  $i$ , limited to those that are shown to successfully transport particles to site  $i$  in the hydrodynamic simulation. The maximum number of transportation weeks is set to five (see point (iii) below). The weights  $w_{jits}$  are composed of four main factors: (i) total transportation rate as well as the distribution of transportation time, (ii) daily mortality rate before reaching a new host, (iii) time windows to find a new host, and (iv) attachment success rate once reaching a new host. While additional details are provided in the Supplemental Material, these are modelled as follows:

- (i) Let  $K_{ji}$  denote the connectivity from site  $j$  to  $i$  as number of particles successfully transported in the simulation (in millions). We also derive an empirical probability  $f_{jis}$  of transporting particles from site  $j$  to  $i$  within  $s-1$  to  $s$  weeks based on transportation times. Overall,  $K_{ji}f_{jis}$  is the number of particles transported from site  $j$  to  $i$  within  $s-1$  to  $s$  weeks.
- (ii) The weekly survival rate is set to  $\omega = \exp\left(\frac{7}{4}\log(0.5)\right) = 0.2973$  based on (Myksvoll et al., 2018; Stien et al., 2005), with the interpretation that the survival rate has a half-life of four days.
- (iii) There is a specific time window where a successful transmittance can take place. The lower limit of the window comes from the fact that larvae must develop before they attach to a new host. The upper limit comes from the fact that they run out of nutrition and die within a certain number of weeks. This time window is illustrated by the black lines in Figure III.8. Within the relevant temperature range, there is a maximum transportation time of 5 weeks.
- (iv) Infectivity is defined as the rate of successful infection once reaching a new host, which is both age and temperature dependent (Skern-Mauritzen et al., 2020). We have estimated a function  $I(A, T)$  for infectivity based on data from (Samsing et al., 2016; Skern-Mauritzen et al., 2020) (see the Supplemental Material), and use the average infectivity during a week  $\bar{I}_s(T)$  to infer infestation success. The function  $I(A, T)$  is illustrated in Figure III.8.

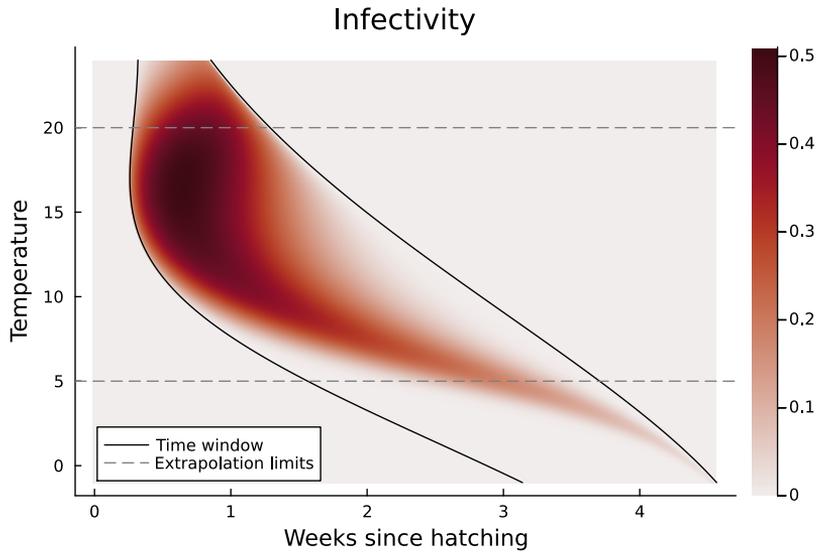
Based on (i)–(iv), the weights that account for successful transportation and attachment are given as:

$$w_{jits} = A_{it}A_{j,t-s}\bar{I}_s(T_{it})K_{ji}f_{jis}\omega^s, \quad s \in \{1, \dots, 5\}. \quad (\text{III.14})$$

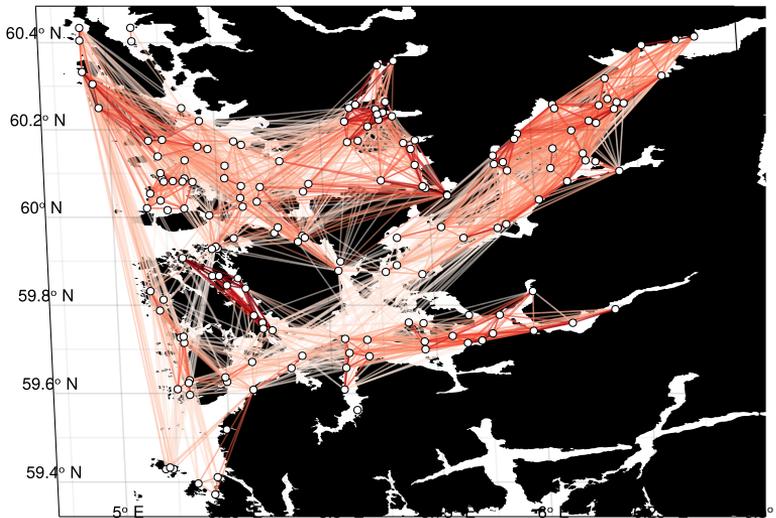
The indicator  $A_{it}$  is equal to one if site  $i$  is active in week  $t$  and zero otherwise; hence,  $A_{it}A_{j,t-s}$  takes value one if both the emitting site is active during emission and the receiving site is active during arrival. In Figure III.9, we illustrate the (logarithm of the) aggregated transferal and attachment rate between sites (i.e.,  $\log(\sum_{s=1}^5 w_{jits})$  for all  $i$  and  $j$ ) for  $T = 9^\circ\text{C}$  and assuming all sites are active.

The estimate of within-site recruits  $\hat{Y}_{it}^{\text{RS}}$  is derived from the same components as before but excluding travel time. Its estimate takes expression

$$\hat{Y}_{it}^{\text{RS}} = \iota^{\text{RS}}A_{it}\sum_{s=1}^5 A_{i,t-s}\bar{I}_s(T_{it})H(T_{i,t-s})Y_{i,t-s}^{\text{AF}}, \quad (\text{III.15})$$



**Figure III.8:** Infectivity  $I(A, T)$  as a function of weeks since hatching ( $A$ ) and temperature ( $T$ ). The infection window is shown in black. In the model, we extrapolate beyond the data range of temperatures from  $5^{\circ}\text{C}$  to  $20^{\circ}\text{C}$  (dashed lines) to temperatures from  $-1^{\circ}\text{C}$  to  $24^{\circ}\text{C}$  (indicated by the axis limits).



**Figure III.9:** Logarithm of the aggregated transferal and attachment rate between sites at temperature  $T = 9^{\circ}\text{C}$  assuming all sites are active, in production area 3. Only rates above the 80th quantile are shown.

which accounts for the past abundance of adult female lice, hatch rate, infectivity, and the active status of the site. The fitted parameter  $\iota^{\text{RS}} > 0$  is a scaling.

Lastly, the estimate of recruits from unexplained sources  $\hat{Y}^{\text{RU}}$  mainly acts as an intercept term, expressed as

$$\hat{Y}_{it}^{\text{RU}} = \iota^{\text{RU}} \exp(\beta_T(T_{it} - 10)). \quad (\text{III.16})$$

This is limited to a temperature correction to reflect seasonality and scaling by the parameter  $\iota^{\text{RU}} > 0$ . By letting the intercept term contribute to recruits instead of  $\bar{\mu}^*$  directly, we get a gradual build-up of lice abundance directly after a site has been re-opened and the intercept will also be affected by treatments.

### 3.2 Treatment model

When forecasting treatments, we are mainly interested in the the potential need for future treatments rather than the specific kind of treatment; thus, we model only whether a treatment occurs or not. Cleaner fish are mainly used as a preventative measure and are ignored for this purpose. Let  $W_{it} = V_{it}^{\text{mec}} \wedge V_{it}^{\text{mec}}$  denote the binary random variable for whether a mechanical or medical treatment occurs at site  $i$  in week  $t$ . We then assume the occurrence of treatment follows the conditional distribution

$$(W_{it} \mid \mathcal{F}_{t-1}) \sim \text{Bernoulli}(p_{it}), \quad (\text{III.17})$$

where  $p_{it}$  is the probability of treatment and  $\mathcal{F}_{t-1}$  is the information set up to time  $t - 1$  as defined in Section 3.1. This probability is modelled as a Generalised Auto-Regressive (GAR) model with explanatory variables using a logistic response function.

We express the probability of treatment through  $\eta_{it} = \text{logit}(p_{it})$  and let  $\bar{\eta}_{it}$  denote the best-guess estimate of the log-odds for treatment based on explanatory variables. This best-guess estimate is given as

$$\bar{\eta}_{it} = \beta_0^p + \beta_L^p L_{it} + \sum_{\star \in \{\text{MB}, \text{AF}\}} (\gamma_i^* + \gamma^{I, \star} W_{i, t-1}) (Y_{i, t-1}^{\star})^{\pi^*}, \quad (\text{III.18})$$

where  $\beta_0^p$  is an intercept, and the indicator variable  $L_{it}$  takes value one if the treatment limit is low (i.e., the limit is below 0.5) and zero otherwise. The last term represents the effect of observed lice abundance in the previous week, as well as its interaction with last week's treatment. We find that no further lags on lice abundance are required due to small and insignificant coefficient estimates. Using stationary lice ( $Y_{it}^{\text{ST}}$ ) to infer treatment has a negligible effect; hence, it is omitted. The parameters  $\beta_0^p, \beta_L^p, \gamma_i^*, \gamma^{I, \star}$  and  $\pi^*$  (non-linearity) for  $\star \in \{\text{AF}, \text{MB}\}$  are subject to estimation.

We also correct this best-guess estimate ( $\bar{\eta}$ ) using auto-regressive relationships on its past prediction errors to improve its forecasting ability. The corrected log-odds of treatment ( $\eta$ ) takes expression

$$\eta_{it} = \bar{\eta}_{it} + \sum_{l=1}^{20} A_{itl} \phi_l \xi_{i, t-1}, \quad (\text{III.19})$$

where

$$\xi_{it} = \tanh^{-1}(W_{it} - \text{logit}^{-1}(\bar{\eta}_{it})), \quad (\text{III.20})$$

denotes the innovation of  $\bar{\eta}_{it}$  with respect to the observation  $W_{it}$ . The indicator  $A_{itl}$  terminates the memory of the process whenever the site has been closed between weeks  $t - l$  and  $t$ . Since binary random variables give infinite values in the response space

( $\text{logit}(V_{it}) = \pm\infty$ ), we make comparisons in the input-space  $[0, 1]$  using the transformation  $\text{logit}^{-1}(\bar{\eta}_{it})$  first, and then transform differences back using  $\tanh^{-1}(\cdot)$  which allows for negative values. Observe also that whenever  $\text{logit}^{-1}(\bar{\eta}_{it}) \approx W_{it}$ , we have that  $\xi_{it} \approx 0$  and we default to the best-guess estimate; meanwhile, larger discrepancies between  $\text{logit}^{-1}(\bar{\eta}_{it})$  and  $W_{it}$  give progressively larger values of  $\xi_{it}$  causing larger corrections. Through model estimation, we find an exponential decay in the auto-regressive coefficients; hence, we let

$$\phi_l = \phi^{\text{scale}} \exp(\phi^{\text{rate}}(l-1)), \quad (\text{III.21})$$

and estimate only the two parameters  $\phi^{\text{scale}}$  and  $\phi^{\text{rate}}$ .

### 3.3 Joint forecasting

We now explain the steps and underlying assumptions used to derive joint forecasts using both models for lice abundance and treatment. Let  $f(Y_{i,t+1}^* | \mathcal{F}_t, V_{i,t+1})$  denote the conditional distribution for lice abundance in stage  $\star$  at site  $i$  and week  $t+1$  (as defined in Section 3.1) and let  $f(W_{i,t+1} | \mathcal{F}_t)$  denote the conditional distribution of the occurrence of treatment (as defined in Section 3.2). For simplicity, we let  $Y_{it} = (Y_{it}^{\text{ST}}, Y_{it}^{\text{MB}}, Y_{it}^{\text{AF}})$  denote the concatenation of all stages of lice.

Assuming the realisation of lice abundance in each stage are independent of each other, conditional on  $\mathcal{F}_t$  and  $V_{i,t+1}$ , we determine their joint distribution as

$$f(Y_{i,t+1} | \mathcal{F}_t, V_{i,t+1}) = \prod_{\star \in \{\text{ST, MB, AF}\}} f(Y_{i,t+1}^* | \mathcal{F}_t, V_{i,t+1}). \quad (\text{III.22})$$

The need for treatment  $W_{it}$  is resolved by assuming farmers only use mechanical treatments since, lately, the majority of treatments are mechanical (see Figure III.3). Thus, the distribution for treatment is determined as

$$f(V_{i,t+1} | \mathcal{F}_t) = f(V_{i,t+1} | W_{i,t+1})f(W_{i,t+1} | \mathcal{F}_t), \quad (\text{III.23})$$

where the (deterministic) distribution  $f(V_{i,t+1} | W_{i,t+1})$  translates  $W_{i,t+1}$  to  $V_{i,t+1}$ . Lice abundance depends on treatment within the same week, and their joint distribution is then determined as

$$f(Y_{i,t+1}, V_{i,t+1} | \mathcal{F}_t) = f(Y_{i,t+1} | \mathcal{F}_t, V_{i,t+1})f(V_{i,t+1} | \mathcal{F}_t). \quad (\text{III.24})$$

Consequently, the realisation of treatment  $V_{i,t+1}$  must always be determined *before* lice abundance  $Y_{i,t+1}$ . It follows that the one-week-ahead forecast distribution at each site is a two component mixture consisting of two cases: with treatment or without treatment, where the probability of treatment determines their respective weights.

Up to now, we have only addressed the forecast distribution at an individual site. For simplicity, let  $\mathbf{Y}_t = \{Y_{it}\}_{i=1, \dots, N}$  and  $\mathbf{V}_t = \{V_{it}\}_{i=1, \dots, N}$  denote the lice abundances and treatments across all sites. We assume that *within a given week*, the realisation of treatment and lice abundance are independent across sites. Their joint distribution is then expressed as

$$f(\mathbf{Y}_{t+1}, \mathbf{V}_{t+1} | \mathcal{F}_t) = \prod_{i=1}^N f(Y_{i,t+1}, V_{i,t+1} | \mathcal{F}_t). \quad (\text{III.25})$$

Recall that lice abundance and treatment at each site depend on the history of (almost) all other sites since these are connected by currents that transport lice between them. To

forecast one additional week at a single site, we must condition on the realisations of  $\mathbf{Y}_{t+1}$  and  $\mathbf{V}_{t+1}$  in the previous week, whose distribution is determined as

$$f(Y_{i,t+2}, V_{i,t+2} | \mathcal{F}_t) = f(Y_{i,t+2}, V_{i,t+2} | \mathcal{F}_t, \mathbf{Y}_{t+1}, \mathbf{V}_{t+1})f(\mathbf{Y}_{t+1}, \mathbf{V}_{t+1} | \mathcal{F}_t). \quad (\text{III.26})$$

While the one-week-ahead prediction has a simple analytical expression, we must resort to joint simulation across all sites to forecast beyond a single week. By repeating the above steps, we may also forecast multiple weeks ahead.

To summarise, the steps required to simulate the joint development of treatment and lice abundance across all sites are:

1. For each site  $i$ :
  - (a) Predict treatment probability  $p_{i,t+1}$  conditional on information  $\mathcal{F}_t$
  - (b) Sample a treatment realisation  $W_{i,t+1}$  using probability  $p_{i,t+1}$  and set  $V_{i,t+1}^{\text{mec}} = W_{i,t+1}$
  - (c) Predict the expected lice abundances  $\mu_{i,t+1}^*$  for each stage  $\star$ , conditional on  $\mathcal{F}_t$  and the realisation of  $V_{i,t+1}$
  - (d) Sample the outcome of each lice count  $n_{i,t+1} Y_{i,t+1}^*$  given its expectation  $n_{i,t+1} \mu_{i,t+1}^*$
2. Update  $\mathcal{F}_{t+1}$  to include the sampled realisations of  $\mathbf{Y}_{t+1}$  and  $\mathbf{V}_{t+1}$  across all sites
3. Repeat for  $t + 2, t + 3, \dots, t + k$

By repeating the simulation scheme above, we can derive an empirical  $k$ -week-ahead forecast distribution  $\hat{f}(\mathbf{Y}_{t+k}, \mathbf{V}_{t+k} | \mathcal{F}_t)$  conditional on all current information  $\mathcal{F}_t$  up to week  $t$ .

The reason for finding the joint forecast distribution is to infer different sites' exposure to future treatments. More importantly, we are also concerned with *heterogeneity* in their risk exposure to guide where to harvest within a portfolio of sites. As a proxy for risk exposure, we use the above simulation scheme to derive the empirical distribution of the aggregated future treatment count at each site within a  $k$ -week horizon:

$$R_{it}^k = \sum_{s=1}^k W_{i,t+s} | \mathcal{F}_t. \quad (\text{III.27})$$

Meanwhile, risk management models apply the distribution  $\hat{f}(\mathbf{Y}_{t+k}, \mathbf{V}_{t+k} | \mathcal{F}_t)$  directly. Information on future treatments can be incorporated into harvest plans by minimising overall exposure among multiple sites. Since harvest plans are made collectively, these must trade off which fish groups are to be exposed the longest, and such considerations may also be combined with information about the health state of the fish. This is investigated further in Section 4.3.

### 3.4 Estimation

The lice and treatment models were fitted separately by maximizing likelihood estimation (MLE). The log-likelihoods were implemented with the assistance of Template Model Builder (TMB), a free and open-source R package specifically designed for estimating complex non-linear models (Kristensen et al., 2016). The various parameter constraints are handled by maximizing a re-parameterised version of the log-likelihood, which varies over a set of unconstrained parameters. For instance, the constraint  $\rho > 0$  for baseline weekly

mortality is parameterized as  $\rho = \exp(u)$ , where  $u$  is an unconstrained parameter. TMB provides the gradient and Hessian computed using automatic differentiation (Fournier et al., 2012) at machine precision, and this is used in the R-routine *nminb* for optimization. The C++ template functions have been made available online<sup>3</sup>. Standard deviations are computed using the Delta-method from the gradient and Hessian evaluated at the optimal parameter estimates.

Estimation is performed on data in the period 2018–2021 due to structural instabilities that stabilise by 2018. This is mainly attributed to the decline in the efficacy of medical treatments and changes in the counting dispersion of adult female lice. The transition was found by performing rolling estimation on fixed-width, two-year time windows. The Supplemental Material describes these observations in more detail.

## 4 Results and validation

We emphasize the forecasting ability of the joint model but also interpret the estimated models (Section 4.1) and each of their forecasting abilities (Section 4.2). The joint model is analysed in Section 4.3.

### 4.1 Model interpretation and in-sample analysis

Estimates of all parameters for both models are given in Table III.1 as well as their standard deviation derived by the Delta-method. For the lice model, we find that:

- The dispersion parameters  $\nu^\star$  are incrementally increasing for  $\star \in \{\text{ST}, \text{MB}, \text{AF}\}$ , which means counting precision gets incrementally better for later stages. This corresponds well with reported experiences that adult female lice are easier to identify than those in the stationary stage (Thorvaldsen et al., 2019).
- The baseline weekly mortality rate is approximately  $1 - \exp(\rho) = 5.47\%$  while the total effect of treatments are  $1 - \exp(\sum_k \delta_k^{\text{mec}}) = 38.8\%$  for mechanical and  $1 - \exp(\sum_k \delta_k^{\text{med}}) = 31.7\%$  for medical.
- We can infer that the memory from past innovations is sufficiently low at the last included lag due to the coefficient value of  $\theta_{20} = \theta^{\text{scale}} \exp(\theta^{\text{rate}} \cdot 19) = 0.006$ .

For the treatment model, we find that:

- Having a lower treatment limit ( $L_{it}$ ) increases the odds of treatment by a multiplicative factor  $\exp(\beta_L) = 2.16$ , which is to be expected.
- The effect of lice on the probability of treatment is significant for both  $\gamma^{\text{AF}}$  and  $\gamma^{\text{MB}}$ . Meanwhile, the effect of stationary lice ( $Y_{it}^{\text{ST}}$ ) on the probability of treatment was insignificant and removed during model selection.
- The memory from past innovations is sufficiently low since the last lagged autoregressive coefficient has a value of  $\phi_{20} = \phi^{\text{scale}} \exp(\phi^{\text{rate}} \cdot 19) = 0.037$ .

---

<sup>3</sup>[Note: link will be provided after review]

**Table III.1:** Parameter estimates for both models. Standard deviations (SD) are derived by the Delta-method.**Lice model**

Parameter	Estimate	SD
$\nu^{\text{AF}}$	0.0880	0.0019
$\nu^{\text{MB}}$	0.0535	0.0008
$\nu^{\text{ST}}$	0.0253	0.0004
$\iota^{\text{RU}}$	0.0446	0.0028
$\iota^{\text{RS}}$	3.5767	0.1746
$\iota^{\text{RN}}$	1.0337	0.0945
$\beta^T$	0.0584	0.0108
$\alpha$	1.9760	0.0907
$\rho$	-0.0563	0.0064
$\delta_0^{\text{med}}$	-0.4911	0.0137
$\delta_0^{\text{med}}$	-0.0000	0.0001
$\delta_1^{\text{med}}$	-0.0935	0.0355
$\delta_2^{\text{med}}$	-0.1187	0.0431
$\delta_3^{\text{med}}$	-0.0575	0.0486
$\delta_4^{\text{med}}$	-0.1113	0.0552
$\sum_k \delta_k^{\text{med}}$	-0.3810	0.0613
$c$	0.0065	0.0002
$\theta^{\text{scale}}$	0.5709	0.0067
$\theta^{\text{rate}}$	-0.2387	0.0060

**Treatment model**

Parameter	Estimate	SD
$\beta_0$	-6.5164	0.4789
$\beta_L$	0.7705	0.1190
$\gamma^{\text{AF}}$	3.9822	0.2264
$\gamma^{\text{MB}}$	3.0541	0.5056
$\gamma^{I,\text{AF}}$	-1.4766	0.8216
$\gamma^{I,\text{MB}}$	-2.2604	0.4635
$\pi^{\text{AF}}$	0.5774	0.0397
$\pi^{\text{MB}}$	0.1734	0.0292
$\phi^{\text{scale}}$	0.4733	0.0459
$\phi^{\text{rate}}$	-0.1336	0.0180

**Sources of lice**

Three sources contribute to lice recruits: neighbours (RN), within-site (RS), and unexplained sources (RU). When also accounting for development time and mortality rates, we can decompose the sources of lice by analysing each of their contributions to  $\bar{\mu}_{it}^* = \lambda_{it}^{*,\text{RS}} + \lambda_{it}^{*,\text{RN}} + \lambda_{it}^{*,\text{RU}}$  where

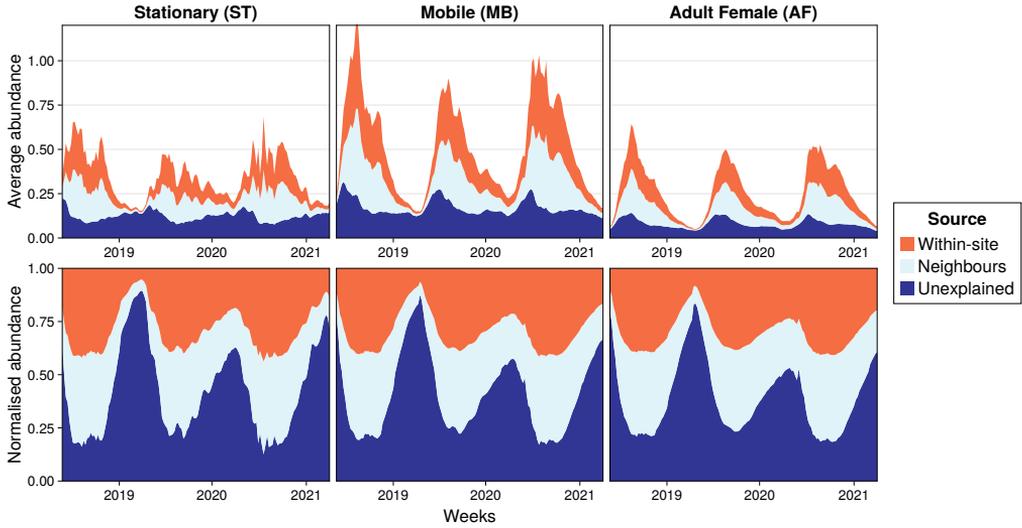
$$\lambda_{it}^{*,\text{RS}} = \sum_{l=1}^{20} r_{itl}^* \hat{Y}_{i,t-l}^{\text{RS}}, \quad \lambda_{it}^{*,\text{RN}} = \sum_{l=1}^{20} r_{itl}^* \hat{Y}_{i,t-l}^{\text{RN}}, \quad \lambda_{it}^{*,\text{RU}} = \sum_{l=1}^{20} r_{itl}^* \hat{Y}_{i,t-l}^{\text{RU}}. \quad (\text{III.28})$$

We do an in-sample analysis of this decomposition by averaging each source over all active sites in a given week to get a time-development of relative contributions:  $\frac{1}{N} \sum_{i=1}^N \lambda_{it}$ . These are illustrated in Figure III.10 in terms of average and normalised weekly quantities.

First, we see from Figure III.10 a clear seasonal pattern due to temperature variations. Second, in late winter periods we observe that the explainable sources of lice reaches almost zero. This may be explained by increased time until infection and increased development time (due to low temperatures) such that the aggregated mortality rate (over longer time periods) is higher. Infectivity is also lower at low temperatures (see Figure III.8). Third, we observe that there seems to be an almost equal contribution to lice abundance from neighbours and from hatching at the same site. Lastly, we observe that the overall level of mobile lice is higher compared to the other stages; hence, the model reflects that this stage includes adult male lice in addition to pre-adults from both genders. This is also consistent with higher levels of mobile lice in the data.

**Treatment patterns**

While the auto-correlation function is a useful diagnostics tool for continuous ARMA processes, the *auto-persistence function* (APF) is a valuable counterpart for binary time-

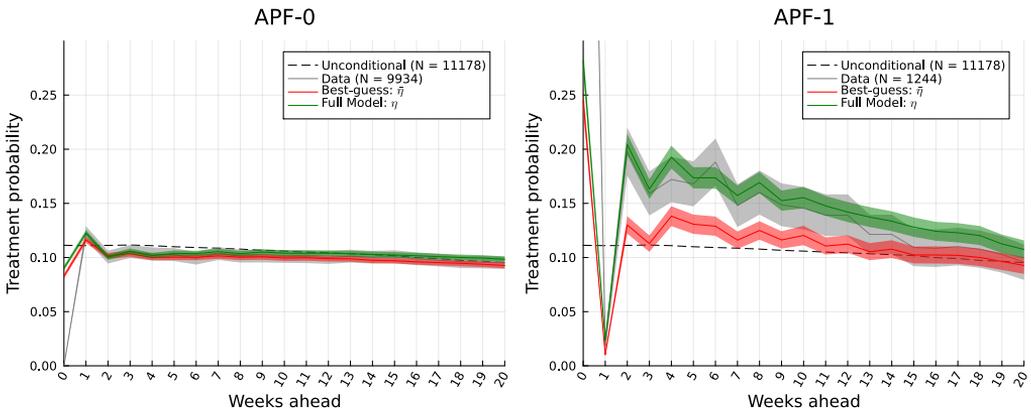


**Figure III.10:** Sources that contribute to expected lice abundance at different stages, averaged over sites.

series data (Startz, 2008). The APF is stated as the conditional probability of future treatments:

$$\text{APF}_k^0 = P(V_{i,t+k} | V_{it} = 0), \quad \text{APF}_k^1 = P(V_{i,t+k} | V_{it} = 1). \quad (\text{III.29})$$

Figure III.11 illustrates the empirical estimates of the APFs along with the corresponding in-sample estimates from the treatment model; both in terms of explanatory variables ( $\bar{\eta}$ ) and using the correction by auto-regression ( $\eta$ ).



**Figure III.11:** Auto-persistence function (APF) illustrating the empirical probability of future treatments conditional on treatment in the current week (APF-0 = No treatment, APF-1 = Treatment) in 20-week time windows of active weeks. Corresponding probabilities using in-sample model estimates with explanatory variables ( $\bar{\eta}$ ) and with an added auto-regressive correction ( $\eta$ ) are also shown. Ribbons give 95% confidence intervals.

From APF-1 in Figure III.11, we see that treatments in the week following a week

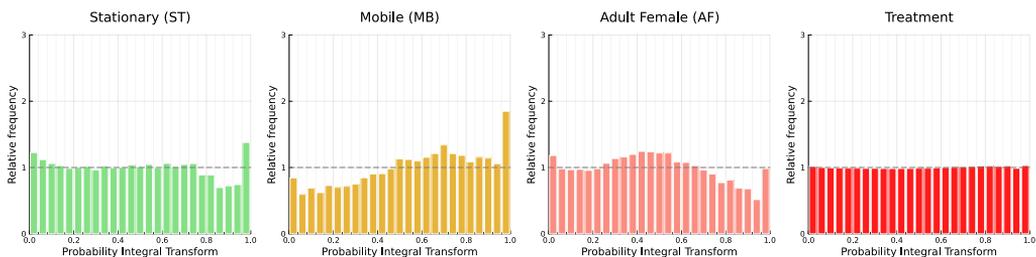
with treatments are very unlikely; however, this is also enforced in the data processing (see Section 2). APF-1 also shows a clear tendency that there is an increased probability of treatments from the second consecutive week and on-wards, given there is treatment in the current week. There are no notable characteristics in APF-0, neither in the data nor in the model predictions.

We may interpret the model-inferred value for  $\text{logit}(\bar{\eta})$  (which uses only explanatory variables) as the conditional probability of treatment, corrected for the effect of observed lice abundance. We observe from APF-1 in Figure III.11 that this gives predictions that are slightly better than the baseline unconditional probability of treatment, but underestimates the conditional probability of treatment as observed in the data. Meanwhile, the full model ( $\text{logit}(\eta)$ ) fits more closely to the empirical data and successfully corrects for the discrepancy between  $\text{logit}(\bar{\eta})$  and the data. This suggests lice abundance cannot predict treatments alone and that auto-regressive relations on previous treatments are able to explain additional effects.

On average, the treatment model assigns a 28.2% probability of treatment in weeks of treatment, while assigning a 9.0% probability of treatment in non-treatment weeks (from model estimates at week zero in Figure III.11). A naïve model alternative would be an unconditional Bernoulli distribution specified by the average probability of treatment, which is 11%. Clearly, our model improves on this. Note that these numbers are derived within the sub-selection of weeks that start off all 20 week time windows where a site is active.

## 4.2 One-week-ahead forecasting ability

Risk management models use distributional forecasts as input directly; hence, we emphasize the forecast distributional fit of the current models. For this purpose, we use the probability integral transform (PIT) as suggested by Dawid (1984). The idea is to assess the percentile of the data within their forecast distribution: A correct diagnostic should resemble a uniform distribution of percentiles with the interpretation that the data resemble samples from their respective forecast distribution. For discrete distributions, the PIT must be corrected to give a uniform distribution of percentiles (see Czado et al., 2009). By estimation up to time  $t$ , we find the forecast distribution for time  $t + 1$  at each site and evaluate the observed data within these (out-of-sample). The one-week-ahead forecast distributions are given analytically by (III.1) and (III.17), where we use the actual one-week-ahead treatment outcome to infer the distribution for lice abundance. This is repeated by rolling estimation, and evaluations are combined across all sites (totally 67 weeks and 5576 active site-weeks). The resulting PITs are illustrated in Figure III.12.



**Figure III.12:** Corrected Probability Integral Transforms for forecast distributions for lice stages, and treatment probabilities at all sites.

The forecast distribution of treatments and stationary (ST) lice give PITs that are very close to uniformity. Still, the PITs for mobile (MB) and adult female (AF) lice indicate that the corresponding forecast distributions are slightly biased (by being non-centred) and over-dispersed (inverted U-shape), while some data points are more extreme than expected (inflated probabilities at the ends).

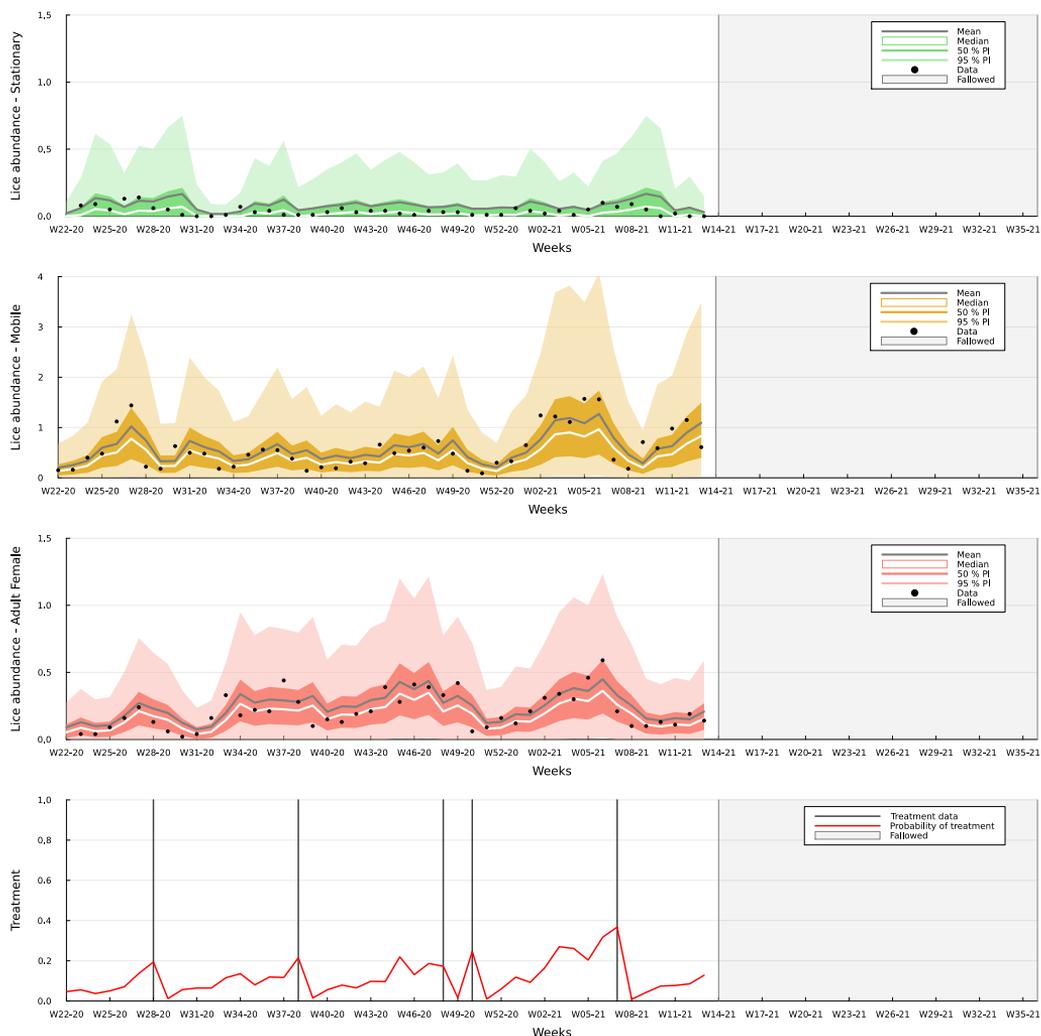
The irregularities in the forecast distributions of mobile (MB) and adult female (AF) lice have some possible explanations. First, the forecast distribution of adult female (AF) lice has a tendency to predict too large levels of lice. It is known that there are clear incentives for farmers to under-count adult female lice since this leads to costly treatments (Jeong et al., 2023), which might explain this bias. Furthermore, we also see a corresponding opposite bias in the predictions of mobile lice (MB); predicted lice levels are too low. These opposite biases may be related since there is no individual scaling between those quantities in the model, meaning the (incentive) bias in adult females may be the cause of both: predictions for MB are scaled down to account for the bias in AF. Various attempts to correct this bias typically lead to poor predictions of treatments in the joint model; hence, we do not correct this explicitly. Second, some data points are more extreme than expected, which is particularly pronounced for mobile lice (MB). This suggests the model does not capture some very rapid increases in lice abundance. We note there are occurrences of extreme lice abundance (above 5–10) in the data that are outside the range of all other counts, which may also explain some of these outliers. The GMA correction also adjusts the overall level of forecasts in consecutive time steps after an outlier, and only an initial high lice abundance should cause large prediction errors.

We illustrate one-week-ahead out-of-sample predictions of the respective models in Figure III.13 for an example site, along with actual data and prediction intervals of 50% and 95% confidence. Additional sites are illustrated in the Supplemental Material. First, we observe in Figure III.13 that the forecast distributions for lice abundance seem to follow the data closely and replicate the dynamics well. For the treatment model, there is a close correspondence between higher predicted treatment probability and weeks of actual treatments; hence, it gives a meaningful distinction between weeks of lower and higher probability of treatment. In this particular case, the predicted probabilities in weeks of actual treatments are in the range of 20–40%, meaning there is still some ambiguity to the specific week a treatment will take place. Lastly, we observe that the forecast distributions for lice abundance show tendencies of over-dispersion since more data points are within the 50% prediction intervals than expected. A single site is not sufficient to draw the conclusion of over-dispersion, but this is further supported by the PITs in Figure III.12.

### 4.3 Long-term joint forecasting

The primary aim of the joint forecasting scheme is to predict the total number of treatments within a  $k$ -week-ahead horizon, denoted  $R_{it}^k$  in (III.27), since this is a major driver of costs and biological risk. The expected number of future treatments is an interesting measure of risk, but variability is also important since that reflects exposure to sudden large changes that may not be accounted for in existing production plans.

First, we validate the forecasting ability of  $R_{it}^k$ . Figure III.14 shows out-of-sample prediction intervals of aggregated future treatments within a 20-week horizon (starting April 2021) along with actual treatment counts during the same period. We see most data points are between these 95% prediction intervals (specifically, 98.8% are within). We also notice that these prediction intervals are relatively wide, meaning there is large variability in the need for future treatments. Still, the joint forecasting model reflects this variability well.

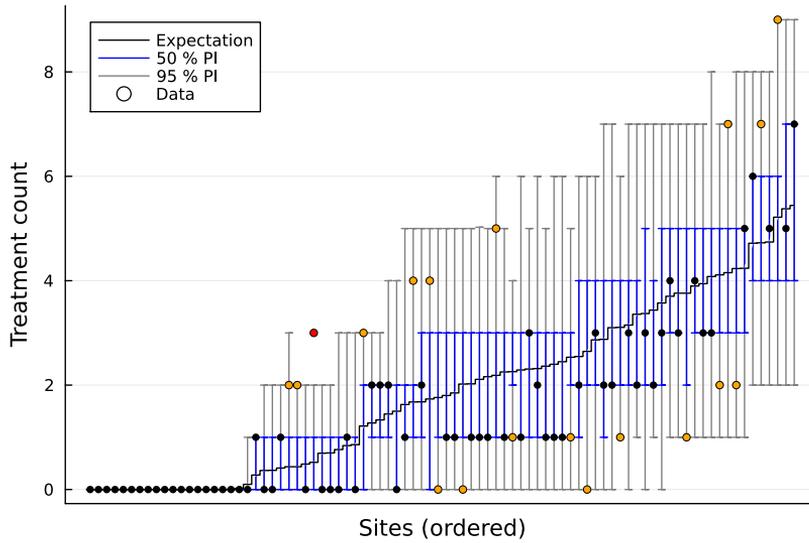


**Figure III.13:** Out-of-sample one-week-ahead forecast distributions as prediction intervals and probability of treatment for an example site.

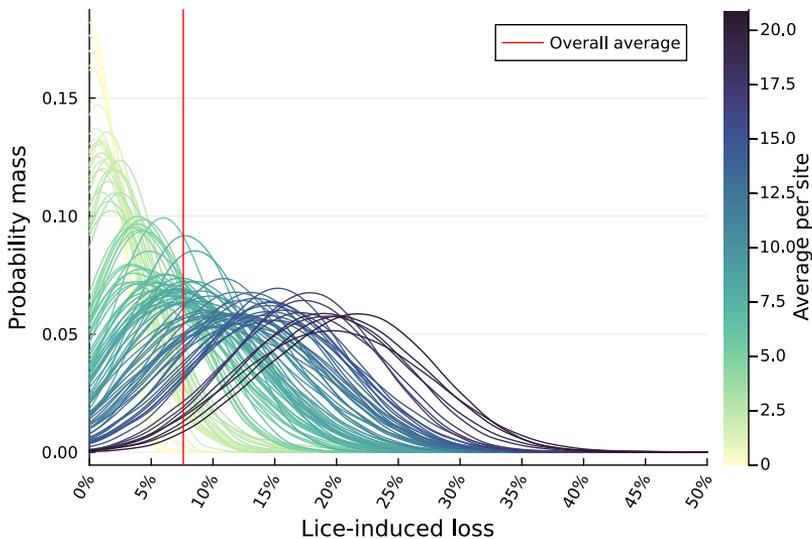
### Heterogeneity in risk exposure

For harvest planning and risk management, heterogeneity between sites is particularly interesting to guide the decision of *where* to harvest first. Farmers want to minimise overall risk exposure until a fish group is harvested subject to operational constraints that require different groups to be harvested in sequence. We do a simplified quantification of lice-induced losses to illustrate why heterogeneity in risk exposure matters. We account both for the direct cost of treatment and the indirect cost from loss in growth and increased fish mortality which is compared to having no treatments at all. Costs are then scaled by the sales value of fish to get a relative measure of lice-induced loss. Detailed assumptions are listed in the Supplemental Material. Figure III.15 illustrates the distribution of site-specific losses within a 20-week horizon as kernel densities.

First, we observe from Figure III.15 that the majority of sites have an expected loss



**Figure III.14:** Forecasted 50% and 95% prediction interval for treatment count within a 20-week horizon. Sites on the first axis are sorted by average forecasted treatment counts and non-active sites are removed. The centers of the prediction intervals are averages. Observations that are outside the 50% prediction interval are colored orange, while those that are also outside the 95% prediction interval are colored red.



**Figure III.15:** Site-specific risk exposure related to direct and indirect costs of lice treatments. Based on a 20-week forecast of treatment frequency on active sites. Illustrated as kernel densities for each site.

between 0% and 21%, with an overall average exposure of 7.60% loss. Already, this constitutes very considerable amounts while the tail end of the worst loss distributions also reaches up to 40%. This quantification of risk can be very valuable since such estimates of

exposure to still unrealised losses can be anticipated in advance and acted on. The aim of the forecasting tool developed in this paper is to plan for such risk exposure proactively instead of only reacting to on-going developments. Second, we see that different sites have widely different levels of exposure that constitute very considerable amounts. For example, the probability of more-than 10% losses is significantly different among sites. Farmers may act on this information by prioritising to harvest the more exposed fish groups earlier than others.

## 5 Discussion and concluding remarks

Constructing a model for the given data has shown particularly challenging due to a relatively low signal-to-noise ratio, and an important consideration to ensure long-term forecasting ability has been to avoid over-fitting. The main strategy to prevent over-fitting has been to incorporate as much existing knowledge about lice dynamics as possible while also reducing the number of parameters. Model selection using in-sample criteria like likelihood, AIC and BIC has mostly been unhelpful, while building the model based on physical intuition has shown to be more effective. Still, the structure of the final model acknowledges that not every aspect of lice dynamics can be explained by known causal mechanisms since it also requires an observation-based correction to more recent data (through GARMA models).

Incorporation of spatial effects is essential to determine heterogeneity in risk-exposure among sites since we already know this to be of major importance. In fact, the only source of heterogeneity in the joint model is stream patterns which, in turn, drive treatments at each site. Explicit use of stream patterns to derive heterogeneity between sites facilitates better trust that the model can perform well on future data and, due to its strong physical motivation, there is also reason to believe it performs well in new situations if sites are moved, removed or created. We emphasize trustworthiness since the model is to be applied to risk management, and validation of the joint model shows a good fit to the historical data of treatments that also replicates past heterogeneity between sites.

We model treatments as a stochastic process because the ambiguity is too high for when treatment limits are truly surpassed, likely due to the high counting dispersion. Simply determining treatments based on whether simulated lice counts are above the limit does not work very well in the forecasting model. In Figure III.11, we show that even a regression model on several explanatory variables that includes lice abundance is insufficient to explain treatment patterns in the data. Only once auto-regressions on past treatments are added do we replicate what is observed in the data. An interpretation of why auto-regressive relations matter is that farmers attain more information than is contained in the lice counts. For example, farmers may observe the fish visually without taking them out for counting, or they may have experience with how fast lice abundance develops at their site. Such cues are not reported and cannot be used in a model; however, auto-regressive relations on past treatments can capture similar relations to give an effective prediction model. In other words, by modelling auto-regressive relations we assume more information goes into treatment decisions than is contained in the lice counts themselves.

Industry know-how suggests that the effectiveness of treatments is in the range of 90%; thus, our parameter estimates (approximately 30–40%) show less effectiveness than expected. Aldrin et al. (2019) also found this effect to be in a similar range to ours. There are two possible explanations to this large discrepancy. First, it may be due to a spurious relation between treatments and lice abundance. Treatments co-occur with increased lice

abundance but treatment is *not* the cause of more lice; rather, the opposite is true. Since it is difficult to separate the causal effect of treatments on lice abundance from their co-occurrence, the prediction power for lice abundance may improve by (counter-intuitively) lowering the effect of treatment. This is also the reason we constrain the sign of all parameters on the effect of treatment. Second, one may speculate whether lice can end up in the water before (or during) treatments and then return to the host afterwards. Even if the treatment itself has a 90% efficacy, the overall effect must be lower if the louse later returns to its host. Our model estimates would only reflect the overall effect of treatment, which should then also be lower than the expected 90% if lice later return to their hosts.

In future research, it may be advantageous to incorporate higher resolution stream patterns to account for known seasonal variations. This would require running longer hydrodynamic simulations, which is outside the scope of the current paper. Furthermore, future research may also validate the models using more precise machine vision counting which is currently being deployed in the industry. In principle, these could be applied directly without major alterations.

## References

- Abolofia, J., Asche, F., & Wilen, J. E. (2017). The Cost of Lice: Quantifying the Impacts of Parasitic Sea Lice on Farmed Salmon. *Marine Resource Economics*, *32*(3), 329–349. <https://doi.org/10.1086/691981>
- Albretsen, J., Sperrevik, A. K., Staalstrøm, A., Sandvik, A. D., Vikebø, F., & Asplin, L. (2011). NorKyst-800 Rapport nr. 1 : Brukermanual og tekniske beskrivelser. 48 s. Retrieved June 22, 2021, from <https://imr.brage.unit.no/imr-xmlui/handle/11250/113865>  
Accepted: 2011-07-13T11:53:51Z.
- Aldrin, M., Huseby, R., Stien, A., Grøntvedt, R., Viljugrein, H., & Jansen, P. (2017). A stage-structured Bayesian hierarchical model for salmon lice populations at individual salmon farms – Estimated from multiple farm data sets. *Ecological Modelling*, *359*, 333–348. <https://doi.org/10.1016/j.ecolmodel.2017.05.019>
- Aldrin, M., Jansen, P. A., & Stryhn, H. (2019). A partly stage-structured model for the abundance of salmon lice in salmonid farms. *Epidemics*, *26*, 9–22. <https://doi.org/10.1016/j.epidem.2018.08.001>
- Aldrin, M., Storvik, B., Kristoffersen, A. B., & Jansen, P. A. (2013). Space-Time Modelling of the Spread of Salmon Lice between and within Norwegian Marine Salmon Farms (M. Krkosek, Ed.). *PLoS ONE*, *8*(5), e64039. <https://doi.org/10.1371/journal.pone.0064039>
- Asplin, L., Albretsen, J., Johnsen, I. A., & Sandvik, A. D. (2020). The hydrodynamic foundation for salmon lice dispersion modeling along the Norwegian coast. *Ocean Dynamics*, *70*(8), 1151–1167. <https://doi.org/10.1007/s10236-020-01378-0>
- Bang Jensen, B., Qviller, L., & Toft, N. (2020). Spatio-temporal variations in mortality during the seawater production phase of Atlantic salmon (*Salmo salar*) in Norway. *Journal of Fish Diseases*, *43*(4), 445–457. <https://doi.org/10.1111/jfd.13142>
- Benjamin, M. A., Rigby, R. A., & Stasinopoulos, D. M. (2003). Generalized Autoregressive Moving Average Models. *Journal of the American Statistical Association*, *98*(461), 214–223. <https://doi.org/10.1198/016214503388619238>
- Czado, C., Gneiting, T., & Held, L. (2009). Predictive Model Assessment for Count Data. *Biometrics*, *65*(4), 1254–1261. <https://doi.org/10.1111/j.1541-0420.2009.01191.x>
- Dawid, A. P. (1984). Present Position and Potential Developments: Some Personal Views: Statistical Theory: The Prequential Approach. *Journal of the Royal Statistical Society. Series A (General)*, *147*(2), 278. <https://doi.org/10.2307/2981683>

- Elghafghuf, A., Vanderstichel, R., Hammell, L., & Stryhn, H. (2020). Estimating sea lice infestation pressure on salmon farms: Comparing different methods using multivariate state-space models. *Epidemics*, *31*, 100394. <https://doi.org/10.1016/j.epidem.2020.100394>
- Elghafghuf, A., Vanderstichel, R., St-Hilaire, S., & Stryhn, H. (2018). Using state-space models to predict the abundance of juvenile and adult sea lice on Atlantic salmon. *Epidemics*, *24*, 76–87. <https://doi.org/10.1016/j.epidem.2018.04.002>
- Fournier, D. A., Skaug, H. J., Ancheta, J., Ianelli, J., Magnusson, A., Maunder, M. N., Nielsen, A., & Sibert, J. (2012). Ad model builder: Using automatic differentiation for statistical inference of highly parameterized complex nonlinear models. *Optimization Methods and Software*, *27*(2), 233–249. <https://doi.org/10.1080/10556788.2011.597854>
- Hamre, L. A., Eichner, C., Caipang, C. M. A., Dalvin, S. T., Bron, J. E., Nilsen, F., Boxshall, G., & Skern-Mauritzen, R. (2013). The Salmon Louse *Lepeophtheirus salmonis* (Copepoda: Caligidae) Life Cycle Has Only Two Chalimus Stages. *PLOS ONE*, *8*(9), e73539. <https://doi.org/10.1371/journal.pone.0073539>
- Hamre, L. A., Bui, S., Oppedal, F., Skern-Mauritzen, R., & Dalvin, S. (2019). Development of the salmon louse *Lepeophtheirus salmonis* parasitic stages in temperatures ranging from 3 to 24°C. *Aquaculture Environment Interactions*, *11*, 429–443. <https://doi.org/10.3354/aei00320>
- Hellemo, L., Barton, P. I., & Tomasgard, A. (2018). Decision-dependent probabilities in stochastic programs with recourse. *Computational Management Science*, *15*(3-4), 369–395. <https://doi.org/10.1007/s10287-018-0330-0>
- Iversen, A., Hermansen, Ø., Nystøyl, R., & Hess, E. J. (2017). *Kostnadsutvikling i lakseoppdrett – med fokus på før- og lusekostnader*. Nofima AS. Retrieved December 5, 2020, from <https://nofima.brage.unit.no/nofima-xmlui/handle/11250/2481501>  
Accepted: 2018-02-01T14:47:27Z.
- Jansen, P. A., Kristoffersen, A. B., Viljugrein, H., Jimenez, D., Aldrin, M., & Stien, A. (2012). Sea lice as a density-dependent constraint to salmonid farming. *Proceedings of the Royal Society B: Biological Sciences*, *279*(1737), 2330–2338. <https://doi.org/10.1098/rspb.2012.0084>
- Jensen, E. M., Horsberg, T. E., Sevatdal, S., & Helgesen, K. O. (2020). Trends in de-lousing of Norwegian farmed salmon from 2000–2019—Consumption of medicines, salmon louse resistance and non-medicinal control methods. *PLOS ONE*, *15*(10), e0240894. <https://doi.org/10.1371/journal.pone.0240894>
- Jeong, J., Arriagada, G., & Revie, C. W. (2023). Targets and measures: Challenges associated with reporting low sea lice levels on Atlantic salmon farms. *Aquaculture*, *563*, 738865. <https://doi.org/10.1016/j.aquaculture.2022.738865>
- Jeong, J., & Revie, C. W. (2020). Appropriate sampling strategies to estimate sea lice prevalence on salmon farms with low infestation levels. *Aquaculture*, *518*, 734858. <https://doi.org/10.1016/j.aquaculture.2019.734858>
- Jonsbråten, T. W., Wets, R. J.-B., & Woodruff, D. L. (1998). A class of stochastic programs with decision dependent random elements. *Annals of Operations Research*, *82*(0), 83–106. <https://doi.org/10.1023/A:1018943626786>
- King, A. J., & Wallace, S. W. (2012, June). *Modeling with Stochastic Programming*. Springer New York, NY. <https://doi.org/10.1007/978-0-387-87817-1>
- Kristensen, K., Nielsen, A., Berg, C., Skaug, H., & Bell, B. (2016). TMB: Automatic differentiation and laplace approximation. *Journal of Statistical Software, Articles*, *70*(5), 1–21. <https://doi.org/10.18637/jss.v070.i05>
- Kristoffersen, A. B., Qviller, L., Helgesen, K. O., Vollset, K. W., Viljugrein, H., & Jansen, P. A. (2018). Quantitative risk assessment of salmon louse-induced mortality of seaward-migrating post-smolt Atlantic salmon. *Epidemics*, *23*, 19–33. <https://doi.org/10.1016/j.epidem.2017.11.001>
- Lovdata. (2008). Forskrift om drift av akvakulturanlegg (akvakulturdriftsforskriften) - Lovdata. Retrieved May 5, 2023, from <https://lovdata.no/dokument/SF/forskrift/2008-06-17-822>

- Lovdata. (2012). Forskrift om bekjempelse av lakselus i akvakulturanlegg - Lovdata. Retrieved September 8, 2021, from <https://lovdata.no/dokument/SF/forskrift/2012-12-05-1140>
- Lovdata. (2017, March). Forskrift om endring i forskrift om bekjempelse av lakselus i akvakulturanlegg - Lovdata. Retrieved July 5, 2023, from <https://lovdata.no/dokument/LTI/forskrift/2017-03-06-275>
- Lovdata. (2018, April). Forskrift om endring i forskrifter om akvakultur for deres tilpasning til transport, oppbevaring, bruk og produksjon av rensefisk - Lovdata. Retrieved July 5, 2023, from <https://lovdata.no/dokument/LTI/forskrift/2018-04-19-674>
- Myksvoll, M. S., Sandvik, A. D., Albretsen, J., Asplin, L., Johnsen, I. A., Karlsen, Ø., Kristensen, N. M., Melsom, A., Skardhamar, J., & Ådlandsvik, B. (2018). Evaluation of a national operational salmon lice monitoring system—From physics to fish. *PLOS ONE*, *13*(7), e0201338. <https://doi.org/10.1371/journal.pone.0201338>
- Samsing, F., Johnsen, I., Stien, L. H., Oppedal, F., Albretsen, J., Asplin, L., & Dempster, T. (2016). Predicting the effectiveness of depth-based technologies to prevent salmon lice infection using a dispersal model. *Preventive Veterinary Medicine*, *129*, 48–57. <https://doi.org/10.1016/j.prevetmed.2016.05.010>
- Shapiro, A., Dentcheva, D., & Ruszczyński, A. P. (2014). *Lectures on stochastic programming: Modeling and theory* (Second edition). Society for Industrial and Applied Mathematics : Mathematical Optimization Society. <https://doi.org/10.1137/1.9781611973433>
- Skern-Mauritzen, R., Sissener, N. H., Sandvik, A. D., Meier, S., Sævik, P. N., Skogen, M. D., Vågseth, T., Dalvin, S., Skern-Mauritzen, M., & Bui, S. (2020). Parasite development affect dispersal dynamics; infectivity, activity and energetic status in cohorts of salmon louse copepodids. *Journal of Experimental Marine Biology and Ecology*, *530–531*, 151429. <https://doi.org/10.1016/j.jembe.2020.151429>
- Solberg, I., Finstad, B., Berntsen, H. H., Diserud, O. H., Frank, K., Helgesen, K. O., Jeong, J., Kristoffersen, A. B., Nytrø, A. V., Revie, C. W., Sivertsgård, R., Solvang, T., Sunde, L. M., Thorvaldsen, T., Uglem, I., & Mo, T. A. (2018). *Kartlegging og testing av metodikk for telling av lakselus og beregning av luseforekomst* (tech. rep.). Norsk Institutt for Naturforskning (NINA). Retrieved July 20, 2022, from <https://brage.nina.no/nina-xmlui/handle/11250/2560950>  
Accepted: 2018-09-05T11:33:46Z.
- Startz, R. (2008). Binomial Autoregressive Moving Average Models With an Application to U.S. Recessions. *Journal of Business & Economic Statistics*, *26*(1), 1–8. <https://doi.org/10.1198/073500107000000151>
- Stien, A., Bjørn, P. A., Heuch, P. A., & Elston, D. A. (2005). Population dynamics of salmon lice *Lepeophtheirus salmonis* on Atlantic salmon and sea trout. *Marine Ecology Progress Series*, *290*, 263–275. <https://doi.org/10.3354/meps290263>
- Thorvaldsen, T., Frank, K., & Sunde, L. M. (2019). Practices to obtain lice counts at Norwegian salmon farms: Status and possible implications for representativity. *Aquaculture Environment Interactions*, *11*, 393–404. <https://doi.org/10.3354/aei00323>
- Walde, C. S., Jensen, B. B., Pettersen, J. M., & Stormoen, M. (2021). Estimating cage-level mortality distributions following different delousing treatments of Atlantic salmon (*salmo salar*) in Norway. *Journal of Fish Diseases*, *44*(7), 899–912. <https://doi.org/10.1111/jfd.13348>
- Walde, C. S., Stormoen, M., Pettersen, J. M., Persson, D., Røsæg, M. V., & Bang Jensen, B. (2022). How delousing affects the short-term growth of Atlantic salmon (*Salmo salar*). *Aquaculture*, *561*, 738720. <https://doi.org/10.1016/j.aquaculture.2022.738720>



## Paper IV

# Harvest Planning under Uncertainty in Salmon Aquaculture

Benjamin S. Narum, Julio C. Goetz and Stein W. Wallace

### Abstract

We address operational risk management in aquaculture using multistage stochastic programming to address biological, operational and market risk. The model is applied to Norwegian salmon aquaculture, and we find that decision support tools provide valuable insights across a multitude of different situations in a complex environment. Our results show there is considerable value in actively incorporating consideration of uncertainty into harvest planning.

## 1 Introduction

Harvest planning in Norwegian salmon aquaculture faces considerable biological, operational and market uncertainty while being subject to regulatory constraints and operational limitations. Large capital investments in the stock also means the downside risk is considerable. Each farming company has a large portfolio of heterogeneous sites which adds to the complexity of planning. Combined, these factors make the *Aquaculture Harvest Planning* (AHP) problem well suited for application of quantitative decision support models that incorporate uncertainty. The paper presents the modelling considerations and solution procedures used to address the planning problem by multistage stochastic programming, and aims to investigate whether decision support models can aid in managing the complexity aquaculture farmers face. Our contribution lies in addressing an existing and well established logistical problem by applying new techniques. Harvest planning models to this level of detail is a novel contribution to the literature.

### 1.1 Background

Salmon aquaculture consists of rearing fish in ocean-submerged cages located at sites along the coastline. Typically, each site contains 1–12 cages. Fish are hatched on land

and transferred to the ocean at a weight of approximately 0.1–0.7 kg and grown for 12–18 months until harvested at 3–8 kg, not necessarily by harvesting the entire site at once. The fish are directly exposed to the ocean and are affected especially by temperature, parasitic lice and spread of disease, the latter two being the major biological sources of uncertainty which spread between sites. Lice is a particular challenge, and during increased levels of lice, farmers must conduct costly treatments that are also harsh on fish health (Walde et al., 2021, 2022). This is a major concern for the industry and has an estimated cost of around 9% of annual revenue (Abolofia et al., 2017; Iversen et al., 2017). In an effort to mitigate the lice problem, farmers are subject to regulatory requirements to perform delousing treatments, to fulfil limitations on maximal total biomass (MTB), and to close sites for eight weeks between production cycles (Lovdata, 2012). Spot prices for salmon are highly volatile, making farmers vulnerable to a sudden requirements to harvest, either due to health considerations or regulatory constraints. Most of companies’ capital is bound up in the currently standing biomass, meaning they have large risk exposure in case of large mortality events. Simultaneous occurrence of unforeseen disease and lice treatments are known to be especially unfortunate. Historically, the mortality rate per fish cohort is 14–17% on average (25% mortality at the 75th percentile, 40 % mortality at the 97.5th percentile, and 60–80% mortality at the maxima) as investigated by Bang Jensen et al. (2020). The current state of harvest planning in the industry is dominated by domain expertise applied to manual setup of schedules supported by spreadsheet models. Common rules-of-thumb include harvesting the largest fish first and simplifications to optimise for total harvested biomass instead of profit (which we investigate in Section 4).

The AHP problem is an inherent sequential decision problem under uncertainty where timing and sequencing are the essential considerations to be made. Harvest decisions are irreversible (since fish cannot be de-slaughtered) meaning myopic approaches are expected to be sub-optimal, and precise account of the future value of standing biomass is required. We pose the harvest planning problem as a stochastic multistage program to effectively deal with uncertainty, consideration of portfolio effects, operational details and regulation. We chose this methodology to effectively deal with constraints, to encode the logic of harvest operations using integer variables, for interpretability, and to account for far-reaching temporal dependencies in the uncertain quantities. In general, strong and relevant modelling assumptions are required to solve sequential decision problems under uncertainty (Powell, 2019) and in their basic form, their complexity grows at rates that can be considered computationally intractable to solve (Shapiro & Nemirovski, 2005). Hence, we apply extensive out-of-sample validation techniques to ensure we obtain good solutions with respect to the approximations required for computational tractability.

## 1.2 Literature

Forsberg and Guttormsen (2006) performed an early study that confirmed there is value in information of future prices for harvest plans, but disregard operational limitations. Oglend and Tveteras (2009) find that spatial proximity of sites has correlated risk exposure and should be diversified; however, this is only actionable on the strategic level of site acquisition. It is also known that companies have acquired smaller companies partly as a risk mitigation strategy (Asche et al., 2013). Engehagen et al. (2021) analyse the risk of algal blooms in aquaculture by real options posed as an optimal timing problem for when to harvest. Pettersen et al. (2015) found that early harvest triggered by observed spread of disease is economically beneficial but also that this is sensitive to other factors. Schütz and Westgaard (2018) analyse risk-aversion by multistage stochastic programming considering market risk, but simplify much of the operations. A series of master’s theses (Denstad

et al., 2015; Hæreid, 2011; Hornsletten, 2017; Langan & Toftøy, 2011) address harvest planning under uncertainty but face tractability issues which limit the scope of planning considerably. Meanwhile, this paper addresses harvest planning up to the effective end-of-horizon, which is needed to reflect site heterogeneity. Føsund and Strandkleiv (2021) solve a deterministic problem of similar structure by decomposition and achieve significantly improved solution times. This implies similar decomposition procedures can be effective for the stochastic problem as well.

A fundamental difference between agriculture and aquaculture is that aquaculture farmers are in less control of the surroundings of the livestock, which is a major source of uncertainty. Models incorporating uncertainty have increasingly been applied to agriculture, and the review by Borodin et al. (2016) concludes that stochastic programming problems are dominated by two-stage models with recourse or chance constraints with very few real-world implementations of multistage models. Furthermore, those that exist address only a single source of either market or production risk. Dowson et al. (2019) provide a good example of multistage models in agriculture where they apply Stochastic Dual Dynamic Programming (SDDP) to dairy production. We believe the modelling techniques in this work are applicable to other operational planning problems, like in agriculture. Characterising traits could then include: managing stock with some associated biological risk, time developing capacity constraints, irreversible decisions to sell the stock to market, and consideration of both sequencing and timing.

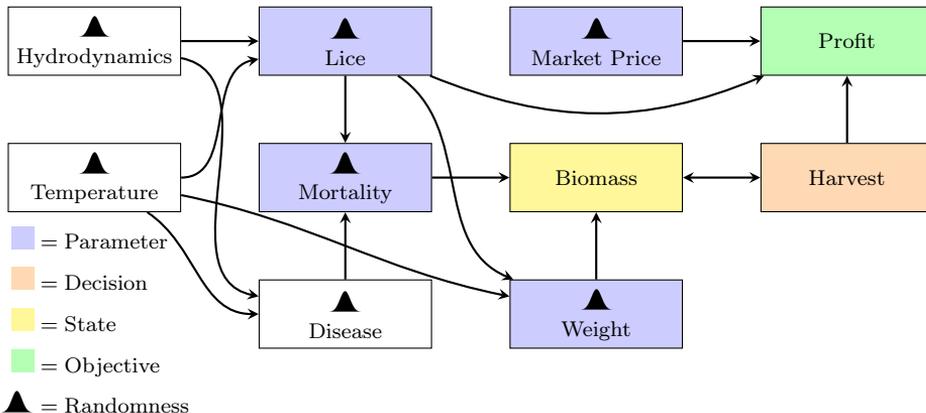
## 2 Problem statement and modelling considerations

The AHP problem represents a highly complex sequential decision-making problem under uncertainty. The planning horizon is generally infinite since sites are re-stocked continuously, and these have different startup dates to ensure consistent supply. Decisions are discrete in nature since an important driver of costs comes from hiring well-boats to transport the harvested fish. There is significant uncertainty in the biological state of the fish, and the operational cost of handling biological challenges is also considerable. Biological factors are based on statistical models of complicated non-linear relations with long-term dependence on history (up to 20 weeks for lice development) which prevents approximation by assuming Markovian time-development. The industry is under regulatory constraints to limit its total standing biomass, which means harvest decisions are mainly triggered by limiting constraints rather than reaching optimal harvest weights based on growth and feed costs. Since harvest decisions are driven by company-wide limiting constraints, all sites must be considered simultaneously to make a coherent harvest plan. There is large heterogeneity between sites, both due to offsets in when fish are deployed and in the exposure to biological risk, which means they all contribute their own individual considerations to the overall planning. While biological risk affects planning on a medium-term time horizon (3–10 weeks), we also have very high short-term price volatility (within 1–2 weeks) that affects the materialised profit, as well as biomass limitations that are relevant for both short- and long-term planning (1–80 weeks). In this section, we present the modelling considerations that allow solving of the problem numerically while still capturing its essential characteristics.

### 2.1 Modelling considerations

A characteristic of major importance is that the model must capture how fish develop over time, and must replicate how harvest decisions alter the state of the portfolio. We for-

mulate this concisely by representing the state in terms of fish count  $y$ , where all aspects such as growth, weight, health, mortality, and prices, are built into stochastic coefficients multiplied by  $y$  to give linear expressions of  $y$ . A great advantage of this formulation is that we can have arbitrarily complicated models to determine these coefficients, based on domain knowledge and forecasting models of high complexity. Still, we have a linear decision model that allows solving problems of large scale. Furthermore, we may encode logic that alters the state  $y$  through constraints using binary variables. Since fish count never increases (as opposed to weight), the initial fish count provides a tight big-M constant. One caveat of having complicated relations in the stochastic coefficients is the challenge of representing their dependence pattern within a scenario tree.

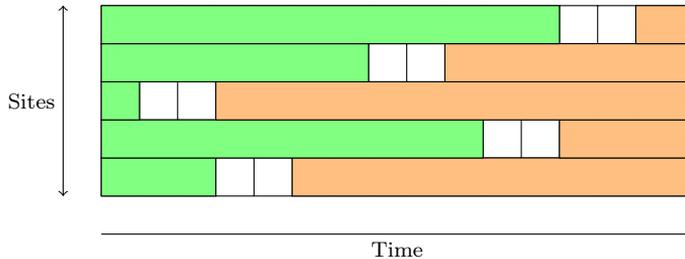


**Figure IV.1:** Overview of uncertainty, and relationships as reflected by the model.

All sources of uncertainty can be summarised by four stochastic parameters within the decision model: (i) weight, (ii) mortality, (iii) lice treatment, and (iv) market prices. There are causal mechanisms driving these parameters, as well as unresolved random components. Figure IV.1 illustrates an overview of the relations between all uncertain factors. The main exogenous drivers are temperature and hydrodynamics, which determine development of lice and disease, as well as the growth rate of fish. Lice treatments and disease are the most important drivers of biological risk since they affect mortality rates. Lice treatments are forecasted but we simplify incorporation of disease by deriving the mortality distribution conditional on lice treatment but unconditional on the effect of disease. This means the upper tail of the mortality distribution is heavier due to the effect of disease. The price model is based on first deriving a smoothed trend from historical data using a wide-spanning moving average; this simplification is used to represent domain knowledge and expert forecasts that are hard to replicate but normally available to farmers. The precision of these price trends may be slightly optimistic in assuming better information than is truly the case. Surrounding this trend, we represent variability in prices as an auto-regressive process of log-transformed prices having three weeks of memory. Detailed descriptions of the statistical models are given in the Supplemental Material.

Harvest planning is mainly a timing problem, but consideration of an entire portfolio also makes it a sequencing problem of combinatorial nature. The decision to harvest is a discrete decision due to its initiation cost and harvesting at one site can alleviate the need to harvest at a different site by fulfilling total biomass limitations. We must avoid decisions to harvest very small quantities each time since it is mainly the initiation that

drives costs by ordering a well-boat. Still, we assume some operational flexibility once the decision to harvest has been made through hourly pay and overtime work. Hence, we let the number of boats used for harvests take values in  $\{0\} \cup [1, \infty)$ , encoded by one binary and one continuous variable instead of an integer variable. This reflects the lump cost of initiating harvests, but alleviates the complicated combinatorics that arise from determining integer boat counts for each harvest operation.



**Figure IV.2:** Horizon with respect to currently active sites. The green segments represent the relevant sites, while the orange segments represent re-opened sites which contribute to the total biomass but have no decision variables. Sites are required to be closed for eight weeks between production cycles, as indicated by the spacing between closing and re-opening dates.

A fundamental aspect of operational planning is to determine the *alternative cost of harvesting*, given by the conditional expected value of the current fish stock. Value estimates are sensitive to the length of the planning horizon, and large heterogeneity between sites together with portfolio effects make end-of-horizon value approximations difficult. The horizon of the problem is generally infinite, but very rarely is it profitable to harvest small fish. As an approximation, we consider decision variables only up to the stage where all currently active sites must be emptied (see Figure IV.2). Re-opened sites cannot be ignored since they still contribute to biomass limitations; instead, these are simplified using linear harvest schedules to reflect their biomass contribution. The decision problem can then be solved as a finite horizon problem that explicitly considers the future value of each site up to their effective end-of-horizon. This is deployed in a rolling horizon environment that progressively adds more sites once they are relevant.

The planning problem is aggregated to consider entire sites instead of individual cages. We base this on the fact that heterogeneity between fish populations is mainly a result of spatial location and time of deployment. Cages within a site are usually re-stocked simultaneously not to delay further startup dates.

Decision-dependent uncertainty (Jonsbråten et al., 1998) can cause challenges in solving stochastic programming models due to difficulties in representing this kind of uncertainty in a concise manner (Apap & Grossmann, 2017; Hellemo et al., 2018). Technically, the problem at hand must address *when to treat* for lice and such treatments affect growth, mortality and lice development; thus, introducing decision-dependent uncertainty. To alleviate the computational challenges implied by decision-dependence, we instead model treatments as an exogenous stochastic process. This simplification is based on the observation that while harvest plans may depend on treatments (by harvesting in advance of costly treatments), treatments do not depend as strongly on harvest decisions. Namely, treatments are more strongly driven by lice since they are required by regulation once the lice level surpasses a certain limit. By predicting future levels of lice, we may also predict the need for treatment. Furthermore, the primary purpose for delaying a treatment would be to harvest first, but since there is flexibility in harvest timing, the option

to shift the relative timing of these two actions is still present. The forecasting model for lice development and treatments is described in detail in a separate publication (Narum & Berentsen, 2024).

### 3 Methodology and mathematical formulation

We present the mathematical formulation of the AHP problem and the methodology for validating its performance in terms of a prescribed stochastic process. The formulation is presented in Section 3.2, while Section 3.3 elaborates on the validation procedure. For all practical purposes, we solve the optimisation problem on a discrete scenario tree to approximate uncertainty. It is still valuable for validation purposes to think about the optimisation problem in terms of any stochastic process, a discussion left for Section 3.3 after the formulation has been presented. Scenario tree generation is described in Section 3.4. First, we explain the notation for stochastic processes and multistage stochastic programs.

#### 3.1 Notation on stochastic processes and multistage stochastic programs

Stages are indexed by a finite number of discrete steps  $t \in \mathcal{T} = \{1, \dots, T\}$  where  $T$  is the last stage. We let  $\omega$  denote an *outcome path* (commonly also referred to as a scenario) and let  $\Omega$  denote the collection of all possible outcome paths. One may think of  $\omega$  as a label (like  $1, 2, \dots$ ) used to refer to specific outcomes. Letting  $Y$  represent the stochastic process of some uncertain quantity (like a number, vector or matrix), we index these by stage  $t \in \mathcal{T}$  and outcome path  $\omega \in \Omega$  using the notation  $Y_t(\omega)$ . For simplicity, we may omit reference to  $\omega$  when it is understood from the context. The *history* of  $Y(\omega)$  up to stage  $t$  is denoted  $Y_{[t]}(\omega) = (Y_1(\omega), \dots, Y_t(\omega))$ .

Outcome paths are subject to a causal *information structure* across all stages  $\mathcal{T}$  which is denoted  $\mathcal{F} = (\mathcal{F}_t)_{t \in \mathcal{T}}$ . The information structure describes whether outcome paths are *distinguishable*; indeed, the ability to distinguish outcome paths from each other *is* information about what may or may not happen as we go forward in time. In terms of a stochastic quantity  $Y$ , we have that  $Y_t(\omega) = Y_t(\omega')$  whenever  $\omega$  and  $\omega'$  are indistinguishable in stage  $t$ . We assume all stochastic quantities must adhere to this information structure and that all stage-specific quantities  $Y_t$  can be known in stage  $t$ . As long as we do optimisation, it is essential that uncertainty is represented in a way to reflect that information is incrementally gained over time, and outcomes  $\omega \in \Omega$  are fundamentally linked through the information structure. This is achieved using scenario trees to represent uncertainty. Probabilities are determined by the distribution  $P$  which is described by statistical models of the respective uncertain phenomena. Let

$$\mathbb{E}_P^t [\cdot],$$

denote the expectation using distribution  $P$  conditional on information in stage  $t$ . For simplicity, we use the abbreviation  $\mathbb{E}^t [\cdot]$  when the distribution is left implicit, and let  $\mathbb{E} [\cdot]$  denote the unconditional expectation. Note that  $\mathbb{E}^t [\cdot](\omega)$  is a random variable in stage  $t$  that can also be specified by  $\omega$ . The problem may be defined in terms of different distributions as described by a scenario tree ( $R$ ), the distribution ( $P$ ) or by sampling paths ( $\dot{P}$ ) used for evaluation, which is why we leave the distribution implicit in the formulation.

Decisions are considered to be *policies* that may change according to information states in  $\mathcal{F}$ . We let  $\xi$  denote the stochastic process that contains all random parameters and  $x$  the decision policy that contains all decision variables. The objective to be maximised in each stage  $t$  is denoted  $f_t(x_t, \xi_t)$  and we let

$$F(x, \xi) = \sum_{t \in \mathcal{T}} f_t(x_t, \xi_t), \quad (\text{IV.1})$$

denote the overall objective. The overall optimisation problem can be expressed as

$$\max_{x \in \mathcal{X}} \{\mathbb{E} [F(x, \xi)]\}, \quad (\text{IV.2})$$

where  $\mathcal{X}$  is the feasible set, which includes the requirement that the policy must adhere to the given information structure (also known as non-anticipativity).

### 3.2 Mathematical formulation

The mathematical formulation aims to decide when and where to harvest in order to maximise profits under stochastic state development. The validation procedure (described in Section 3.3) requires relatively complete recourse, achieved using soft-constraints for all constraints that contain stochastic parameters. An overview of affecting factors are illustrated in Figure IV.1 and a summary of all variables, parameters and sets is given in Table IV.1. The formulation is stated in a stage-wise fashion, using indices  $t$  to denote stages and indices  $i$  to denote sites. For simplicity, we omit explicit reference to outcome paths  $\omega$ , and let constraints be defined to hold for all  $\omega \in \Omega$  (almost surely with respect to the distribution). The information structure and the distribution is defined by the scenario tree used to solve the formulation.

We use weekly resolution where each stage is indexed by  $t \in \mathcal{T}$  and locations by  $i \in \mathcal{L}$  where  $\mathcal{L}$  is the set of all sites. Let  $\mathcal{A}_t \subset \mathcal{L}$  denote the set of decision-relevant sites in stage  $t$ , which can be partitioned into

$$\mathcal{C}_t := \mathcal{A}_t \setminus \mathcal{A}_{t+1}, \quad \mathcal{O}_t := \mathcal{A}_t \cap \mathcal{A}_{t+1}, \quad (\text{IV.3})$$

to denote sites that *must* be closed by the end of stage  $t$  and those that *may* stay active after stage  $t$ . These are inferred from pre-determined stocking decisions, while decision variables ultimately determine when a site is closed. Sites that may stay active ( $\mathcal{O}_t$ ) can also be closed sooner, but there always exists a stage where a given site must be closed (as determined by  $\mathcal{C}_t$ ).

Stochastic parameters are either coefficients or right-hand side constants in constraints. We let  $w_{it}(\omega)$  denote the average fish weight at the end of stage  $t$  within site  $i$ , and

$$\Delta w_{it}(\omega) := w_{it}(\omega) - w_{i,t-1}(\omega), \quad (\text{IV.4})$$

the corresponding change in weight during stage  $t$ . We also let  $\sigma_{it}(\omega)$  denote the survivability going into stage  $t$ . Fish count can never increase, meaning we can infer the maximal in-going fish count as

$$M_{it}(\omega) := y_{i0} \prod_{s=1}^t \sigma_{is}(\omega), \quad (\text{IV.5})$$

where  $y_{i0}$  is the initial in-going fish count. Let  $\psi_{it}(\omega)$  denote a binary stochastic variable for whether a treatment is commenced in stage  $t$ . Lastly, let  $\pi_{it}(\omega)$  denote the sales price of

**Table IV.1:** Overview of sets, variables and parameters. Indices  $i$  refer to sites, and  $t$  to stages.

Sets	
$\mathcal{T}$	Set of stages
$\mathcal{L}$	Set of sites
$\mathcal{A}_t$	Set of decision-relevant sites in stage $t$
$\mathcal{C}_t$	Subset of sites that must be closed during stage $t$
$\mathcal{O}_t$	Subset of sites that remain active after stage $t$
$\tilde{\mathcal{A}}_t$	Locations that have been re-opened during the planning horizon
Primal variables	
$y_{it}$	Number of out-going fish
$y_{it}^\circ$	Binary for whether there is out-going fish
$h_{it}$	Number of fish harvested
$h_{it}^\circ$	Binary for whether to harvest
$b_{it}$	Number well-boats employed
$z_t^{\text{bfix}}$	Violation of boat availability up to additional flexibility
$z_t^{\text{blmt}}$	Violation of boat availability beyond additional flexibility
$z_t^{\text{cmp}}$	Violation of company-wide biomass limit
Stochastic parameters	
$\sigma_{it}$	Survival rate of in-going fish
$w_{it}$	Average weight at the end of week $t$
$\pi_{it}$	Revenue per kg harvested at weight $w_{it}$
$\psi_{it}$	Binary for whether delousing is performed
Parameters	
$y_{i0}$	Initial fish count
$c^{\text{feed}}$	Cost of feed per weight increase
$c^{\text{slaughter}}$	Cost of slaughtering per weight
$c^{\text{boat}}$	Cost of well-boat
$c^{\text{delouse}}$	Cost of delousing
$c^{\text{active}}$	Cost of keeping a site active
$r^{\text{hog}}$	Head-over-gut ratio
$P^{\text{boat}}$	Penalty for violating boat capacity
$P^{\text{cmp}}$	Penalty for violating the company biomass limit
$P^{\text{loc}}$	Penalty for violating the location biomass limit
$C^{\text{boat}}$	Weight-capacity of well-boats
$N^{\text{boat}}$	Number of available well-boats
$B^{\text{cmp}}$	Biomass limitation of the company
$B_i^{\text{loc}}$	Biomass limitation of location $i$
$\tilde{y}_{it}$	Fish count at re-opened sites

fish where the weight  $w_{it}(\omega)$  determines its price class. Recall that  $\xi = (w, \Delta w, \sigma, \psi, \pi, M)$  denotes the collection of these stochastic quantities. We summarise the distributions and estimation procedures for all stochastic parameters in the Supplementary Material.

The state variable  $y_{it}$  denotes *out-going* fish count at site  $i$  in stage  $t$ , while the decision variables  $h_{it}$  denote harvest count and  $b_{it}$  denotes boat count. These decision variables may alter the state  $y_{it}$  through constraints. We have that  $\sigma_{it}y_{i,t-1}$  denotes the in-going fish count, and since the realisation of mortality is observed on in-going fish, we cannot know the effect of mortality until *after* decisions in a given stage  $t - 1$  have been determined and the state  $y_{i,t-1}$  is being passed to the next stage. This pessimistically assumes less knowledge about mortality than might be the case.

We use binary variables  $y_{it}^\diamond$  to reflect *whether there is out-going fish*, and  $h_{it}^\diamond$  to reflect *whether to initiate a harvest*. To account for the lump cost of harvesting, we require that boat count  $b_{it}$  is contained in the set  $\{0\} \cup [1, \infty)$ , encoded using the binary variable  $h_{it}^\diamond$ . This prevents getting solutions that continuously harvest small quantities in every stage while alleviating the combinatorial complexity that would result from having integer valued boat counts.

To make soft-constraints, which ensure solutions are always feasible for any realisation of uncertainty, we let  $z_t^{\text{cmp}}$  be an auxiliary variable for violation of the company-wide biomass limitation. Meanwhile, the location-wise biomass limitation is strictly enforced. Furthermore, we let  $z_t^{\text{bflx}}$  and  $z_t^{\text{blmt}}$  be auxiliary variables for violation of boat availability which has a progressive penalty.

The objective to be maximised in each stage  $t$  is stated as:

$$\begin{aligned} f_t(x_t, \xi_t) = & \sum_{i \in \mathcal{A}_t} (\pi_{it} r^{\text{hog}} - c^{\text{slaughter}}) w_{it} h_{it} - c^{\text{boat}} b_{it} \\ & - (c^{\text{active}} + c^{\text{delouse}} \psi_{it}) y_{it}^\diamond - \sum_{i \in \mathcal{O}_t} c^{\text{feed}} \Delta w_{it} y_{it} \\ & - (c^{\text{boat}}/2) z_t^{\text{bflx}} - P^{\text{boat}} z_t^{\text{blmt}} - P^{\text{cmp}} z_t^{\text{cmp}}. \end{aligned} \quad (\text{IV.6})$$

These terms incorporate: sales revenue, well-boat cost, variable cost of running an active site, delousing cost and feed cost. Then there are the penalties associated with soft-constraints for boat availability and for biomass restrictions. Feed cost incurs on out-going biomass with the interpretation that the fish grow during the week, but not if they are harvested. The same goes for delousing and the variable cost of running a site.

The constraints are formulated as follows:

- Fish count balance

$$y_{it} + h_{it} = \sigma_{it} y_{i,t-1}, \quad (\phi_{it}) \quad \forall i \in \mathcal{O}_t, \quad (\text{IV.7a})$$

$$h_{it} = \sigma_{it} y_{i,t-1}, \quad (\phi_{it}) \quad \forall i \in \mathcal{C}_t. \quad (\text{IV.7b})$$

We use an equality constraint since fish must be harvested to be removed from a site. This prevents discarding fish in situations where the cost of operations is higher than the value of the fish. In the last stage before closing a site, all available fish must be harvested.

- Logic of binary variables

$$y_{it} \leq M_{it} y_{it}^\diamond, \quad (\nu_{it}^y) \quad \forall i \in \mathcal{O}_t, \quad (\text{IV.8})$$

$$h_{it} \leq M_{it} h_{it}^\diamond, \quad (\nu_{it}^h) \quad \forall i \in \mathcal{A}_t, \quad (\text{IV.9})$$

where the maximal in-going fish count  $M_{it}$  is used as a big-M parameter.

- Harvesting capacity

$$w_{it}h_{it} \leq C^{\text{boat}}b_{it}, \quad (\kappa_{it}^C) \quad \forall i \in \mathcal{A}_t, \quad (\text{IV.10})$$

$$h_{it}^\diamond \leq b_{it}, \quad (\kappa_{it}^{h^\diamond}) \quad \forall i \in \mathcal{A}_t, \quad (\text{IV.11})$$

$$\sum_{i \in \mathcal{A}_t} b_{it} \leq N^{\text{boat}} + z_t^{\text{bflx}}, \quad (\kappa_t^{\text{bflx}}) \quad (\text{IV.12})$$

$$\sum_{i \in \mathcal{A}_t} b_{it} \leq 2N^{\text{boat}} + z_t^{\text{blmt}}, \quad (\kappa_t^{\text{blmt}}) \quad (\text{IV.13})$$

Here, (IV.10) ensures the capacity of each well-boat is respected, while (IV.11) encodes that at least one boat must be utilised once harvesting is initiated. Boat availability is limited to  $N^{\text{boat}}$  but we assume there is some additional flexibility that enables getting up to twice as many boats at a 50% price premium using the auxiliary variable  $z_t^{\text{bflx}}$ . Beyond this point, violation is more heavily penalised by the parameter  $P^{\text{blmt}}$  using auxiliary variable  $z_t^{\text{blmt}}$ .

- Biomass restrictions

$$\sum_{i \in \mathcal{O}_t} w_{it}y_{it} \leq (B^{\text{cmp}} - \sum_{i \in \tilde{\mathcal{A}}_t} w_{it}\tilde{y}_{it}) + z_t^{\text{cmp}}, \quad (\beta_t^{\text{cmp}}) \quad (\text{IV.14})$$

$$w_{it}y_{it} \leq B_i^{\text{loc}}, \quad (\beta_{it}^{\text{loc}}) \quad \forall i \in \mathcal{O}_t. \quad (\text{IV.15})$$

For all practical purposes, biomass restrictions are strict, but we still need to resolve (out-of-sample) infeasibility using soft-constraints. Constraint (IV.15) can always be respected by requiring higher harvest count, while the company-wide constraint (IV.14) is penalised by the parameter  $P^{\text{cmp}}$  using auxiliary variable  $z_t^{\text{cmp}}$ .

- Variable ranges

$$0 \leq y_{it}, \quad \forall i \in \mathcal{O}_t, \quad (\text{IV.16})$$

$$0 \leq h_{it}, b_{it}, \quad \forall i \in \mathcal{A}_t, \quad (\text{IV.17})$$

$$0 \leq z_t^{\text{bflx}}, z_t^{\text{blmt}}, z_t^{\text{cmp}}. \quad (\text{IV.18})$$

- Integer restrictions

$$y_{it}^\diamond, h_{it}^\diamond \in \{0, 1\}, \quad \forall i \in \mathcal{A}_t. \quad (\text{IV.19})$$

We also add the tightening constraints

$$y_{it}^\diamond \leq y_{i,t-1}^\diamond, \quad (\tau_{it}^{y^\diamond}) \quad \forall i \in \mathcal{O}_t, \quad (\text{IV.20})$$

$$h_{it}^\diamond \leq y_{i,t-1}^\diamond, \quad (\tau_{it}^{h^\diamond}) \quad \forall i \in \mathcal{A}_t, \quad (\text{IV.21})$$

to relate binary variables across stages.

The penalties  $P^{\text{blmt}}$  and  $P^{\text{cmp}}$  associated with constraints (IV.13) and (IV.14) are set as low as possible while still preventing infeasibility (with respect to soft-constraints) in the optimisation problem. The reason to have these low is that out-of-sample evaluation will almost certainly lead to infeasibility (with respect to soft-constraints) and if penalties are too large, they will have a disproportionate effect on objective evaluations. Excessive penalisation in soft-constraints do not necessarily represent operations since the model

would likely be re-run in a rolling horizon environment that better reflects conditional information; hence, we aim to set penalties low. The penalty  $P^{\text{blmt}}$  is set according to the maximal value a well-boat can carry within its capacity, so that violation of well-boat availability will cancel out profits. The penalty  $P^{\text{cmp}}$  is set to the highest possible value gain from one stage to the next based on the highest conceivable growth rate and price increase. This is derived from historical price volatility where truncation of the noise process to a 99% prediction interval allows deriving an absolute bound. These penalties ensure it is unprofitable to violate either soft-constraint. One exception is if the total stock is too large to be harvested within the required closing date without violating biomass constraints or well-boat availability, in which case any penalty below  $+\infty$  would be too low; this is ignored, and we let losses be limited to a bit more than cancellation of positive profit.

### Shadow price of fish stock

In a broader sense, the AHP problem aims to find the alternative cost of harvesting today by estimating the future value of fish stock with respect to time dynamics, uncertainty and operational flexibility. Some interesting conclusions can be made in this regard by analysing the dual formulation of the LP relaxation. Dual variables are listed as constraint multipliers in the above primal formulation, and the complete dual formulation is provided in the Supplementary Material as a reference. We refer to Rockafellar (1999) for a more general discussion on the time structure of multistage stochastic programs and their duals (particularly the conditional expectations found in dual constraints). We now address some specific aspects of the dual.

Consider first that the dual variable  $\phi_{it}(\omega)$  associated to constraint (IV.7) can be interpreted as the *shadow price of in-going fish stock*, which quantifies the marginal value of inserting fish in the state described by site  $i$ , stage  $t$  and outcome  $\omega$ . Furthermore,  $\phi_{i,t=1}$  (which is deterministic) denotes the initial marginal value of fish stocks at each site.

Through complementarity slackness, we can make qualitative observations of when a given decision variable should be zero; if a dual constraint is non-binding, its associated primal variable must be zero, while if the dual constraint is binding, its associated primal variable can be non-zero (Bertsimas & Tsitsiklis, 1997). Specifically, we want to address when to harvest ( $h_{it} > 0$ ) and whether to keep a site active ( $y_{it} > 0$ ). Among other terms, the dual formulation aims to minimise the initial in-going value  $\sum_{i \in \mathcal{A}_t} y_{i0} \sigma_{i,t=1} \phi_{i,t=1}$ , which means  $\phi$  will generally be as low as possible.

The decision to harvest in any given state  $h_{it}$  can be inferred from its associated dual constraint,

$$\phi_{it} \geq (\pi_{it} r^{\text{hog}} - c^{\text{slaughter}}) w_{it} - \nu_{it}^{h^\circ} - w_{it} \kappa_{it}^C =: \theta_{it}, \quad (h_{it}) \quad \forall i \in \mathcal{A}_t, \quad (\text{IV.22})$$

where  $\phi_{it}$  is the state value of fish stock which must be larger or equal to its sales revenue  $\pi_{it} r^{\text{hog}} w_{it}$  minus slaughtering costs  $c^{\text{slaughter}} w_{it}$  and the cost of well-boat operations  $\nu_{it}^{h^\circ} + w_{it} \kappa_{it}^C$ . We may only harvest ( $h_{it} > 0$ ) when (IV.22) is binding. We let  $\theta_{it}$  denote the right-hand side expression, interpreted as the current sales profit. The decision to hold onto fish stock for longer ( $y_{it} > 0$ ) can be inferred by its dual constraint

$$\phi_{it} \geq \mathbb{E}^t [\sigma_{i,t+1} \phi_{i,t+1}] - c^{\text{feed}} \Delta w_{it} - \nu_{it}^{y^\circ} - w_{it} (\beta_t^{\text{cmp}} + \beta_{it}^{\text{loc}}), \quad (y_{it}) \quad \forall i \in \mathcal{O}_t, \quad (\text{IV.23})$$

where the current state value  $\phi_{it}$  must be larger or equal to the conditional expected value of future states  $\mathbb{E}^t [\sigma_{i,t+1} \phi_{i,t+1}]$  (discounted by mortality) minus feed costs  $c^{\text{feed}} \Delta w_{it}$ ,

the variable cost of keeping a site active  $\nu_{it}^{y^\circ}$  (which includes lice treatments) and the operational cost of biomass constraints  $w_{it}(\beta_t^{\text{cmp}} + \beta_{it}^{\text{loc}})$ . Observe that we may only hold onto fish stock ( $y_{it} > 0$ ) while (IV.23) is binding. Most of the time, (IV.23) will be binding as long as there are future states that are more valuable than the current state, while at these more valuable states, (IV.22) becomes binding to indicate that harvesting should be initiated. Observe also that (IV.23) connects state values  $\phi_{it}$  across stages, and that this is the only dual constraint to do so. Based on this relation, we may think of dual states as propagating backwards in time due to the conditional expectation of future dual states (Rockafellar, 1999). On the contrary, primal states propagate forward by being inferred by past states.

We can conclude from the above dual constraints that state values ( $\phi$ ) are determined by the highest future sales value (from (IV.22)), discounted by weekly costs going backwards in time (from (IV.23)). Observe also that limitations on biomass and well-boat availability contribute to this discounting; hence, we incorporate future operational flexibility into value estimates. For the model to hold onto fish stocks, constraint (IV.23) must be binding, and the current state value must correspond to the conditional expected future value, minus weekly costs. If the estimate of future state values  $\mathbb{E}^t[\sigma_{i,t+1}\phi_{i,t+1}]$  is too low, we must have that  $y_{it} = 0$ . For the model to decide to harvest, the current state value  $\phi_{it}$  must be binding on the sales profit  $\theta_{it}$ . The only reason (IV.22) may be non-binding (so that  $h_{it} = 0$ ) is if there exists more valuable future value states whose cost discounting is lower than the value gain. These observations are examined numerically in Section 4.4.

### 3.3 Out-of-sample validation

A fundamental challenge in multistage stochastic programming is knowing whether a scenario tree represents a prescribed stochastic process to sufficient precision. To ensure good performance, we use an out-of-sample validation scheme based on extending an optimal decision policy from a scenario tree, referred to as an *extension policy*. This validation approach was explored by Casey and Sen (2005) and Keutchayan et al. (2017). See also (Narum et al., 2023, Section 4) for a review. Furthermore, we parameterise scenario tree generation and tweak its parameters through surrogate optimisation to improve out-of-sample performance (described in Section 3.4).

The validation approach is based on providing a feasible policy that is defined for any outcome path  $\omega \in \Omega$ , used to evaluate the objective value. If  $\tilde{x}$  is a feasible primal policy and  $x^*$  is an optimal policy, we have that

$$\mathbb{E}[F(\tilde{x}, \xi)] \leq \mathbb{E}[F(x^*, \xi)]. \quad (\text{IV.24})$$

Optimal policies are generally unavailable, but feasible policies can more easily be constructed. Once a feasible policy is defined, we can make a sampling estimate of its expected objective value by sampling outcome paths from the prescribed stochastic process to evaluate its performance (Shapiro, 2003); that is, optimisation requires a scenario tree (to reflect information structure) while sampled paths are sufficient to evaluate the objective of any given (fixed) policy.

Feasible policies are constructed by extending scenario tree policies to other outcome paths by means of a nearest neighbour extrapolation. The nearest outcome path within the scenario tree must be determined using the *nested distance*, equipped with a distance metric  $d_t(\omega, \hat{\omega})$ , to preserve adherence to the original information structure. The nested distance was introduced by Pflug and Pichler (2012) to incorporate causality in measures

of distance between stochastic processes. This is achieved by letting distances in earlier stages have complete priority over distances in later stages, based on the fact that future realisations of the stochastic process cannot be known in advance. For our purposes, we want to find the path within a scenario tree that is closest to any given outcome path with respect to the nested distance. We achieve this by recursively (one stage at a time) picking the branch within the scenario tree that is closest to the outcome path, where ties are resolved by the order of child nodes within the scenario tree.

The distance metric  $d_t(\omega, \hat{\omega})$  is constructed by converting the unit of each stochastic parameter to its approximate corresponding profit. We assume each fish has approximately a final sales value of 50 NOK/kg, at weight 4 kg/fish, and that there are 100 000 fish per cage whose average treatment cost is then 2.5 NOK/fish. Using these estimates, we translate distances between stochastic parameters to the unit NOK/fish and use the squared Euclidean norm as a distance metric. The distance between outcomes  $\omega$  and  $\hat{\omega}$  in stage  $t$  is then expressed as

$$d_t(\omega, \hat{\omega}) = \|2.5 \cdot (\psi_t(\omega) - \psi_t(\hat{\omega}))\|^2 + \|4 \cdot (\pi_t(\omega) - \pi_t(\hat{\omega}))\|^2 + \|50 \cdot (w_t(\omega) - w_t(\hat{\omega}))\|^2 + \|200 \cdot (\sigma_t(\omega) - \sigma_t(\hat{\omega}))\|^2, \quad (\text{IV.25})$$

where  $\|\cdot\|^2$  denotes the squared Euclidean norm.

### Extension policy

The aim of the extension rule is to provide feasible policies that also perform well. Keep in mind that there are alternative ways of doing this, and that the quality of the extension rule also affects the quality of the resulting evaluation. Let  $\hat{x}$  denote the solution on the scenario tree defined for a restricted set of outcomes  $\hat{\Omega}$ , while the extension policy is denoted  $\check{x}$ . Let also  $\omega \in \Omega$  denote any outcome path, and let  $\hat{\omega} \in \hat{\Omega}$  denote the path in the scenario tree that is closest to  $\omega$  in terms of the nested distance using distance metric  $d_t(\omega, \hat{\omega})$ . This means  $\hat{\omega}$  is contained in the scenario tree, while  $\omega$  may not. The order matters for the extension rule and is inferred chronologically for each stage  $t \in \mathcal{T}$ . Within a stage  $t$ , the extension rule is defined as follows:

1. Harvest count:

$$\check{h}_{it}(\omega) \leftarrow \max \left\{ \frac{\hat{h}_{it}(\hat{\omega})}{\sigma_{it}(\hat{\omega})\check{x}_{i,t-1}(\hat{\omega})} \sigma_{it}(\omega)\check{x}_{i,t-1}(\omega), \sigma_{it}(\omega)\check{x}_{i,t-1}(\omega) - \frac{B_i^{\text{loc}}}{w_{it}(\omega)} \right\}. \quad (\text{IV.26})$$

The harvest quantity is extended as the relative harvest count, compared to the in-going fish count. This ensures sites are emptied in the same stage for the extension policy as for the tree policy, and prevents harvesting more than the in-going fish count. Site-level biomass restrictions are fulfilled by forced harvest in the event that the limit is surpassed.

2. Out-going fish count:

$$\check{x}_{it}(\omega) \leftarrow \sigma_{it}(\omega)\check{x}_{i,t-1}(\omega) - \check{h}_{it}(\omega). \quad (\text{IV.27})$$

3. Binary variables:

$$\check{h}_{it}^{\diamond}(\omega) \leftarrow \left\lceil \frac{\check{h}_{it}(\omega)}{M_{it}(\omega)} \right\rceil, \quad \check{x}_{it}^{\diamond}(\omega) \leftarrow \left\lceil \frac{\check{x}_{it}(\omega)}{M_{it}(\omega)} \right\rceil, \quad (\text{IV.28})$$

where  $\lceil \cdot \rceil$  signifies rounding up to the nearest integer. Rounding is omitted for the LP relaxation, meaning  $\check{y}_{it}^{\diamond}(\omega)$  and  $\check{h}_{it}^{\diamond}(\omega)$  are set as low as possible.

4. Boat count:

$$\check{b}_{it}(\omega) \leftarrow \max \left\{ \check{h}_{it}^{\diamond}(\omega), \frac{w_{it}(\omega)\check{h}_{it}(\omega)}{C^{\text{boat}}} \right\}. \quad (\text{IV.29})$$

5. Violation of soft-constraints:

$$\check{z}_{it}^{\text{bflx}}(\omega) \leftarrow \left( \sum_{i \in \mathcal{A}_t} b_{it} - N^{\text{boat}} \right)^+, \quad (\text{IV.30})$$

$$\check{z}_{it}^{\text{blmt}}(\omega) \leftarrow \left( \sum_{i \in \mathcal{A}_t} b_{it} - 2N^{\text{boat}} \right)^+, \quad (\text{IV.31})$$

$$\check{z}_t^{\text{cmp}}(\omega) \leftarrow \left( \sum_{i \in \mathcal{O}_t} w_{it}(\omega)\check{x}_{it}(\omega) + \sum_{i \in \tilde{\mathcal{A}}_t} w_{it}\check{y}_{it} - B^{\text{cmp}} \right)^+. \quad (\text{IV.32})$$

Observe that the extension policy is completely described by the extension (IV.26), while all other quantities can be inferred from  $\check{h}$  as well as the outcome  $\omega \in \Omega$ . The interpretation is clear: The policy for harvest count is the most essential part of the decision problem, and all other quantities are there to reflect the consequences of this decision.

### 3.4 Scenario generation

Scenario tree generation for this problem is particularly challenging due to high dimensions, different kinds of stochastic variables, complicated dependence patterns and a long planning horizon. We now describe the procedure for generating scenario trees, which is a combination of ideas from existing approaches in the literature (Galuzzi et al., 2020; Kaut, 2014; Prochazka & Wallace, 2020). The procedure is later referred to as *Quantile Selection* (QS).

The stochastic process  $\xi_t$  can be represented as a transformation

$$\xi_t = g_t(\xi_{[t-1]}, \epsilon_t), \quad (\text{IV.33})$$

which infers future realisations based on the history  $\xi_{[t-1]}$  and a stochastic term  $\epsilon_t \sim U[0, 1]^{d_t}$  where  $d_t$  is the dimensionality of randomness in stage  $t$ . The stochastic term  $\epsilon_t$  represents the percentile of each random variable in stage  $t$ , and dependence patterns are expressed in terms of transformations of  $\epsilon_t$ . Scenario generation aims to represent the term  $\epsilon_t$  as a discrete set of outcomes within each stage. This approach is inspired by Kaut (2014). All random variables except for prices are assumed to be conditionally independent within each stage, but are still strongly affected by the history  $\xi_{[t-1]}$ . There are nine highly dependent price classes, which are instead represented by six conditionally independent latent factors.

Let  $N_t$  denote branching at stage  $t$ , meaning we have a total of  $\prod_{t \in \mathcal{T}} N_t$  outcome paths within the scenario tree, and let  $d_t$  denote the number of random variables in each stage. The dependence among random variables within each stage is generated using conditional latin hypercube sampling: For each of the  $d_t$  random variables, we first partition the range of (cumulative) probabilities  $[0, 1]$  into  $N_t$  equally large sub-ranges  $[a_s, b_s)$  for  $s = 1, \dots, N_t$ . Then, the order of these sub-ranges are shuffled randomly for each random variable  $i = 1, \dots, d_t$  and, based on the resulting order, we assign a sub-range from each random variable  $i$  to each of the  $N_t$  scenarios.

Given a sub-range  $[a, b)$ , we parameterise which quantile to use for each *kind* of random variable (prices, mortality and treatment) as  $p_{k_i} \in [0, 1]$  where  $k_i$  denotes the kind of random variable. Weight is inferred directly from temperature and from treatments (which reduce growth). Then we use the quantile function  $Q_i(\cdot)$  of random variable  $i$  (i.e., the inverse CDF) to determine the value in each given scenario as

$$Q_i((1 - p_{k_i}) \cdot a + p_{k_i} \cdot b). \quad (\text{IV.34})$$

The weights  $p$  specify whether to choose conservative or optimistic values within a percentile range. One exception is the binary random variable for treatment. Instead of deterministically choosing a quantile for treatment, we use weighted sampling to determine a (cumulative) probability  $v \in [a, b)$ , so that

$$Q_{\text{treat}}(v), \quad v \sim \text{Triangular}(a, b, c = (1 - p_{\text{treat}})a + p_{\text{treat}}b), \quad (\text{IV.35})$$

where  $\text{Triangular}(a, b, c)$  is a triangular distribution with support  $[a, b]$  whose mode lies in  $c$ . This ensures more variability in the binary variable than if the same quantile was chosen each time, and improves the performance of the scenario tree. The intuition for why we choose weights  $p$  that are different from 0.5 is that planning for more optimistic or pessimistic realisations of uncertainty can be constructive for the resulting policy to perform well.

The branching structure of the scenario tree must also be specified. In the interest of making parsimonious scenario trees, we cannot represent short-term developments throughout since that requires an excessively large scenario tree. Instead, we prioritise long-term developments accounted for by branching periodically throughout the horizon, every 6 weeks. This results in up to 8128 leaf nodes within 78 stages. For reference, the historical average time between lice treatments (which is the most important factor for long-term considerations) is 7 weeks. Larger scenario trees beyond this leads to tractability issues.

We use surrogate optimisation to find the parameters  $p$  that achieve the best possible out-of-sample objective value. The surrogate function then represents the computationally demanding steps of: scenario tree generation using parameters  $p$ , solving the formulation with respect to the scenario tree, extending the tree-based policy, and out-of-sample evaluation. Furthermore, the LP relaxation is used during this procedure to decrease computational time. To get more consistent results, the dependence pattern within each stage produced by conditional latin hypercube sampling uses the same seed for each evaluation. We use radial basis functions (Hastie et al., 2009) as a surrogate representation of the relation between  $p$  and the out-of-sample objective value.

This methodology is inspired by the problem-based approach to scenario generation by Prochazka and Wallace (2020) who fit a scenario set (for two-stage problems) to approximate out-of-sample evaluations well. Galuzzi et al. (2020) also explore whether Bayesian Optimisation (whose surrogate representation is a Gaussian Process) can be applied for a similar purpose in multistage problems using the out-of-sample evaluation approach by Keutchayan et al. (2017). A novelty in our approach is that we optimise the *parameterisation* of scenario generation instead of the scenarios themselves; hence, it scales better for large problems.

Out-of-sample evaluation paths are drawn randomly by latin hypercube sampling to reduce variance, and to ensure we more effectively cover the support of a very high-dimensional distribution. We do this by considering independence in the stochastic terms  $\epsilon_t$  for  $t \in \mathcal{T}$ , and collect  $(\epsilon_t)_{t \in \mathcal{T}}$  as a  $\sum_{t \in \mathcal{T}} d_t$ -dimensional random variable from which we sample  $S$  outcomes. Within each percentile range  $[a, b)$ , we draw a random value for

the sample paths (i.e., instead of using weights  $p$ ). Afterwards, we infer each sample path using the transformation (IV.33). Note that this amounts to stratified sampling across all stages and random variables, as opposed to the method used for Quantile Selection which uses *conditional* latin hypercube sampling within each stage. A single common sample is used per problem instance to compare the out-of-sample performance of different scenario trees. Given that the parameters  $p$  are optimised for out-of-sample evaluations, there is a risk of over-fitting scenario trees. To control for this, we have drawn two sets of sample paths; a *validation set* used to tune parameters, and a *testing set* used to check whether we get similar results. No statistically significant difference was found when comparing these evaluations.

### Alternative scenario generation methods

We compare the Quantile Selection (QS) method to two different scenario tree generation methods: (i) Conditional Latin Hypercube Sampling (CLHS), and (ii) Minimum Transportation Distance (MTD). The CLHS method uses the same procedure as QS but instead of choosing a specific quantile, we sample uniformly from each range  $[a, b]$  before evaluating the quantile function  $Q_i(\cdot)$ . The MTD method aims to minimise the (nested) transportation distance from the scenario tree to the prescribed stochastic process. Pflug and Pichler (2014) explain its underlying theory and methodology, motivated by the fact that the minimum (nested) transportation distance between a scenario tree and the prescribed stochastic process provides a bound on their discrepancy in optimal objective values. In principle, lowering the former discrepancy also decreases the latter to provide better solutions. We generate MTD scenario trees by a customised version of k-means clustering (Lloyd, 1982) applied to a tree structure. This heuristic is described in the Supplemental Material. All methods are specified by the same branching structure as described above.

## 4 Case studies

In this section, we apply the model to a set of case study problems that represent a variety of setups. The instances replicate insights from proprietary data and dialogues with domain experts. We use public data as far as possible where the companies, capacities and portfolio setups are real (but anonymised), while operational data is reconstructed. Further details are provided in the Supplemental Material. We investigate seven instances that represent small, medium and reasonably large companies. Their characteristics are summarised in Table IV.2. Harvest plans are all initialised on April 26th, 2018, which is on the brink of warmer months that have faster fish growth and higher activity of parasitic lice. In Section 4.3, we validate solutions of the proposed model with respect to the prescribed stochastic process and then we investigate managerial insights in Section 4.4. First, we explain evaluation metrics (Section 4.1) and the experimental setup (Section 4.2).

### 4.1 Evaluation metrics

This section serves as a reference for the evaluation metrics used in the other sections. Bounds on the optimal objective value gives indications on how well approximate solutions perform and to quantify the effects of uncertainty, while empirical metrics on scenario generation are valuable to assess the methodology.

**Table IV.2:** Characteristics of case study problems.

Instances:	I-7382	I-4292	I-5512	I-3082	I-8432	I-6721	I-2743
Planning horizon [weeks]	73	55	73	77	75	78	43
Number of sites	8	5	16	24	18	6	3
Number of cages	49	18	63	97	74	18	12
MTB [tonnes]	21240	8424	28968	44790	34284	8346	5616
Starting weights [kg]	0.4–3.7	1.4–5.7	0.0–6.2	0.0–6.3	0.0–5.7	0.4–7.0	2.2–6.0

### Objective evaluation of a candidate decision

Let  $R$  denote the discrete distribution that represents a scenario tree, and  $\check{P}$  the empirical distribution found by sampling outcome paths from the distribution  $P$ . For any feasible candidate policy  $x \in \mathcal{X}$ , we refer to

$$\mathbb{E}_R [F(x, \xi)], \quad \mathbb{E}_{\check{P}} [F(x, \xi)], \quad (\text{IV.36})$$

as *in-sample* and *out-of-sample evaluations*, respectively. The discrepancy between in-sample and out-of-sample evaluations is evaluated as

$$\text{Bias}(x) = \mathbb{E}_{\check{P}} [F(x, \xi)] - \mathbb{E}_R [F(x, \xi)], \quad (\text{IV.37})$$

which represents the approximation error of the scenario tree. We are primarily concerned with the performance of out-of-sample evaluations since this reflects the real performance of any given policy; still, the other quantities are useful to indicate whether we have reasonable approximations.

### Bounds and measures

To find the best estimate of the optimal objective value, we use out-of-sample evaluation on a collection of  $K$  candidate extension policies  $\check{\mathcal{X}} = \{\check{x}^1, \dots, \check{x}^K\}$  found by solving the problem on various scenario trees. Let

$$\text{Primal LB} = \max_{\check{x} \in \check{\mathcal{X}}} \mathbb{E}_{\check{P}} [F(\check{x}, \xi)], \quad (\text{IV.38})$$

denote the largest out-of-sample objective evaluation among these candidate policies. This is a lower-bound since the objective evaluation of any feasible policy is less than or equal to the optimal objective value. There may still exist better solutions but Primal LB represents our best estimate.

Various measures are used to quantify the value of planning for uncertainty. We refer to Birge and Louveaux (2011, Chapter 4) for a review on these measures which we adjust to a multistage setting using out-of-sample evaluations (see also Narum et al., 2023). The measures are expressed as:

$$\text{EVPI} = \mathbb{E}_{\check{P}} \left[ \min_{x \in \mathcal{X}} F(x, \xi) \right], \quad \text{EEV} = \mathbb{E}_{\check{P}} [F(\check{x}, \xi)]. \quad (\text{IV.39})$$

where  $\check{x} \in \text{Ext}(\arg \min_{x \in \mathcal{X}} F(x, \bar{\xi}))$ ,  $\bar{\xi}$  is the deterministic expected forecast, and  $\text{Ext}(\cdot)$  denotes extension of a tree policy. The *expected value of perfect information* (EVPI) represents the best obtainable performance given that we know the future perfectly (which is usually impossible); hence, EVPI is a loose upper-bound on the optimal objective value. The *expectation of the expected value solution* (EEV) represents the performance of a naïve

plan optimised for the deterministic expected future; hence, EEV is a loose lower-bound on the optimal objective value. Observe that EEV is not necessarily unique if there are multiple in-sample solutions that perform differently out-of-sample. We use whichever solution is provided from the numerical solver. From these measures, we may derive the quantities:

$$\text{VSS} = \frac{\text{Primal LB} - \text{EEV}}{\text{Primal LB}} \cdot 100\%, \quad \text{VPI} = \frac{\text{EVPI} - \text{Primal LB}}{\text{Primal LB}} \cdot 100\%, \quad (\text{IV.40})$$

known as the *value of the stochastic solution* (VSS) and the *value of perfect information* (VPI). VSS represents the value of planning for an uncertain future instead of a deterministic forecast (also interpreted as the value of actively hedging), while VPI represents the potential value of obtaining better information about the future if possible. Consider also that the range from EEV to EVPI represents the span of objective values within which the optimal objective value must reside. Relative to Primal LB, the size of this range can be quantified as

$$\frac{\text{EVPI} - \text{EEV}}{\text{Primal LB}} \cdot 100\% = \text{VSS} + \text{VPI}, \quad (\text{IV.41})$$

to represent the relative range of objective values in a stochastic environment. Its size is indicative of the effect of uncertainty on planning, which decomposes into VSS and VPI.

### Evaluation of scenario generation

Empirical evaluation of scenario generation is used to assess which method is likely to give the best results and to assess their stability. We refer to Kaut and Wallace (2007) for details, whose methodology we adjust to a multistage context. Let  $\mathcal{M}$  denote a scenario generation method and  $\mathcal{X}(\mathcal{M}) = \{\tilde{x}^1, \dots, \tilde{x}^L\}$  a collection of  $L = 10$  candidate policies extended from the optimal policy on scenario trees found using method  $\mathcal{M}$ . Scenario trees are different due to randomness in the generation method. Based on this, we assess the following quantities:

$$\mu(\mathcal{M}, \dot{P}) = \text{Average} \left\{ \mathbb{E}_{\dot{P}} [F(\tilde{x}, \xi)] \right\}, \quad (\text{IV.42a})$$

$$\sigma(\mathcal{M}, \dot{P}) = \text{Std} \left\{ \mathbb{E}_{\dot{P}} [F(\tilde{x}, \xi)] \right\}, \quad (\text{IV.42b})$$

$$\text{RMSB}(\mathcal{M}) = \sqrt{\text{Average} \left\{ \left( \mathbb{E}_{\dot{P}} [F(\tilde{x}, \xi)] - \mathbb{E}_R [F(\tilde{x}, \xi)] \right)^2 \right\}}. \quad (\text{IV.42c})$$

For a given method  $\mathcal{M}$ , these measures respectively signify quality in terms of the average performance ( $\mu(\mathcal{M}, \cdot)$ ), stability in terms of consistency in solutions ( $\sigma(\mathcal{M}, \cdot)$ ), and approximation error measured as the Root Mean Squared Bias (RMSB). The quality and stability measures can be found using in-sample evaluations ( $R$ ) as well as out-of-sample evaluations ( $\dot{P}$ ).

## 4.2 Experimental setup

The model is implemented in the Julia programming language (Bezanson et al., 2017) using the JuMP mathematical programming package (Dunning et al., 2017). Linear programming and mixed-integer programming formulations are solved using the v11.0 Gurobi solver (Gurobi Optimization, LLC, 2022). We use the Dual Simplex method and

set the MIP gap tolerance to 0.1% (1e-3). The optimality tolerance is set to (1e-9) due to low probability weights in the objective function. Surrogate optimisation uses the `Surrogates.jl` package. Computations are run on 20-core (40 threads) machines that each have 96 GB of RAM, while the largest instances are solved using 396 GB of RAM. LP relaxations typically solve within 20–60 seconds, while the larger MIP formulations typically solve within 12–24 hours.

We sample  $S = 3000$  outcome paths for the validation set. The standard errors of out-of-sample estimates of expected objective values are in the order of 0.1%. In terms of Monte Carlo Sampling, this implies that the size of a 95% confidence interval for the expected objective value corresponds to  $\pm 0.2\%$ . Given that we use latin hypercube sampling to generate paths, and since all comparison are made using a common sample, the margin of error in out-of-sample estimates is effectively lower than  $\pm 0.2\%$ .

### 4.3 Validation

The aim of validation is to ensure the optimisation model provides good solutions to the problem at hand, given that it approximates uncertainty through scenario trees. We investigate this by evaluating bounds on the optimal objective value as well as common measures on the effects of planning under uncertainty. Then, we investigate the reliability of the scenario generation method. Lastly, we investigate the effects of integers in the optimisation model through MIP gaps. The measures used below are described in Section 4.1.

#### Bounds and the value of information

Evaluating bounds on optimal objective values is valuable to assess how well an approximate model performs with respect to the prescribed stochastic process. Table IV.3 reports bounds on the optimal objective value as well as metrics on the effect of uncertainty. Figure IV.3 illustrates the accumulated profit of an extension policy as well as the distribution of total profit for some selected problem instances. Additionally, Figure IV.3 illustrates profit distributions for extending a deterministic policy (derived from EEV) as well as the perfect information policy (derived from EVPI). These also represent distributions whose expectations bound the optimal objective value.

**Table IV.3:** Bounds on the out-of-sample performance of tree-based policies. With respect to the optimal solutions, EEV is a lower-bound and EVPI is an upper-bound. VSS + VPI signifies the span of objective values due to the impact of uncertainty.

Instance	Primal LB	EEV	VSS (%)	EVPI	VPI (%)	VSS + VPI
I-7382	897.23	849.15	5.36%	1000.05	11.46%	16.82%
I-4292	328.89	311.88	5.17%	415.94	26.47%	31.64%
I-5512	924.87	881.17	4.73%	1089.25	17.77%	22.50%
I-3082	1607.74	1542.74	4.04%	1931.80	20.16%	24.20%
I-8432	1003.89	988.12	1.57%	1164.35	15.98%	17.56%
I-6721	230.35	222.50	3.41%	309.82	34.50%	37.91%
I-2743	173.87	161.41	7.17%	214.09	23.14%	30.30%

We observe from Table IV.3 that the span of objective values due to uncertainty are in the range of 17–38%, which is considerable. These numbers can be further decomposed into the effect of planning for an uncertain future instead of a deterministic one (VSS),

and the value of perfect information (VPI). These are in the ranges 1.6–7.2% for VSS and 11.5–34.5% for VPI.

The values of VSS imply that planning by explicit account of uncertainty can lead to increased profit in the order of 5%. For operational problems which consist of many small adjustments made consistently over time, we consider 5% to be a good improvement. Some modelling assumptions also imply VSS could be larger; for example, we assume less information about mortality rates than might be the case since these affect in-going fish instead of out-going fish (Section 3.2). Furthermore, we have found the problem to be sensitive to the ratio of income to expenses, since during years of lower sales prices VSS can be in the range of 15–20% across instances. The reason is that when margins are lower, the relative impact of acting on better information is larger.

The value of perfect information (VPI) is in the order of 20%, which implies there is large value in acting on better information than the model currently incorporates. Modelling health more precisely can be constructive in this respect, and incorporation of disease forecasting models may also help to provide more informed conditional estimates of mortality. Better scenario generation could also improve results further by collecting a larger share of the value implied by the VPI. One aspect is that larger scenario trees, for example enabled by decomposition methods, could provide better solutions.

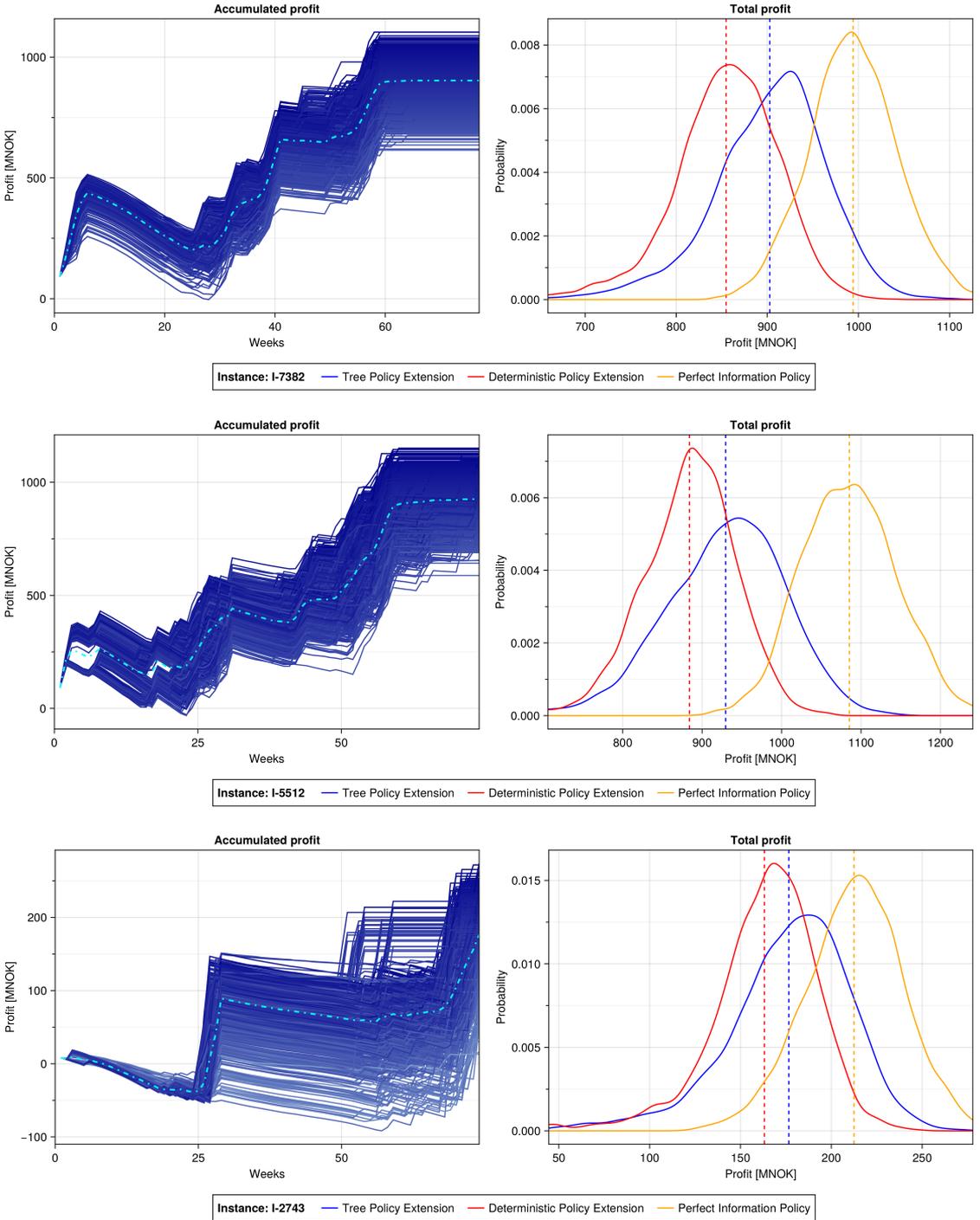
Consider also that the range of values for VSS and VPI are quite large across problem instances. This is a result of different portfolio setups as well as the initialisation state, not from approximation of uncertainty (which is shown to be consistent in Section 4.3). As mentioned, we also experience that initialising the problem at different dates leads to similar variability within the same company. We interpret this as a characteristic of the high complexity of the planning problem, where strategies and trade-offs differ based on the portfolio initialisation and the characteristics of each company. This attests to the utility of applying decision models to effectively and consistently navigate such high complexity.

Generally, we observe from Figure IV.3 that the variability in profit is large. We also find that the span of profits using the Perfect Information Policy is similar to that of the Tree Policy Extension, meaning there is wide profit variability that necessarily cannot be hedged effectively; namely, if having perfect information does not enable hedging risks, nothing will. Still, the differences in expected profit are relatively large when comparing the Deterministic Policy Extension to the Tree Policy Extension, meaning there still exist effective hedging strategies that increase the overall expected profit, even if the variability is large. These observations imply that aquaculture operations is subject to quite large levels of operational risk. We investigate this further in Section 4.4.

### Scenario generation

The aim of scenario generation is to provide solutions that perform as good as possible with respect to the prescribed distribution. We evaluate the scenario generation method proposed in Section 3.4 and compare to alternative methods in the literature. Table IV.4 compares different empirical metrics used to evaluate the different scenario generation methods, while Table IV.5 reports the final parameters chosen in the Quantile Selection method.

We observe from Table IV.4 that Quantile Selection (QS) consistently performs better than the alternative methods in terms of out-of-sample objective values. Furthermore, the stability of QS is much better than Conditional Latin Hypercube Sampling (CLHS). The stability of Minimum Transportation Distance (MTD) seems to imply it consistently provides similar results but at an overall lower quality than QS. Overall, bias is quite high



**Figure IV.3:** Accumulated profit over time (left), and distributions of the total profit (right). Vertical dashed lines (right) show the expected value of the distributions. The expected values of the Deterministic Policy Extension and the Perfect Information Policy are bounds on the optimal objective value.

**Table IV.4:** Quality, stability and bias of scenario generation methods, measured in-sample and out-of-sample. Quality and stability are reported as  $\mu(\mathcal{M}, \cdot) \pm 2\sigma(\mathcal{M}, \cdot)$  for in-sample ( $R$ ) and out-of-sample ( $\hat{P}$ ) evaluations. See also (IV.42).

Instance	Quantile Selection			CLHS			MTD		
	In-sample	Out-of-s.	RMSB	In-sample	Out-of-s.	RMSB	In-sample	Out-of-s.	RMSB
I-7382	945.76±10.90	894.86 ± 2.57	51.12	978.36±26.31	847.50±11.11	131.46	967.21 ± 0.78	878.04 ± 2.49	89.18
I-4292	375.42 ± 7.42	326.85 ± 2.57	48.75	397.48±25.35	290.52±24.51	107.82	388.37 ± 0.37	318.40 ± 1.56	69.97
I-5512	1046.96±14.17	921.04 ± 5.41	126.10	1072.74±30.11	858.65±14.24	214.52	1057.27 ± 1.10	893.01 ± 2.32	164.27
I-3082	1818.86±17.48	1605.06 ± 2.77	213.98	1870.45±46.86	1505.06±17.02	366.09	1840.65 ± 1.49	1576.46 ± 3.09	264.19
I-8432	1165.61±26.78	998.90 ± 4.90	167.20	1138.25±34.19	946.45±23.65	192.43	1115.89 ± 1.54	982.54 ± 1.88	133.36
I-6721	302.33 ± 8.28	229.28 ± 1.22	73.15	295.82±14.14	206.64 ± 7.18	89.43	288.96 ± 0.40	220.24 ± 1.65	68.72
I-2743	226.40 ± 8.52	173.62 ± 0.41	52.94	209.14±11.48	164.15 ± 3.59	45.38	203.02 ± 0.13	169.25 ± 0.43	33.77

for all methods and highest for CLHS. High bias implies the quality of scenario trees can likely be improved further, but consistent out-of-sample evaluations (as we find here) is of higher priority.

**Table IV.5:** Parameters in the best performing scenario trees for each instance using Quantile Selection.

Instance	Stages	Leaves	Nodes	$p_{\text{price}}$	$p_{\text{mortality}}$	$p_{\text{treatment}}$	LP Objective
I-7382	77	8192	81909	0.42	0.17	0.68	901.30
I-4292	75	8192	65525	0.65	0.87	0.61	334.28
I-5512	74	8192	57333	0.46	0.58	0.87	930.43
I-3082	76	8192	73717	0.51	0.64	0.83	1628.20
I-8432	75	8192	65525	0.57	0.77	0.79	1011.81
I-6721	74	8192	57333	0.56	0.70	0.83	237.87
I-2743	74	8192	57333	0.64	0.87	0.44	177.77

From Table IV.5, we find no evident pattern in the parametrisation of QS; still, we have found some important trade-offs in the specification of these. For example, we have found that jointly letting  $(p_{\text{price}}, p_{\text{mortality}})$  or  $(p_{\text{mortality}}, p_{\text{treat}})$  be close to 0.5 simultaneously gives worse performance. Furthermore, letting any of the parameters  $p$  be very large or very low also seems disadvantageous. Generally, sensitivity on the parameters  $p$  is low but there are some specifications that are clearly more effective than others.

## Integrality

Table IV.6 compares optimal objective value of the MIP formulation to its LP relaxation, as well as their out-of-sample extension policies. We see that the MIP gap is comparable in-sample and out-of-sample, which seems to imply that tuning scenario trees with respect to the LP relaxation can give reasonably effective results. While the MIP gap is relatively small, simply rounding the solution from the LP relaxation leads to larger gaps, and going from a solution to the LP relaxation to a MIP solution is not necessarily trivial. This is also reflected by the longer solve-time of the MIP formulation.

## 4.4 Managerial insights

We give some managerial insights related to risk exposure in aquaculture, valuation of fish stock, and harvest timing. Lastly, we benchmark a common rule-of-thumb to optimise biomass instead of profit. Keep in mind that profit in the objective function refers to

**Table IV.6:** MIP gaps for the best performing scenario tree in each instance. Tree policies are solved using the MIP formulation and its LP relaxation, then extended for out-of-sample evaluation. Only the MIP policy extension respects integrality.

Instance	In-sample			Out-of-sample		
	MIP	LP	MIP gap (%)	MIP	LP	MIP gap (%)
I-7382	947.05	955.10	0.85%	897.23	905.23	0.89%
I-4292	375.61	378.35	0.73%	328.89	330.65	0.53%
I-5512	1047.50	1052.39	0.47%	924.87	930.67	0.63%
I-3082	1826.56	1839.95	0.73%	1607.74	1626.16	1.15%
I-8432	1157.82	1162.01	0.36%	1003.89	1010.41	0.65%
I-6721	304.46	308.10	1.20%	230.35	235.28	2.14%
I-2743	227.78	229.94	0.95%	173.87	176.94	1.77%

*variable operational profit* only; that is, profit directly associated to harvest planning without any fixed or other expenses (smolt, maintenance, administrative, etc.).

### Volatility in profit

Large variability in profits as found in Figure IV.3 is indicative of large levels of operational risk. This is disadvantageous since volatile profit is a bad precondition for tactical planning of long-term considerations, like re-stocking. Specifically, operational profits are met by fixed costs, and when operational profit is volatile and fixed costs are high, this can cause irreversible consequences from particularly large losses or bankruptcy.

We do a further analysis of the distributions illustrated in Figure IV.3 by computing a variety of risk measures. These are listed in Table IV.7. Specifically, we examine the expected profit and the standard deviation of the profit distribution. We also derive volatility measure from the width of a 95% prediction interval, reported as a percentage of the expected objective value. We also find the 5% Value at Risk ( $\text{VaR}_{5\%}$ ); i.e., the profit at the 5th percentile, and the 5% Conditional Value at Risk ( $\text{CVaR}_{5\%}$ ); i.e., the conditional expected value below  $\text{VaR}_{5\%}$ .

**Table IV.7:** Variability and risk measures of total profit for different problem instances. Volatility refers to the width of a 95% prediction interval, relative to the expected profit. Value at Risk ( $\text{VaR}_{5\%}$ ) denotes the 5th percentile, while the Conditional Value at Risk ( $\text{CVaR}_{5\%}$ ) is the conditional expected profit below  $\text{VaR}_{5\%}$ .

Instance	Expectation	St. Dev.	Volatility	$\text{VaR}_{5\%}$	$\text{CVaR}_{5\%}$
I-7382	902.69	62.46	27.12%	791.03	748.30
I-4292	333.20	41.34	48.63%	258.53	235.50
I-5512	929.65	74.32	31.34%	803.90	761.78
I-3082	1624.27	116.87	28.21%	1428.59	1370.21
I-8432	1009.71	70.16	27.24%	890.23	847.22
I-6721	234.82	43.62	72.82%	156.01	131.40
I-2743	176.59	34.21	75.95%	115.13	87.46

We observe in Table IV.7 that the volatility is in the very considerable range of 27–76%.  $\text{CVaR}$  is also quite low, at almost half of the expected profit in some instances. Based on the results from Section 4.3, it is particularly interesting that this variability seems non-trivial to hedge in itself only by optimising the expected profit. A potential extension of the current model is to additionally (or instead) optimise the  $\text{CVaR}$  of the

total profit in an attempt to alleviate some of the observed down-side. Necessarily, this results in an expected profit that is as good or worse.

Asche et al. (2013) have previously discussed that aquaculture companies have acquired smaller companies as a strategic risk mitigation strategy. The results in Table IV.7 supports this conclusion since the volatility for large companies (of higher expected profit) is much lower than for the small companies. Overall, these numbers imply significant levels of risk across all companies.

### Valuation of fish stock

In a broader sense, harvest planning must consider the alternative cost of harvesting today by estimating the future value of fish stock with respect to time dynamics, uncertainty and operational flexibility, as discussed in Section 3.2. For in-sample analysis, using the LP relaxation, it is possible to extract duals directly from an optimal solution to obtain the shadow price of fish stock ( $\phi$ ). This can be compared to current sales profit ( $\theta$ ) and, when these coincide, the model implies that fish stock can be sold immediately at the same profit implied by any other future value state (minus costs in between). In other words, there is no reason to wait. By comparing these quantities, we get an indication of how long we should wait until harvesting. See also the discussion in Section 3.2. Figure IV.4 illustrates the value estimates  $\phi$  and  $\theta$  of fish stock up to the last exhaustive harvest.

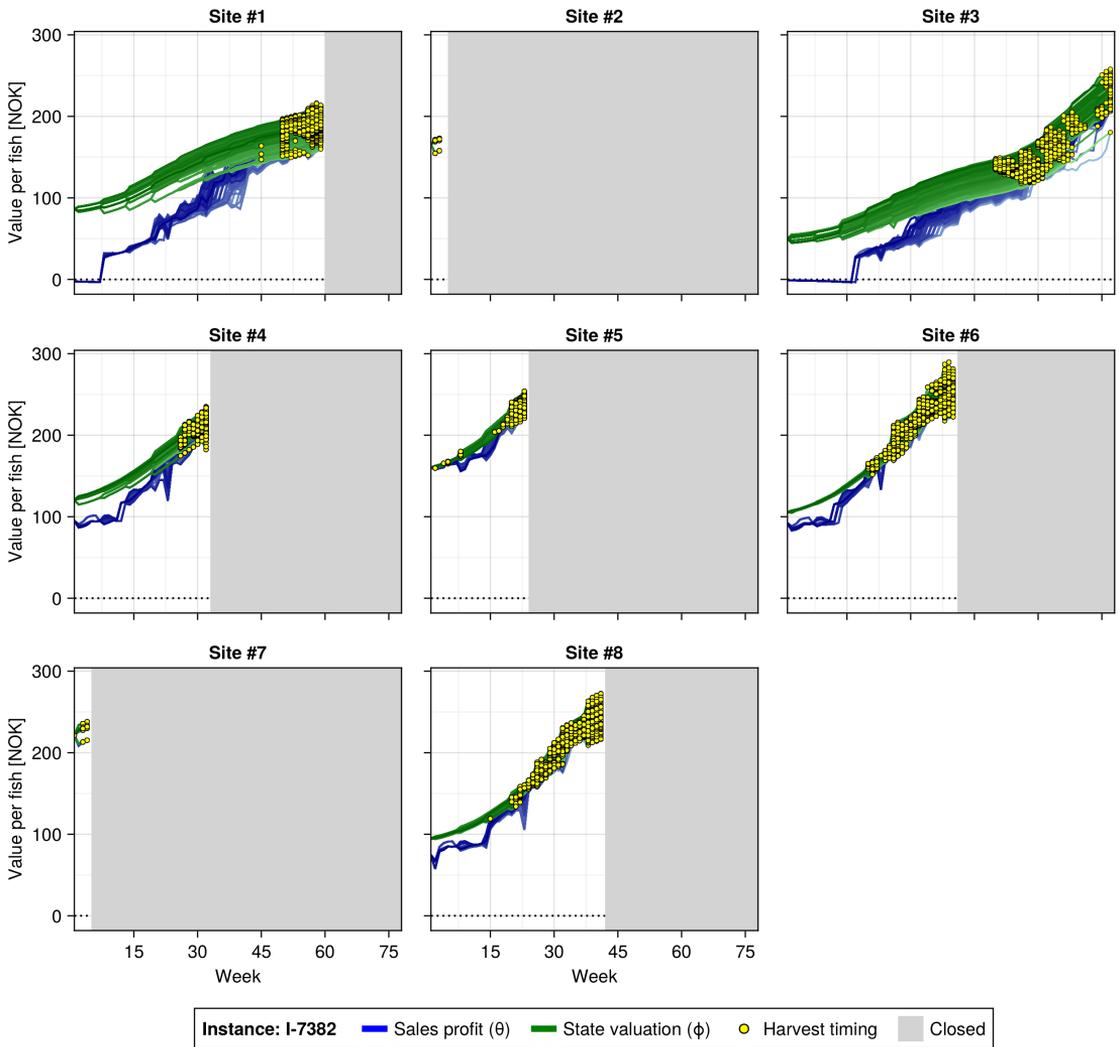
We observe from Figure IV.4 that the value estimate of fish stock ( $\phi$ ) exhibits relatively large variability across outcomes of uncertainty (especially for site #1 and #3) which correspondingly leads to a similar variability in the time of harvests. In conclusion, actively acting on available information implies harvest schedules would change to some meaningful degree.

We also observe that the gap between value states ( $\phi$ ) and sales value ( $\theta$ ) is larger early on, but that they are very close for some time up to the last exhaustive harvest. For sites where these are close for extended periods of time (site #6 and #8), we also see that the times of harvest span very large ranges in the order of 15–20 weeks. In other words, there is large ambiguity in *when to harvest*. Preferably, we would like to have clear cut and obvious conclusions about when to harvest, while here we observe the opposite. Within such sensitive regimes, consideration of uncertainty, portfolio effects and sequencing can be especially important in ways that are not necessarily obvious when planning manually.

### End-of-horizon valuation

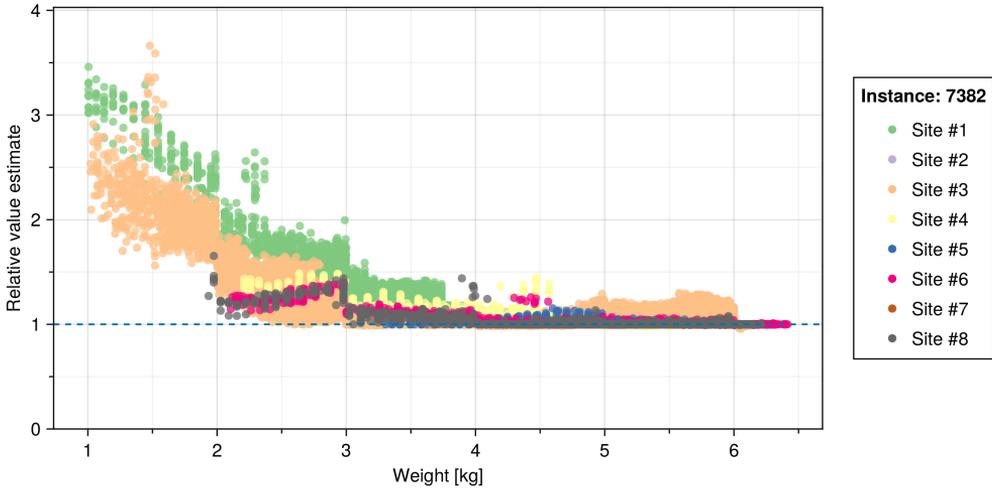
We now want to investigate whether an end-of-horizon value approximation could serve as an effective simplification for the proposed decision model. For this to be attainable, there must exist a simple expression for the expected future value of fish stock as a function of some known quantity, like fish weight. Already (in Section 4.4), we have seen that  $\phi$  quantifies the state value of fish stock. To filter out the implication of price deviations, we further scale this by the sales profit  $\theta$ . The relative value estimate  $\phi/\theta$  then serves as a proxy for what an end-of-horizon value approximation *should* account for. For example, the model accounts for operational flexibility, portfolio effects and heterogeneity in sites, which are difficult to replicate by simpler expressions.

Figure IV.5 illustrates the relative value estimate  $\phi/\theta$  as a function of weight  $w$  across all sites, stages and states of uncertainty. If only prices and weight matter for valuation of fish stock (which we have argued is *not* the case), Figure IV.5 should provide a simple relation between  $\phi/\theta$  and  $w$ . Instead, we observe quite high variability in relative value estimates as a function of weight, and the interpretation is that there is more to valuation



**Figure IV.4:** In-sample estimate of the state value of fish stock ( $\phi_{it}$ ) compared to its corresponding sales profit. When state values and sales profit meet, harvesting is initiated in the model (indicated by yellow points).

of fish stock than weight and sales price. In particular, the ordering of relative value estimates is important to determine the sequence of sites to harvest from, and such an ordering is not immediately clear from the figure. Based on these observations, we conclude that the more extensive decision model adds a level of detail that better reflects the operational situation.



**Figure IV.5:** In-sample estimate of the state value of fish stock ( $\phi$ ) relative to its corresponding sales profit ( $\theta$ ). When these meet ( $\phi/\theta = 1$ ), harvesting should be initiated.

### Benchmark on biomass optimisation

A common heuristic in the Aquaculture industry is to maximise total biomass, where a largest-fish-first policy is a common rule-of-thumb to achieve this. This means that whenever the farmer thinks it is wise to harvest, the largest fish are collected first. To replicate this situation, we formulate a model that optimises biomass in terms of average prices within each price class instead of varying prices, and refer to this as *biomass optimisation* as opposed to *profit optimisation*. We use average prices to reflect that sold biomass should also be traded off with costs. We then benchmark both strategies by out-of-sample evaluation that accounts for variable prices. The profit optimising policy should always perform better than biomass optimisation, simply because it optimises for the evaluation objective. Thus, we are mainly interested in the magnitude of this difference. Note that the extension rule can, in theory, still cause exceptions to this ordering of objective values. Table IV.8 reports the in-sample and out-of-sample performance of optimising either for biomass or profit.

We observe from Table IV.8 that profit optimisation improves profit in the range of 5–10%, and up to 43% in one instance. This shows there are considerable value to be collected by incorporating price forecasts into harvest planning. The very large improvement for one instance is due to a badly timed price trend which caused an especially low profit for biomass optimisation, where profit optimisation prioritised delayed harvest.

**Table IV.8:** Objective values when optimising for profit and for biomass. The last column compares the out-of-sample performance of profit optimisation relative to that of biomass optimisation. The in-sample objective of biomass optimisation assumes static prices, while all other evaluations assume variable prices.

Instance	Biomass optimisation		Profit optimisation		Improvement
	In-sample	Out-of-sample	In-sample	Out-of-sample	
I-7382	937.83	854.76	947.05	897.23	4.97%
I-4292	337.06	230.25	375.61	328.89	42.84%
I-5512	971.78	871.19	1047.50	924.87	6.16%
I-3082	1716.10	1528.81	1826.56	1607.74	5.16%
I-8432	973.64	937.22	1157.82	1003.89	7.11%
I-6721	295.93	210.27	304.46	230.35	9.55%

## 5 Discussion and conclusion

The overall goal of this work is explore whether decision models are able to handle the high complexity of harvest planning in aquaculture. We find that we are able to solve the problem effectively, using a combination of precise modelling assumptions and effective scenario tree generation to approximate uncertainty. It should be appreciated that the Aquaculture Harvest Planning problem exhibits high complexity, and that sensitivity on the initial state and setup of portfolios is high. This emphasises the value of quantitative decision support tools which farmers can apply in a variety of situations to consistently make effective decisions over time. For operational models like this one, consistently performing well is especially important since small contributions add up over time. With this in mind, we believe the proposed model formulation can provide great value to the industry.

We have formulated the decision model such that all uncertain factors enter as coefficients to state variables that represent fish count. An advantage of this formulation is that it allows for large-scale mixed-integer linear programming formulations with very precise account of growth, biology and prices, within a far-reaching planning horizon. Specifically, all non-linearities from these factors are accounted for in the coefficients themselves. Enabling large-scale formulations is especially important to reflect the large heterogeneity observed in the value of fish stock across sites. While some biological aspects have been simplified, the formulation of the decision model also allows input from highly complicated statistical models since we make no assumptions on time structure (non-Markovian), normality or linearity.

### Future research

Stocking decisions of when to deploy new fish is a very relevant tactical consideration that strongly affects operations, and we believe this is an interesting topic for future research. The presented model is an important contribution to enable guiding such tactical decisions from the operational level, while large computational requirements may be an obstacle. However, there is potential to apply decomposition procedures to improve solution times, both by site-level planning (bound together by company-wide constraints) and by scenarios (bound together by non-anticipativity).

## References

- Abolofia, J., Asche, F., & Wilen, J. E. (2017). The Cost of Lice: Quantifying the Impacts of Parasitic Sea Lice on Farmed Salmon. *Marine Resource Economics*, 32(3), 329–349. <https://doi.org/10.1086/691981>
- Apap, R. M., & Grossmann, I. E. (2017). Models and computational strategies for multistage stochastic programming under endogenous and exogenous uncertainties. *Computers & Chemical Engineering*, 103, 233–274. <https://doi.org/10.1016/j.compchemeng.2016.11.011>
- Asche, F., Roll, K. H., Sandvold, H. N., Sørvig, A., & Zhang, D. (2013). Salmon Aquaculture: Larger Companies and Increased Production. *Aquaculture Economics & Management*, 17(3), 322–339. <https://doi.org/10.1080/13657305.2013.812156>
- Bang Jensen, B., Qviller, L., & Toft, N. (2020). Spatio-temporal variations in mortality during the seawater production phase of Atlantic salmon (*Salmo salar*) in Norway. *Journal of Fish Diseases*, 43(4), 445–457. <https://doi.org/10.1111/jfd.13142>
- Bertsimas, D., & Tsitsiklis, J. (1997). *Introduction to Linear Programming*. Athena Scientific.
- Bezanson, J., Edelman, A., Karpinski, S., & Shah, V. B. (2017). Julia: A Fresh Approach to Numerical Computing. *SIAM Review*, 59(1), 65–98. <https://doi.org/10.1137/141000671>
- Birge, J. R., & Louveaux, F. (2011). *Introduction to stochastic programming* (2nd ed.). Springer New York, NY. <https://doi.org/10.1007/978-1-4614-0237-4>
- Borodin, V., Bourtembourg, J., Hnaien, F., & Labadie, N. (2016). Handling uncertainty in agricultural supply chain management: A state of the art. *European Journal of Operational Research*, 254(2), 348–359. <https://doi.org/10.1016/j.ejor.2016.03.057>
- Casey, M. S., & Sen, S. (2005). The Scenario Generation Algorithm for Multistage Stochastic Linear Programming. *Mathematics of Operations Research*, 30(3), 615–631. <https://doi.org/10.1287/moor.1050.0146>
- Denstad, A. G., Lillevand, M., & Ulsund, E. A. (2015). *Production planning and sales allocation in the salmon farming industry* [Master’s thesis, Norwegian University of Science and Technology].
- Dowson, O., Philpott, A., Mason, A., & Downward, A. (2019). A multi-stage stochastic optimization model of a pastoral dairy farm. *European Journal of Operational Research*, 274(3), 1077–1089. <https://doi.org/10.1016/j.ejor.2018.10.033>
- Dunning, I., Huchette, J., & Lubin, M. (2017). JuMP: A Modeling Language for Mathematical Optimization. *SIAM Review*, 59(2), 295–320. <https://doi.org/10.1137/15M1020575>
- Engehagen, S., Hornslien, M., Lavrutich, M., & Tønnessen, S. (2021). Optimal harvesting of farmed salmon during harmful algal blooms. *Marine Policy*, 129, 104528. <https://doi.org/10.1016/j.marpol.2021.104528>
- Forsberg, O. I., & Guttormsen, A. G. (2006). The Value of Information in Salmon Farming. Harvesting the Right Fish at the Right Time. *Aquaculture Economics & Management*, 10(3), 183–200. <https://doi.org/10.1080/13657300600985215>
- Føsund, J. M., & Strandkleiv, E. H. (2021). *Using Branch and Price to Optimize Land-Based Salmon Production* [Master’s thesis, Norwegian University of Science and Technology].
- Galuzzi, B. G., Messina, E., Candelieri, A., & Archetti, F. (2020). Optimal Scenario-Tree Selection for Multistage Stochastic Programming. In G. Nicosia, V. Ojha, E. La Malfa, G. Jansen, V. Sciacca, P. Pardalos, G. Giuffrida & R. Umeton (Eds.), *Machine Learning, Optimization, and Data Science* (pp. 335–346). Springer International Publishing. [https://doi.org/10.1007/978-3-030-64583-0\\_31](https://doi.org/10.1007/978-3-030-64583-0_31)
- Gurobi Optimization, LLC. (2022). Gurobi optimizer reference manual. <https://www.gurobi.com>
- Hæreid, M. B. (2011). *Allocating Sales in the Farming of Atlantic Salmon: Maximizing Profits Under Uncertainty* [Master’s thesis, Norwegian University of Science and Technology].
- Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning*. Springer New York. <https://doi.org/10.1007/978-0-387-84858-7>

- Hellemo, L., Barton, P. I., & Tomasgard, A. (2018). Decision-dependent probabilities in stochastic programs with recourse. *Computational Management Science*, 15(3-4), 369–395. <https://doi.org/10.1007/s10287-018-0330-0>
- Hornsletten, H. (2017). *Optimization Model Aimed for the Aquaculture Industry for Fleet Composition and Routing of Wellboats* [Master's thesis, Norwegian University of Science and Technology].
- Iversen, A., Hermansen, Ø., Nystøyl, R., & Hess, E. J. (2017). *Kostnadsutvikling i lakseoppdrett – med fokus på før- og lusekostnader*. Nofima AS. Retrieved December 5, 2020, from <https://nofima.brage.unit.no/nofima-xmlui/handle/11250/2481501>  
Accepted: 2018-02-01T14:47:27Z.
- Jonsbråten, T. W., Wets, R. J.-B., & Woodruff, D. L. (1998). A class of stochastic programs with decision dependent random elements. *Annals of Operations Research*, 82(0), 83–106. <https://doi.org/10.1023/A:1018943626786>
- Kaut, M. (2014). A copula-based heuristic for scenario generation. *Computational Management Science*, 11(4), 503–516. <https://doi.org/10.1007/s10287-013-0184-4>
- Kaut, M., & Wallace, S. W. (2007). Evaluation of scenario-generation methods for stochastic programming. *Pacific Journal of Optimization*, 3(2), 257–271.
- Keutchayan, J., Gendreau, M., & Saucier, A. (2017). Quality evaluation of scenario-tree generation methods for solving stochastic programming problems. *Computational Management Science*, 14(3), 333–365. <https://doi.org/10.1007/s10287-017-0279-4>
- Langan, T. B., & Toftøy, T. (2011). *Produksjonsoptimering innenfor lakseoppdretten - planlegging under usikkerhet* [Master's thesis, Norwegian University of Science and Technology].
- Lloyd, S. (1982). Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2), 129–137. <https://doi.org/10.1109/TIT.1982.1056489>
- Lovdata. (2012). Forskrift om bekjempelse av lakselus i akvakulturanlegg - Lovdata. Retrieved September 8, 2021, from <https://lovdata.no/dokument/SF/forskrift/2012-12-05-1140>
- Narum, B. S., & Berentsen, G. D. (2024). Joint Forecasting of Salmon Lice and Treatment Interventions in Aquaculture Operations  
Under review.
- Narum, B. S., Maggioni, F., & Wallace, S. W. (2023). On the safe side of stochastic programming: Bounds and approximations. *International Transactions in Operational Research*, 30(6), 3201–3237. <https://doi.org/10.1111/itor.13317>
- Oglend, A., & Tveteras, R. (2009). Spatial Diversification in Norwegian Aquaculture. *Aquaculture Economics & Management*, 13(2), 94–111. <https://doi.org/10.1080/13657300902881674>
- Pettersen, J. M., Osmundsen, T., Aunsmo, A., Mardones, F. O., & Rich, K. M. (2015). Controlling emerging infectious diseases in salmon aquaculture. *Revue Scientifique Et Technique (International Office of Epizootics)*, 34(3), 923–938. <https://doi.org/10.20506/rst.34.3.2406>
- Pflug, G. C., & Pichler, A. (2012). A Distance For Multistage Stochastic Optimization Models. *SIAM Journal on Optimization*, 22(1), 1–23. <https://doi.org/10.1137/110825054>
- Pflug, G. C., & Pichler, A. (2014). *Multistage Stochastic Optimization*. Springer International Publishing. <https://doi.org/10.1007/978-3-319-08843-3>
- Powell, W. B. (2019). A unified framework for stochastic optimization. *European Journal of Operational Research*, 275(3), 795–821. <https://doi.org/10.1016/j.ejor.2018.07.014>
- Prochazka, V., & Wallace, S. W. (2020). Scenario tree construction driven by heuristic solutions of the optimization problem. *Computational Management Science*, 17(2), 277–307. <https://doi.org/10.1007/s10287-020-00369-2>
- Rockafellar, R. (1999). Duality and optimality in multistage stochastic programming. *Annals of Operations Research*, 85(0), 1–19. <https://doi.org/10.1023/A:1018909508556>
- Schütz, P., & Westgaard, S. (2018). Optimal hedging strategies for salmon producers. *Journal of Commodity Markets*, 12, 60–70. <https://doi.org/10.1016/j.jcomm.2017.12.009>
- Shapiro, A. (2003). Inference of statistical bounds for multistage stochastic programming problems. *Mathematical Methods of Operations Research*, 58(1), 57–68. <https://doi.org/10.1007/s001860300280>

- Shapiro, A., & Nemirovski, A. (2005). On Complexity of Stochastic Programming Problems. In V. Jeyakumar & A. Rubinov (Eds.), *Continuous Optimization: Current Trends and Modern Applications* (pp. 111–146). Springer US. [https://doi.org/10.1007/0-387-26771-9\\_4](https://doi.org/10.1007/0-387-26771-9_4)
- Walde, C. S., Jensen, B. B., Pettersen, J. M., & Stormoen, M. (2021). Estimating cage-level mortality distributions following different delousing treatments of Atlantic salmon (*salmo salar*) in Norway. *Journal of Fish Diseases*, *44*(7), 899–912. <https://doi.org/10.1111/jfd.13348>
- Walde, C. S., Stormoen, M., Pettersen, J. M., Persson, D., Røsæg, M. V., & Bang Jensen, B. (2022). How delousing affects the short-term growth of Atlantic salmon (*Salmo salar*). *Aquaculture*, *561*, 738720. <https://doi.org/10.1016/j.aquaculture.2022.738720>

# Supplemental Materials



# Supplement A

## Problem-based Scenario Generation by Decomposing Output Distributions

This Supplemental Material gives a detailed description of all case study problems in the paper. First, we give a high-level problem description, then details on instance generation and distributions, and lastly, mathematical formulations are provided. All formulations are given by a table over variables, parameters and sets, and then the first- and second-stage programs are given. The stochastic vector  $\xi$  is the concatenation of all problem-specific stochastic parameters, and  $Q(x, \xi)$  is the recourse function. The underlying distributions are assumed known by analytical expressions and all observed distributions are generated by sampling 5000 outcomes from these. While the general procedure to randomly generate instances is described, computational experiments in the paper use one instance per problem, chosen for its lack of stability when solved with sampled scenario sets of size three.

### 1 Telecommunication Network Planning

The Telecommunications Network Planning (TNP) problem consists of determining where in a telecommunications network to allocate extra capacity, under uncertainty of demand between point-to-point pairs of nodes. The problem is adapted from the paper by Sen et al. (1994). In reality, routing of requests is done in real-time, but they approximated this by predetermining all possible routes of maximally three links between each point-to-point pair since routes are commonly not longer than two links. Sen et al. (1994) validated this to be a good assumption for their problem instance through simulation.

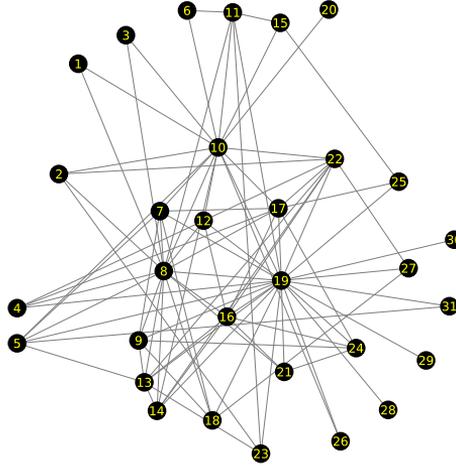
#### Instance generation and distribution

The network topology from the original paper was used. The demand point-to-point pairs were drawn randomly among all feasible pairs connected by maximally three or fewer links, excluding the three centre nodes that are considered to be hubs without their own demand. The existing capacity is set to 10 on all links, while the total expandable capacity is set to 2.5 times the total average demand of the demand distribution. Key

numbers on the problem instance are given in Table A.1 and the network topology is illustrated in Figure A.1.

**Table A.1:** Telecommunication Network Planning key figures.

No. of nodes	31
No. of demand pairs	82
Links	89
No. of routes (median over pairs)	8.5



**Figure A.1:** Network topology for TNP.

From the description of the distribution by Sen et al. (1994) we chose a gamma distribution with a shape parameter of 5 and expectation of 25 for all demand pairs, and assume independence.

## Formulation

Table A.2 gives an overview of variables, sets and parameters. The first-stage program has formulation

$$\min_x \mathbb{E}[Q(x, d)] \quad (\text{A.1a})$$

s.t.

$$\sum_j x_j \leq b, \quad (\text{A.1b})$$

$$x \geq 0 \quad (\text{A.1c})$$

**Table A.2:** Overview of problem variables, sets and parameters for TNP.

Sets	
$I$	set of point-to-point demand pairs
$J$	set of links between nodes
$R(i)$	set routes between point-to-point pair $i$
Deterministic parameters	
$e_j$	existing capacity for link $j$
$A_{irj}$	indicator tensor whose elements are 1 if link $j$ belongs to route $r$ for point-to-point demand pair $i$ , and 0 otherwise
$b$	total available capacity for expansion
Stochastic parameters	
$d_i$	Demand for point-to-point pair $i$
First stage decisions	
$x_j$	Capacity expansion allocated at link $j$
Second stage decisions	
$s_i$	Unserved demand for point-to-point pair $i$
$f_{ir}$	Flow for demand pair $i$ through route $r$

where constraint (A.1b) is the maximum extra capacity to be allocated. The second-stage program has the formulation

$$Q(x, \xi) = \min_{f, s} \sum_i s_i \quad (\text{A.2a})$$

s.t.

$$\sum_i \sum_{r \in R(i)} A_{irj} f_{ir} \leq x_j + e_j \quad \forall j \in J, \quad (\text{A.2b})$$

$$\sum_{r \in R(i)} f_{ir} + s_i = d_i \quad \forall i \in I, \quad (\text{A.2c})$$

$$f, s \geq 0 \quad (\text{A.2d})$$

where constraint (A.2b) is the total capacity constraint on link  $j$  given the total flow going through that link, and constraint (A.2c) is the demand balance where the slack variable  $s$  is used in the objective as a penalty for unmet demand.

## 2 Multidimensional News-vendor with Substitution

The Multidimensional Newsvendor with Substitution (MNV) is a production planning problem where production must be determined beforehand, while demand is observed later. Once demand is observed, the substitution sales must be determined. The model is inspired from the work by Vaagen et al. (2011) (also discussed by King & Wallace, 2012, Chapter 6). The distinction from the traditional news-vendor problem is the number of products, and that they may substitute each other at a certain asymmetric rate  $\alpha_{ij}$ . The application by Vaagen et al. (2011) was fashion, where demand is known to be multi-modal with strong dependence, and the substitution patterns can be determined qualitatively.

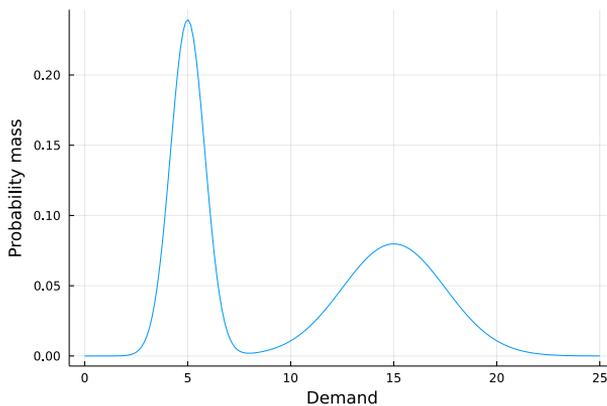
Typically, total demand is well known, so the composition of products must be determined under uncertainty. The advantage of modelling this problem as a stochastic program lies in the ability to deal with a complicated distribution, thus, we use it to demonstrate our scenario generation approach.

The formulation may be posed using (i) manufacturer-directed substitutions, where the manufacturer decides which products to sell, or (ii) customer-directed substitutions, where the customer decides what to buy. The difference lies in whether the stock of a product must be sold out before substituting (consumer-directed), or if substitution sales can be determined based on the current stock and substitution patterns (manufacturer-directed). Consumer-directed substitution requires binary variables in the second stage and is thus more computationally intensive to solve. The two formulations are equivalent up to the relaxation of the binary  $\hat{z}$ , however, computational experiments are performed on the relaxation.

## Instance generation and distribution

The instance is determined by the number of products, possible trend states, and substitution patterns. The number of trend states determines the number of modes in the multivariate distribution and represents how different styles, colours, or other properties of the products become popular in a given year, where dependence is strong. Secondly, substitution patterns are created only between products that have high demand simultaneously, assuming substitutability is associated with trend patterns. Vaagen et al. (2011) discusses the details.

The distribution is a multivariate mixture distribution with binary stochastic variables to determine the trend and two Normal distributions for the specific demand that are either high or low conditional on the trend. The dependence structure is explained solely by the binary stochastic variable for trend, while the Normals are independent. We assume the probability for each product to hit the trend is 0.5. The distribution is generated by first simulating the trend, and then drawing randomly from the high/low demand distributions.



**Figure A.2:** Marginal demand distribution.

The marginal demand distribution is bimodal due to the high-low split, modelled by two normal distributions  $N(\mu_{\text{low}}, \sigma_{\text{low}}^2)$ ,  $N(\mu_{\text{high}}, \sigma_{\text{high}}^2)$ . The variance of each is constructed such that zero is at the 6th standard deviation away from the mean, and any (very

unlikely) negative values are neglected. The low distribution has 50% of the expectation of the overall distribution, and the high has 150% of the overall expectation. Figure A.2 illustrates the marginal demand distribution.

**Table A.3:** Overview of problem variables, sets and parameters for MNV.

Sets	
$I$	Set of products
Deterministic parameters	
$v_i$	Sales price of item $i$
$g_i$	Salvage value of item $i$
$c_i$	Production cost
$\alpha_{ij}$	Substitution rate; average probability that item $j$ can be replaced by item $i$
$C$	Production capacity
$P$	Number of products that can be produced
Stochastic parameters	
$d_i$	Demand for item $i$
First stage decisions	
$x_i$	Production of item $i$
$\hat{x}_i$	Binary whether to produce item $i$
Second stage decisions	
$y_i$	Sales of item $i$
$z_{ij}$	Substitution sale of item $i$ , satisfying demand of item $j$
$\bar{z}_i$	Total substitution sale of item $i$ , satisfying demand from all items $j$
$\hat{z}_j$	Binary whether to start substitution sales satisfying demand for item $j$
$w_i$	Salvage quantity of item $i$

## Formulation

Table A.3 gives an overview of variables, sets and parameters. The first-stage program has formulation

$$\max_{x, \hat{x}} \mathbb{E}[Q(x, \xi)] - \sum_i c_i x_i \quad (\text{A.3a})$$

s.t.

$$\sum_i x_i \leq C, \quad (\text{A.3b})$$

$$\sum_i \hat{x}_i \leq P, \quad (\text{A.3c})$$

$$x_i \leq M \hat{x}_i, \quad (\text{A.3d})$$

$$x \geq 0, \hat{x} \in \{0, 1\} \quad (\text{A.3e})$$

where  $\hat{x}_i$  are binaries to determine the total number of product variations that can be produced. Constraint (A.3b) limits total production, constraint (A.3c) limits the number of different products that can be produced, while (A.3d) links them to be consistent with

each other. The second-stage program has formulation

$$Q(x, \xi) = \max_{y, z, \bar{z}, w} \sum_i v_i y_i + v_i \bar{z}_i + g_i w_i \quad (\text{A.4a})$$

s. t.

$$y_i + \sum_{j:j \neq i} z_{ji} \leq d_i \quad \forall i, \quad (\text{A.4b})$$

$$z_{ij} \leq \alpha_{ij}(d_j - y_j) \quad \forall i, j, i \neq j, \quad (\text{A.4c})$$

$$\bar{z}_i = \sum_{j:j \neq i} z_{ij} \quad \forall i, \quad (\text{A.4d})$$

$$M(\hat{z}_j - 1) \leq y_j - x_j \quad \forall j, \quad (\text{A.4e})$$

$$z_{ij} \leq M \hat{z}_j \quad \forall i, j, \quad (\text{A.4f})$$

$$w_i = x_i - (y_i + \bar{z}_i) \quad \forall i, \quad (\text{A.4g})$$

$$y, z, \bar{z}, w \geq 0, \hat{z} \in \{0, 1\} \quad (\text{A.4h})$$

where constraint (A.4b) signifies that sales (with substitutions) is less than demand, constraint (A.4c) enforces the substitution rate, constraints (A.4e) and (A.4f) enforces the logic of whether direct sales must exhaust the stock before substitutions (consumer-directed substitution), and constraint (A.4g) is the salvage balance. All big-Ms  $M$  are set equal to the total production capacity  $C$ .

### 3 Airlift Operations Scheduling

The Airlift Operations Scheduling (AOS) problem is a two-stage stochastic program. The aim is to plan the allocation of different types of aircraft to different routes in the first stage and then adjust the plan in the second stage with a penalty for doing so. There is also a penalty for not using an already scheduled aircraft. The problem is due to Midler and Wollmer (1969) and was found by its use by Ariyawansa and Felt (2002, 2004).

The scope of decisions in this problem may be seen as strategic in the first stage and tactical in the second stage, with some approximating assumptions. The formulation addresses the number of planes to be allocated to a route, and it is assumed that the scheduling of flights is done outside the model once the decisions are set. With insufficient capacity at a route, we may reschedule an aircraft to do parts (or all) of the flights on that route, measured in hours. Rescheduling carries a premium, where we might imagine the aircraft has to be moved between airports to go through with the flight and then return to its remaining pre-scheduled flights, thus adding to the hours spent and the cost. The number of available hours is what limits the capacity. The problem may be formulated as a MILP by having the number of aircrafts be integer (Ariyawansa & Felt, 2004), however, computational experiments are performed on the relaxation since this is how it was originally formulated.

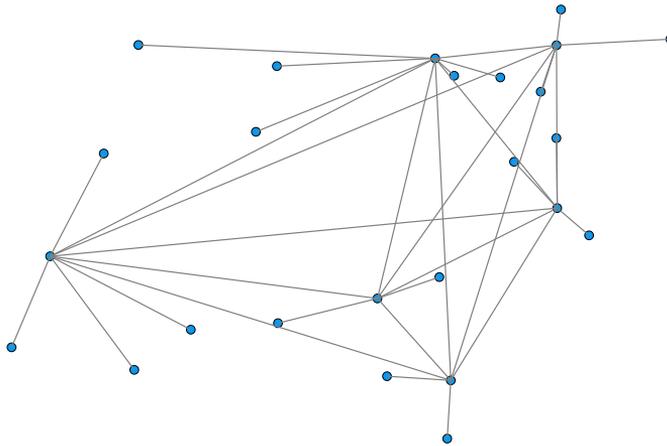
#### Instance generation and distribution

Problem instances are derived from a randomly generated network of airports with routes between them and some defined underlying parameters for costs given in Figure A.3. Instances are created by first drawing a set of random nodes in the range  $[0, 1]^2$ , and then clustering is used to make a given number of hubs. All hubs are connected to each

other by routes, but other nodes only have routes to hubs. Parameters are scaled to be comparable to what is given by Ariyawansa and Felt (2004), while we create larger instances. Flight time is assumed to be proportional to the distance of routes and the costs have a fixed cost plus a variable cost proportional to hours spent on a flight. The rescheduling premium amounts to adding the average distance between node pairs of two routes, which replicates the deviation from the already set schedule. Each kind of aircraft has an efficiency factor that scales the hours spent and capacity (more efficient means fewer hours and more capacity). The specific values are given in Table A.4.

**Table A.4:** AOS problem instance parameters

Parameter	Value
Fixed cost	2000
Variable cost	100
Carrying capacity	50
Penalty unmet demand	500
Penalty unused capacity	50
Flight time per distance	10
Efficiency factor	[0.7, 1.3]



**Figure A.3:** AOS network illustration.

Demand on routes is assumed to be gamma distributed and with spacial dependence between routes determined by a Matérn covariance matrix that defines a Gaussian copula for the joint dependence. Distance between routes is determined by the lowest total distance between pairs of their end points.

**Table A.5:** Overview of problem variables, sets and parameters for AOS.

Sets	
$A$	set of aircraft types (indexed by $i$ )
$R$	set of routes (indexed by $j, k$ )
Deterministic parameters	
$a_{ij}$	flying hours for aircraft type $i$ to do route $j$
$b_{ij}$	carrying capacity of aircraft $i$ doing route $j$
$c_{ij}$	cost for scheduling aircraft $i$ to do route $j$
$F_i$	maximum number of flying hours for aircrafts of type $i$
$a_{ijk}$	flying hours for aircraft $i$ to do route $k$ given it was originally scheduled to do route $j$
$c_{ijk}$	cost for aircraft $i$ to do route $k$ given it was originally scheduled to do route $j$
$c_j^+$	penalty for missed demand for route $j$
$c_j^-$	penalty for unused capacity for route $j$
Stochastic parameters	
$d_j$	Demand for route $j$
First stage decisions	
$x_{ij}$	number of aircraft of type $i$ to assign to route $j$
Second stage decisions	
$y_{ijk}$	number of aircraft of type $i$ to redirect from route $j$ to $k$
$u_{ij}^-$	number of aircraft of type $i$ redirected from $j$
$z_j^+$	excess demand for route $j$
$z_j^-$	unused capacity for route $j$

## Formulation

Table A.5 gives an overview of variables, sets and parameters. The first-stage program has the formulation

$$\min_x \sum_{i,j} c_{ij} x_{ij} + \mathbb{E}[Q(x, d)] \quad (\text{A.5a})$$

s.t.

$$\sum_j a_{ij} x_{ij} \leq F_i \quad \forall i \in A, \quad (\text{A.5b})$$

$$x_{ij} \geq 0 \quad \forall i \in A, \forall j \in R \quad (\text{A.5c})$$

where constraint (A.5b) limits the total hours available for a given aircraft type. The second-stage program has the formulation

$$Q(x, \xi) = \min_{y, z^+, z^-} \sum_{i,j,k \neq j} \left( c_{ijk} - c_{ij} \frac{a_{ijk}}{a_{ij}} \right) y_{ijk} + \sum_j c_j^+ z_j^+ + \sum_j c_j^- z_j^- \quad (\text{A.6a})$$

s.t.

$$\sum_{k \neq j} a_{ijk} y_{ijk} \leq a_{ij} x_{ij} \quad \forall i \in A, \forall j \in R, \quad (\text{A.6b})$$

$$\sum_i b_{ij} \left( x_{ij} - \sum_{k \neq j} \frac{a_{ijk}}{a_{ij}} y_{ijk} + \sum_{k \neq j} y_{ikj} \right) + z_j^+ - z_j^- = d_j \quad \forall j \in R, \quad (\text{A.6c})$$

$$y_{ijk} \geq 0 \quad \forall i \in A, \forall j, k \in R, \quad (\text{A.6d})$$

$$z_j^+, z_j^- \geq 0 \quad (\text{A.6e})$$

where constraint (A.6b) enforces that the originally planned available time for an aircraft type cannot be exceeded and constraint (A.6c) is demand balance. Ariyawansa and Felt (2002) explains additional details.

Rescheduling of aircraft  $i$  from route  $j$  to  $k$ , as encoded by variable  $y_{ijk}$ , reduced the capacity at route  $j$  by  $a_{ijk}$  hours that would otherwise take  $a_{ij}$  hours to complete. Considering that we neglect the exact scheduling, this leads to a loss in carrying capacity of  $b_{ij} \frac{a_{ijk}}{a_{ij}} y_{ijk}$  on route  $j$ . The added capacity at route  $k$  is as if it was originally scheduled at that route.

## 4 Storage Layout and Routing

This problem is constructed for this paper to address difficult scenario generation for highly computationally complex problems. Positions of products in a storage facility must be determined in the first stage, while in the second stage *batches* (subsets) of products will be collected. The objective is to minimise the expected time to collect the batches. Dependence between products to be in the same batch is what makes it relevant to position products strategically. Note that a deterministic counterpart of this problem does not exist.

## Instance generation and distribution

The topology of the storage facility is represented by a rectangular grid and the Cityblock distance metric determines the distance between points on the grid. In principle, any topology can be used since the problem is only specified by pairwise distances between locations.

The distribution is represented by a random binary vector where each index represents a product. An outcome of one means the product must be collected. We assume all products have the same marginal probability of being in a batch, such that the expected batch size is 5 across all instances (i.e. independent of grid size). The dependence structure is what makes positioning products a challenge. We define the dependence structure by a Gaussian copula with Bernoulli marginals. When sampling from the distribution we first draw from the copula and translate those to the marginal outcomes.

The copula is determined by a covariance matrix first constructed manually and then projected onto the semi-definite cone (Boyd & Vandenberghe, 2004) such that its eigenvalues are above 0.1. First, we set the covariance to one along the diagonal. Then between each pair of products we assign a random number in the interval  $[0, 0.3]$ . For one-fifth of product pairs, we then reset the dependence to be 0.5, and similarly to 0.8 for another fifth of product pairs. This reflects how dependence between products may look.

## Formulation

We use the convention that product 1 is the door and is placed at location 1. This will always be in the set of products to collect so that the vehicle must start and end at the door. Routes are represented as undirected graphs encoded by binary variables for whether a link between each location is active or not.

**Table A.6:** Overview of problem variables, sets and parameters for SLR.

Sets	
$L$	Set of locations
$E$	Set of location pairs
$P$	Set of products
Deterministic parameters	
$C$	Maximum number of products at one location
Stochastic parameters	
$P_s$	Set of products to collect in scenario $s$
First stage decisions	
$w_{pk}$	binary on if product $p$ is at location $k$
Second stage decisions	
$z_{kl}$	binary to make route between locations $k$ and $l$

Table A.2 gives an overview of variables, sets and parameters. The first-stage program

has formulation

$$\min_x \mathbb{E}[Q(x, \xi)] \quad (\text{A.7a})$$

s.t.

$$\sum_{p \in P} w_{pk} \leq C \quad \forall k \in L, \quad (\text{A.7b})$$

$$\sum_{k \in L} w_{pk} = 1 \quad \forall p \in P, \quad (\text{A.7c})$$

$$w_{11} = 1, \quad (\text{A.7d})$$

$$w_{pk} \in \{0, 1\} \quad (\text{A.7e})$$

where constraint (A.7b) limits the number of products placed at one location, constraint (A.7c) ensures each product is placed only once, and (A.7d) ensures the door is placed at position 1. The second-stage program has formulation

$$Q(x, \xi) = \min_z \sum_{(k,l) \in E} d_{kl} z_{kl} \quad (\text{A.8a})$$

s.t.

$$\sum_{\{(k,l) \in L: m \in (k,l)\}} z_{kl} \geq w_{pm} \quad \forall p \in P_s, \forall m \in L, \quad (\text{A.8b})$$

$$\sum_{(k,l) \in E} z_{kl} = \begin{cases} 1 & \text{if } |P_s| = 2 \\ |P_s| & \text{otherwise} \end{cases}, \quad (\text{A.8c})$$

$$\sum_{\{(k,l) \in L: m \in (k,l)\}} z_{kl} \leq 2 \sum_{p \in P_s} w_{pm} \quad \forall m \in L, \quad (\text{A.8d})$$

$$\sum_{\{(k,l) \in E'\}} z_{kl} \leq |E'| - 1 \quad \forall E' : E' \in \mathcal{P}(E \setminus \{1\}), |E'| > 1, \quad (\text{A.8e})$$

$$z_{kl} \in \{0, 1\} \quad \forall (k, l) \in E \quad (\text{A.8f})$$

where constraints (A.8b), (A.8c) and (A.8d) enforce that the route to collect products can only go by the locations where the relevant products are placed. Constraint (A.8e) prevents cycles that does not go by the door where  $\mathcal{P}(E \setminus \{1\})$  is the power-set of all edges not going to the door. In practice, this constraint is relaxed and enforced by row generation in the solution procedure. Notice that the second-stage formulation itself is altered by the stochastic outcome, which means warm-starts cannot be used for out-of-sample evaluation.

## References

- Ariyawansa, K. A., & Felt, A. J. (2002). *On a New Collection of Stochastic Linear Programming Test Problems*. Retrieved December 16, 2021, from <https://www4.uwsp.edu/math/afelt/slptestset/download.html>
- Ariyawansa, K. A., & Felt, A. J. (2004). On a New Collection of Stochastic Linear Programming Test Problems. *INFORMS Journal on Computing*, 16(3), 291–299. <https://doi.org/10.1287/ijoc.1030.0037>

- Boyd, S. P., & Vandenberghe, L. (2004). *Convex optimization*. Cambridge University Press.
- King, A. J., & Wallace, S. W. (2012, June). *Modeling with Stochastic Programming*. Springer New York, NY. <https://doi.org/10.1007/978-0-387-87817-1>
- Midler, J. L., & Wollmer, R. D. (1969). Stochastic programming models for scheduling airlift operations. *Naval Research Logistics Quarterly*, 16(3), 315–330.
- Sen, S., Doverspike, R. D., & Cosares, S. (1994). Network planning with random demand. *Telecommunication Systems*, 3(1), 11–30. <https://doi.org/10.1007/BF02110042>
- Vaagen, H., Wallace, S. W., & Kaut, M. (2011). Modelling consumer-directed substitution. *International Journal of Production Economics*, 134(2), 388–397. <https://doi.org/10.1016/j.ijpe.2009.11.012>

# Supplement B

## Joint Forecasting of Salmon Lice and Treatment Interventions

### 1 Biological relations

We use several relations based on the biology of lice that has been derived from empirical laboratory experiments. Due to wider temperature ranges and a requirement to extrapolate outside the temperature ranges in the experiments, we have re-estimated expressions for these relations. The data has been extracted from plots in the original papers using graphing software.

#### 1.1 Development time between stages and hatch rate

Lice development time has been examined experimentally by Hamre et al. (2019) in the temperature range  $6^{\circ}C$  to  $21^{\circ}C$ , and development in the larval (LR) phase by Samsing et al. (2016) in the temperature range  $5^{\circ}C$  to  $20^{\circ}C$ . While they report low or almost no development at  $3^{\circ}C$ , there are also exist accounts of lice developing at temperatures down to  $2^{\circ}C$ . For forecasting lice, we have temperatures in the range  $-0.5^{\circ}C$  to  $24^{\circ}C$ , and require these expressions to extrapolate beyond the given temperatures, which is not possible using the expressions estimated by Hamre et al. (2019) and Samsing et al. (2016) since they give either negative or undefined development times. The data in (Hamre et al., 2019; Samsing et al., 2016) both exhibit second-order tendencies in development times and are required to be positive, hence, we choose an expression for development times in the form:

$$\exp(a + bT + cT^2). \tag{B.1}$$

The newly fitted expressions coincide reasonably close to the original estimated expressions within their relevant temperature ranges. We use the same procedure to infer temperature dependent hatch rates from data in (Samsing et al., 2016).

#### 1.2 Time window and infectivity

The time window of infectivity is defined by the time it takes for larvae to develop and be able to attach to a new host until it runs out of nutrition and dies. We use data from Samsing et al. (2016) to estimate these functions using quantile regression at the 80th

quantile to get a conservative estimate of the width of the time window. The time to develop into the larval stage at the 80th quantile is estimated to

$$D_L^{80}(T) = \exp(1.492041 - 0.027941T - 0.001701T^2), \quad (\text{B.2})$$

and the width of the infestation window at the 80th quantile is estimated to

$$W^{80}(T) = \exp(0.450114 + 0.090447T - 0.005636T^2). \quad (\text{B.3})$$

The lower and upper limits of the time window is then expressed as

$$W_l(T) = D_L^{80}(T), \quad (\text{B.4})$$

$$W_u(T) = D_L^{80}(T) + W^{80}(T), \quad (\text{B.5})$$

respectively.

Skern-Mauritzen et al. (2020) shown there are important interactions between age and temperature to determine infestation success. The temperature effect was examined experimentally by Samsing et al. (2016) in the temperature range  $5^\circ\text{C}$  to  $20^\circ\text{C}$ , while the interaction effect between age and temperature was examined by Skern-Mauritzen et al. (2020) in the temperature range  $5^\circ$  to  $15^\circ\text{C}$ . The expression for infectivity estimated by Skern-Mauritzen et al. (2020) is likely over-fitted since it deviates significantly from the data by (Samsing et al., 2016) at  $20^\circ\text{C}$ , and also gives unreasonable predictions at low temperatures. Skern-Mauritzen et al. (2020) report concerns about over-fitting themselves. Thus, we estimate a new expression for infectivity more well-suited for extrapolation.

We use the time window  $[W_l(T), W_u(T)]$  as a starting point, and specify a uni-modal function of age  $A$  within this range that has temperature dependent parameters. Infectivity then takes the expression:

$$I(A, T) = \begin{cases} C(T) \frac{[A]^{a(T)}(1-[A])^{b(T)}}{\mathcal{B}(a(T), b(T))} & \text{if } W_l(T) \leq A \leq W_u(T) \\ 0 & \text{otherwise,} \end{cases} \quad (\text{B.6})$$

where  $A$  is age in weeks,  $[A] = \frac{A - W_l(T)}{W_u(T) - W_l(T)}$  is the fractional age within the time window, and  $\mathcal{B}(\cdot, \cdot)$  is the Beta function. This expression has been validated to adhere to the data from (Samsing et al., 2016) and has reasonable properties for extrapolation outside the given temperature range of the data. The temperature dependent parameters are estimated to

$$C(T) = \exp(-3.068236 + 0.447180T - 0.012281T^2), \quad (\text{B.7})$$

$$a(T) = \exp(4.465295 - 0.619186T + 0.018293T^2), \quad (\text{B.8})$$

$$b(T) = \exp(0.766574 + 0.133997T - 0.010022T^2), \quad (\text{B.9})$$

and the function  $I(A, T)$  is illustrated in the paper. For evaluation of infestation success for ages between  $s - 1$  and  $s$  weeks, we use the average infectivity

$$\bar{I}_s(T) = \int_{s-1}^s I(A, T) dA, \quad (\text{B.10})$$

during the week that leads up to week  $s$ .

## 2 Structural instability

There is evidence of structural instability in the lice model for the given time-series data. Namely, there is a systematic drift in parameter estimates over time, implying changes in the data generation mechanism not captured by our model. To detect structural instability, we perform rolling estimation within two year time windows, and compare their respective parameter estimates. Rolling parameter estimates are illustrated in Figure B.1 for the lice model, and in Figure B.2 for the treatment model. Note that we fix the effect of cleaner fish for time windows that contain data after 2018 since there is no new data on cleaner fish after that point.

Foremost, we see systematic changes in the lice model generally occurring in the period 2016–2019, and some particular violations of structural stability include:

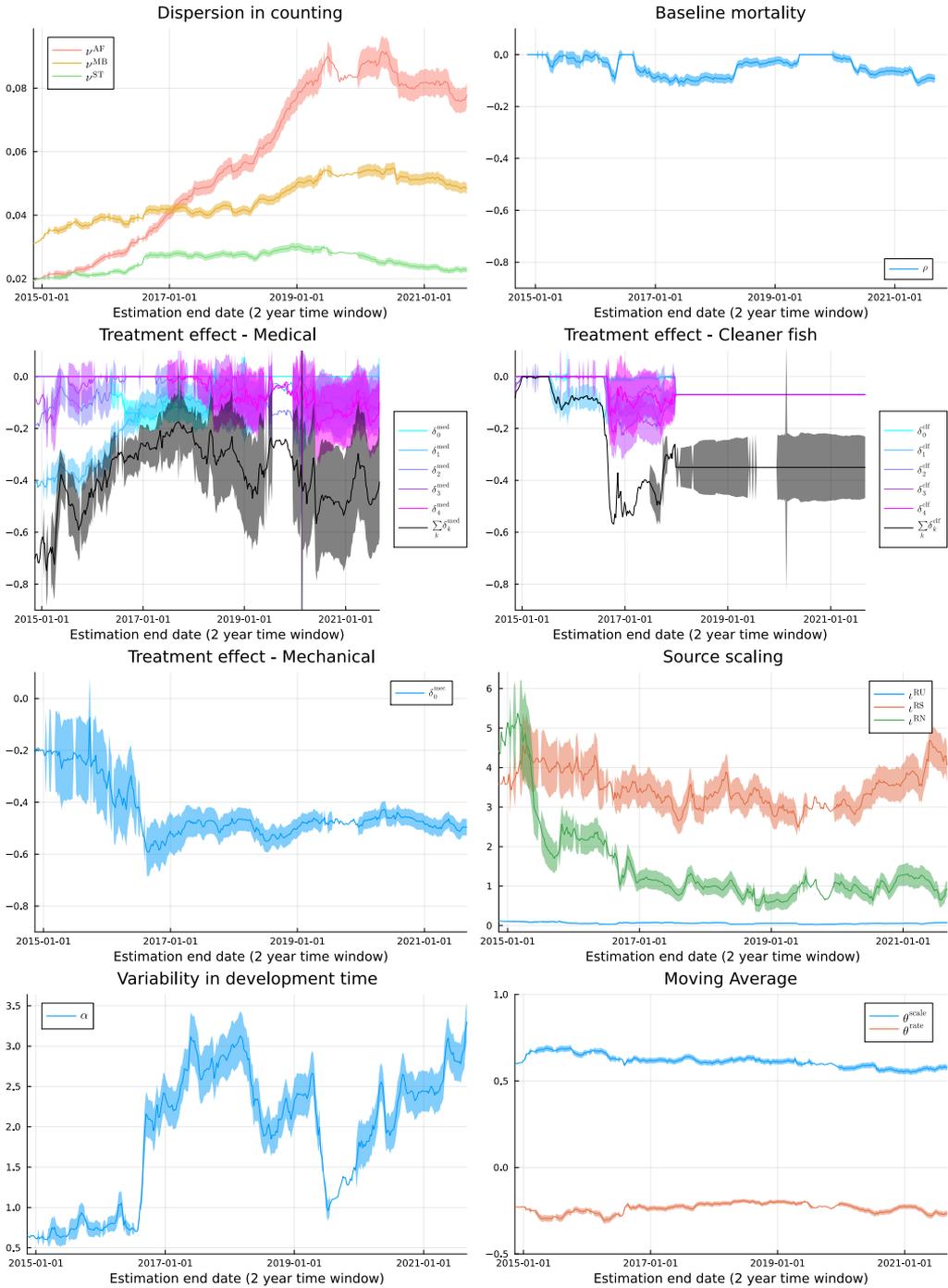
- There seems to be a decrease in dispersion of counts for adult female (AF) lice (by higher values of  $\nu^{\text{AF}}$ ). This may, among other things, be explained by more standardized counting practices (Solberg et al., 2018).
- We see an overall decrease in the effect of medical treatments up to the beginning of 2017. This is a known phenomenon caused by resistance towards the used substances (Jensen et al., 2020).
- There is an increasing effect of mechanical treatments up to 2017. This may be explained by accounts that better treatment practices learned in the industry during 2016–2018.

There are several regulatory changes during this period (Lovdata, 2016, 2017, 2018) that may explain some of these effects. In previous work, Aldrin et al. (2019) concluded that the treatment effect should be on the 1st lag, using data up to the end of 2016. Our analysis shows a shift occurred later and we, in contrast, also use a 0-lagged effect of treatment. It is mainly the introduction of mechanical treatments after 2017 that is the cause of this shift. If we do not distinguishing medical and mechanical treatment, we get severe structural instabilities in the timing of treatment effects.

For the treatment model, the main structural instability is in the effect of adult female lice through the parameters  $\gamma^{\text{AF}}$  and  $\pi^{\text{AF}}$  that seem to be increasing. This may be linked to the dispersion in these lice counts that has become lower during the same time period, meaning the counts of adult female lice provide better signals to infer treatments. There also seems to be an increased effect of using past treatments to infer the probability of treatment, through the parameter  $\phi^{\text{AF}}$ .

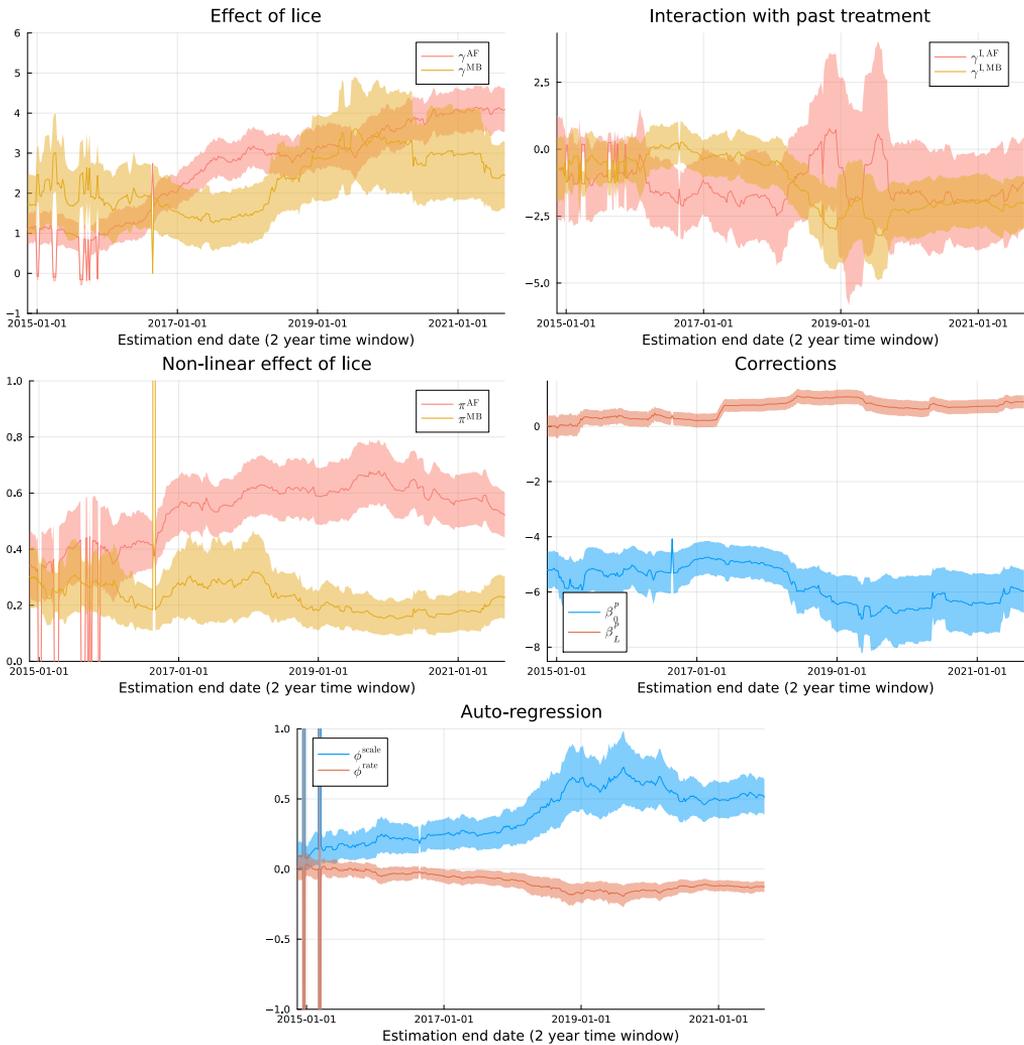
## 3 Quantifying the cost of lice treatments

We make some rough assumptions to quantify the cost of treatment. First, we assume we have a cage of fish at 2 kg and that we have a baseline daily growth rate of 0.5% given a temperature of 10°C (Skretting, 2012). Before a treatment, the fish are starved for 3–5 days to reduce stress and, additionally, the treatment itself cause reduced growth afterwards. Combined, these cause a approximate 32% decreased growth for 1–2 weeks (Walde et al., 2022). Weekly mortality rate increase approximately 1.17 percentage points as a result of mechanical treatments (Walde et al., 2021). The treatment operation in itself costs approximately 250 000 NOK per cage, which we distribute evenly across 200 000 fish. We assume an approximate sales price of 65 NOK/kg. To estimate the cost of lice treatments, we forecast growth and mortality for the next 20 weeks within two scenarios:



**Figure B.1:** Lice model: Rolling parameter estimates within three year time windows. Ribbons give 95% confidence intervals derived by the Delta method.

(i) including the effect of treatments, or (ii) ignoring the effect of treatments. The cost of lice is estimated as the difference in revenue between these two scenarios. The relative

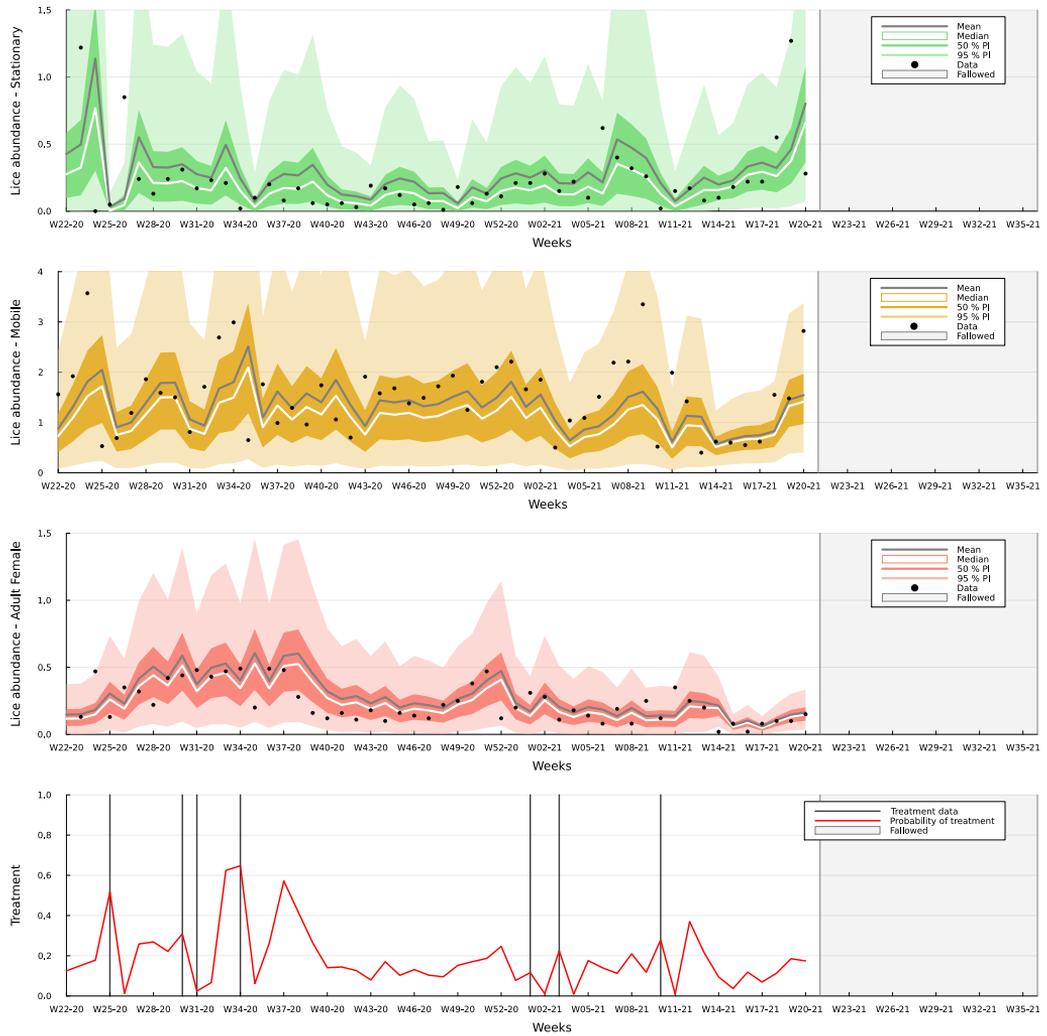


**Figure B.2:** Treatment model: Rolling parameter estimates within three year time windows. Ribbons give 95% confidence intervals derived by the Delta method.

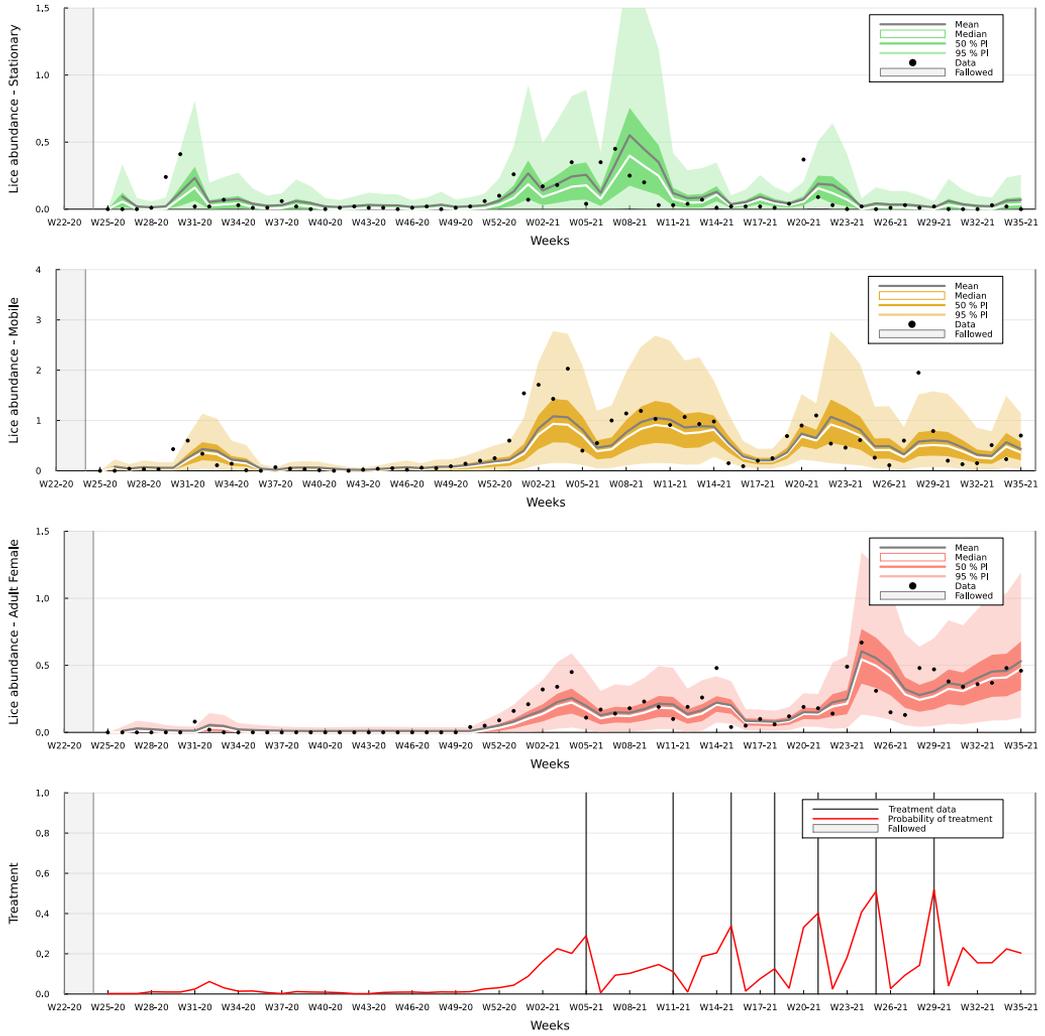
measure of lice-induced loss equals the cost of lice divided by the revenue that includes treatments. More precise accounts of the operational cost of treatment are provided by Abolofia et al. (2017) and Iversen et al. (2017).

## 4 Forecasts distributions for additional sites

One-week-ahead forecast distributions for three additional sites are shown in Figure B.3, B.4 and B.5.



**Figure B.3:** Out-of-sample one week ahead forecast distributions as prediction intervals and probability of treatment.



**Figure B.4:** Out-of-sample one week ahead forecast distributions as prediction intervals and probability of treatment.



**Figure B.5:** Out-of-sample one week ahead forecast distributions as prediction intervals and probability of treatment.

## References

- Abolofia, J., Asche, F., & Wilen, J. E. (2017). The Cost of Lice: Quantifying the Impacts of Parasitic Sea Lice on Farmed Salmon. *Marine Resource Economics*, *32*(3), 329–349. <https://doi.org/10.1086/691981>
- Aldrin, M., Jansen, P. A., & Stryhn, H. (2019). A partly stage-structured model for the abundance of salmon lice in salmonid farms. *Epidemics*, *26*, 9–22. <https://doi.org/10.1016/j.epidem.2018.08.001>
- Hamre, L. A., Bui, S., Oppedal, F., Skern-Mauritzen, R., & Dalvin, S. (2019). Development of the salmon louse *Lepeophtheirus salmonis* parasitic stages in temperatures ranging from 3 to 24°C. *Aquaculture Environment Interactions*, *11*, 429–443. <https://doi.org/10.3354/aei00320>
- Iversen, A., Hermansen, Ø., Nystøyl, R., & Hess, E. J. (2017). *Kostnadsutvikling i lakseoppdrett – med fokus på før- og lusekostnader*. Nofima AS. Retrieved December 5, 2020, from <https://nofima.brage.unit.no/nofima-xmlui/handle/11250/2481501>  
Accepted: 2018-02-01T14:47:27Z.
- Jensen, E. M., Horsberg, T. E., Sevatdal, S., & Helgesen, K. O. (2020). Trends in de-lousing of Norwegian farmed salmon from 2000–2019—Consumption of medicines, salmon louse resistance and non-medicinal control methods. *PLOS ONE*, *15*(10), e0240894. <https://doi.org/10.1371/journal.pone.0240894>
- Lovdata. (2016, March). Forskrift om endring i forskrift om bekjempelse av lakselus i akvakulturanlegg - Lovdata. Retrieved September 8, 2021, from <https://lovdata.no/dokument/LTI/forskrift/2016-03-14-242>
- Lovdata. (2017, March). Forskrift om endring i forskrift om bekjempelse av lakselus i akvakulturanlegg - Lovdata. Retrieved July 5, 2023, from <https://lovdata.no/dokument/LTI/forskrift/2017-03-06-275>
- Lovdata. (2018, April). Forskrift om endring i forskrifter om akvakultur for deres tilpasning til transport, oppbevaring, bruk og produksjon av rensefisk - Lovdata. Retrieved July 5, 2023, from <https://lovdata.no/dokument/LTI/forskrift/2018-04-19-674>
- Samsing, F., Oppedal, F., Dalvin, S., Johnsen, I., Vågseth, T., & Dempster, T. (2016). Salmon lice (*Lepeophtheirus salmonis*) development times, body size, and reproductive outputs follow universal models of temperature dependence. *Canadian Journal of Fisheries and Aquatic Sciences*, *73*(12), 1841–1851. <https://doi.org/10.1139/cjfas-2016-0050>
- Skern-Mauritzen, R., Sissener, N. H., Sandvik, A. D., Meier, S., Sævik, P. N., Skogen, M. D., Vågseth, T., Dalvin, S., Skern-Mauritzen, M., & Bui, S. (2020). Parasite development affect dispersal dynamics; infectivity, activity and energetic status in cohorts of salmon louse copepodids. *Journal of Experimental Marine Biology and Ecology*, *530–531*, 151429. <https://doi.org/10.1016/j.jembe.2020.151429>
- Skretting. (2012). Fôrkatalog.
- Solberg, I., Finstad, B., Berntsen, H. H., Diserud, O. H., Frank, K., Helgesen, K. O., Jeong, J., Kristoffersen, A. B., Nytrø, A. V., Revie, C. W., Sivertsgård, R., Solvang, T., Sunde, L. M., Thorvaldsen, T., Uglem, I., & Mo, T. A. (2018). *Kartlegging og testing av metodikk for telling av lakselus og beregning av luseforekomst* (tech. rep.). Norsk Institutt for Naturforskning (NINA). Retrieved July 20, 2022, from <https://brage.nina.no/nina-xmlui/handle/11250/2560950>  
Accepted: 2018-09-05T11:33:46Z.
- Walde, C. S., Jensen, B. B., Pettersen, J. M., & Stormoen, M. (2021). Estimating cage-level mortality distributions following different delousing treatments of Atlantic salmon (*Salmo salar*) in Norway. *Journal of Fish Diseases*, *44*(7), 899–912. <https://doi.org/10.1111/jfd.13348>
- Walde, C. S., Stormoen, M., Pettersen, J. M., Persson, D., Røsæg, M. V., & Bang Jensen, B. (2022). How delousing affects the short-term growth of Atlantic salmon (*Salmo salar*). *Aquaculture*, *561*, 738720. <https://doi.org/10.1016/j.aquaculture.2022.738720>



# Supplement C

## Harvest Planning under Uncertainty in Salmon Aquaculture

### 1 Statistical and biological models

We explain the statistical models used to represent uncertainty. For conclusions in the main paper, these statistical models represent what we refer to as the prescribed distribution which is denoted  $P$ .

#### 1.1 Market prices

Forecasting prices is normally a complicated process that is outside the scope of this paper. Farmers would normally rely on internal expertise or external analysts to provide forecasts, and we have access to neither of these. Instead, we replicate a similar level of insight based on known historical data.

First, an overall trend is derived by smoothing the historical data using centred moving averages of 15 week time windows. This trend represents a price forecast that might have been provided by other means, and the fact that future observations are included in the moving average represents the insight that might go into forecasts.

Second, we represent uncertainty around this trend by an auto-regressive time series model. There are nine price classes that are highly dependent, represented by the random variable  $X_{kt}$  for price class  $k = 1, \dots, 9$  and time  $t$ . Similarly, we let  $\mu_{kt}$  represent the trend. We express the price process as an AR(3) model on logarithmic scale, expressed as

$$\log(X_{kt}/\mu_{kt}) = \sum_{l=1}^3 \phi_{kl} \log(X_{k,t-l}/\mu_{k,t-l}) + \eta_{kt}, \quad \forall k \quad (\text{C.1})$$

where we assume the noise process  $\eta_t \sim \text{Normal}(0, \Sigma)$  is independent and identically distributed (iid) across time  $t$ . The parameters  $\phi \in \mathbb{R}^{9 \times 3}$  are estimated on historical data using the R-package `stats`.

The conditional dependence between price classes is contained in the covariance matrix  $\Sigma$ . Due to strong dependence between price classes, we decompose  $\Sigma$  by Principal Component Analysis (PCA) to represent 99.14% of the variance through six latent factors.

Then,

$$\eta_t \approx T\epsilon_t, \quad (\text{C.2})$$

where we assume  $\epsilon_t \sim \text{Normal}(0, I_6)$ ,  $I_6$  is the identity matrix of size six, and  $T \in \mathbb{R}^{9 \times 6}$  is the transformation matrix derived from PCA. The signs of the columns in  $T$  follow the convention that positive values of each factor in  $\epsilon_t$  leads to positive values of  $\sum_{k=1}^9 \eta_{kt}$ . Scenario generation acts on the latent variables  $\epsilon_t$  which are transformed to  $X_t$  afterwards.

### Maximal price variation

We quantify the variability in the price process to derive bounds on maximal possible value gain between stages. Let

$$\tilde{X}_{kt} = \log(X_{kt}/\mu_{kt}), \quad (\text{C.3})$$

denote the log-transformed, mean-corrected version of the price process. We then have that

$$\text{Std}(\tilde{X}_{kt}) = \sqrt{\mathbb{E}[\tilde{X}_{kt}\tilde{X}_{kt}]} = \sigma_k \sqrt{\gamma_k(0)}, \quad (\text{C.4})$$

where  $\gamma_k(\cdot)$  is the auto-covariance function of  $\tilde{X}_k$  and  $\sigma_k^2$  is the variance of  $\eta_{kt}$  (Brockwell & Davis, 1991). To bound price variations, we truncate  $\eta_{kt}$  to its 99% prediction interval. Thus,

$$\Phi^{-1}(0.995)\sigma_k \sqrt{\gamma_k(0)}, \quad (\text{C.5})$$

forms a bound on  $\tilde{X}_{kt}$ , where  $\Phi^{-1}$  is the inverse CDF of a standard normal distribution. This is used to derive the maximal deviation from the price trend  $\mu_{kt}$ , where

$$X_{kt} \leq \mu_{kt} \exp\left(\Phi^{-1}(0.995)\sigma_k \sqrt{\gamma_k(0)}\right). \quad (\text{C.6})$$

From this bound, we can bound the value gain from one stage to the next where we also account for maximal growth rate and transition to different price classes.

## 1.2 Lice and treatments

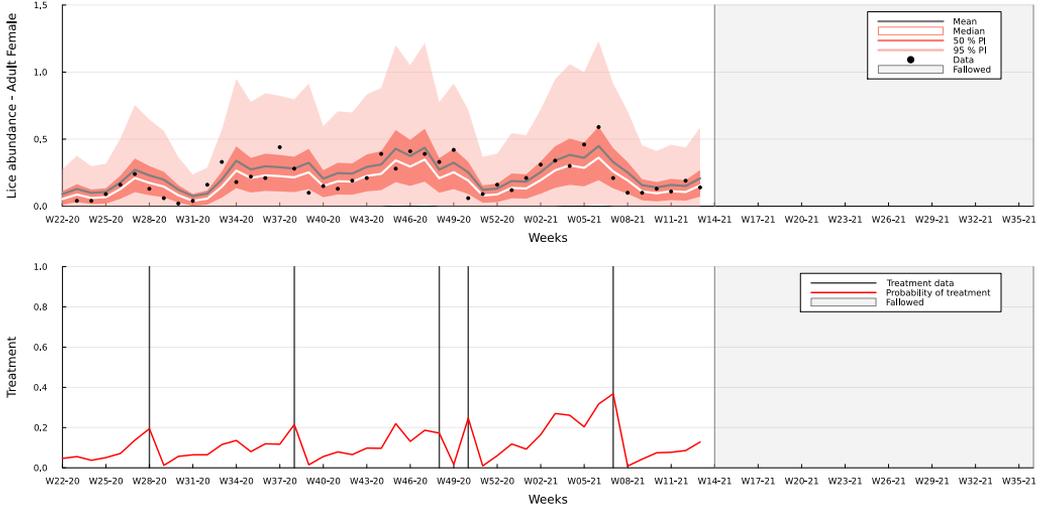
Development of parasitic lice and their counter-acting treatments is a complicated spatio-temporal phenomenon. Specifically, there is strong interaction between sites due to transportation of lice by ocean currents. The statistical model used to forecast this phenomenon is addressed in a different publication by Narum and Berentsen (2024) and we provide a short description here.

The dynamics of lice is modelled as a stochastic process  $Y$ . There is limited flexibility in the decision to perform treatments; hence, we model this as a stochastic process  $V$ , which also avoids decision-dependent uncertainty. There are interactions both ways between lice and treatments; namely, more lice increases the probability of treatment, while treatment decreases the level of lice. Spatial effects are accounted for by transportation patterns derived from hydrodynamic simulation.

Let  $i$  denote a site, and  $t$  the week. The distribution for lice counts is then modelled as

$$(n_{it}Y_{it} \mid V_{it}, \mathcal{F}_{t-1}) \sim \text{NegativeBinomial}(n_{it}\mu_{it}, n_{it}\nu), \quad (\text{C.7})$$

where  $n_{it}$  is the number of fish collected to count lice, and we are interested in the *average* lice count ( $Y_{it}$ ). Here,  $\mu_{it}$  is an expression for the expected average lice count and  $\nu$  is



**Figure C.1:** Out-of-sample forecasts of lice and treatments.

a dispersion parameter. There is actually three different stages of lice that are counted, which we ignore in this description. The distribution for treatments is modelled as

$$(V_{it} \mid \mathcal{F}_{t-1}) \sim \text{Bernoulli}(p_{it}), \quad (\text{C.8})$$

where  $p_{it}$  is an expression for the probability of treatment. Observe that lice depends on treatments within the same week, while treatment relies only on lice in the previous week. This allows estimating these models separately, but also means we must forecast treatments in the next week before lice in the next week. Figure C.1 illustrates one-week-ahead forecasting distributions of lice and treatments. Note that the expressions for  $\mu_{it}$  and  $p_{it}$  are quite involved and rely on far-reaching historical values.

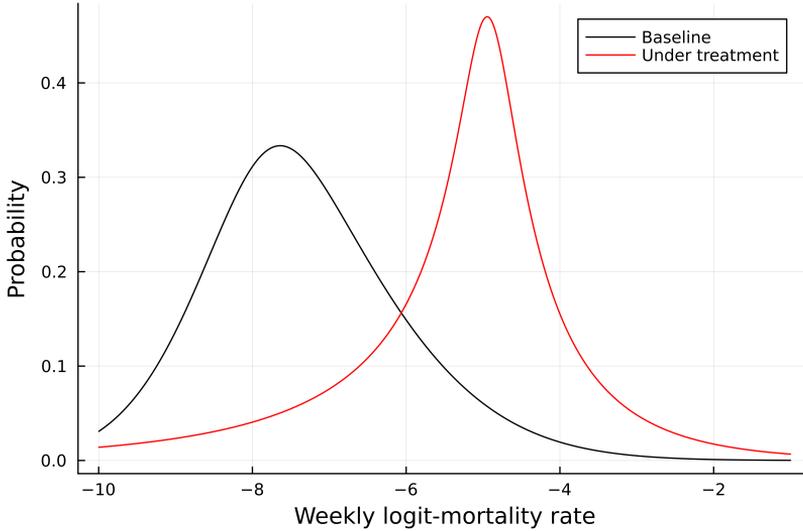
### 1.3 Mortality

Fish mortality is stochastic in the sense that farmers are sometimes surprised by realised mortality rates. It is known that the mortality rate is linked to lice treatments and to presence of disease. Simultaneous occurrence of disease and treatment can be especially unfortunate. For this work, we emphasise forecasting of treatments and not disease, but if a forecasting model for disease was provided we could easily incorporate it. We derive the mortality distribution conditional on lice treatment, meaning the effect of disease is incorporated unconditionally. We use summary statistics from the work by Walde et al. (2021) and fit a parametric distribution to these.

Specifically, we fit the distribution to the following percentiles: 1st, 5th, 10th, P25th, 50th, 75th, 90th, 95th and 99th. We use the *sinh-arcsinh* (SHASH) distribution, which is a flexible uni-modal distribution of four parameters which allows good control over the first four moments (Jones & Pewsey, 2009). It is specified as a sinh and arcsinh transformation of a normal distribution.

Let  $Z$  denote the mortality rate. The expression for the mortality distribution is then

$$\text{logit}(Z \mid V) \sim \text{SHASH}(\mu, \sigma, \eta, \delta), \quad (\text{C.9})$$



**Figure C.2:** Mortality distributions, with and without treatment.

where different parameters  $(\mu, \sigma, \eta, \delta)$  are found for the two cases  $V = 0$  and  $V = 1$ . Parameters are fitted by minimising the summed squared error between the distribution and the specified percentiles from the data, weighted by  $\max\{1 - p, p\}$  to account for less statistical significance in the more extreme percentiles. Figure C.2 illustrates the conditional mortality distributions.

## 1.4 Growth

Growth rates as a function of temperature and weight are collected from Skretting’s industry manual.<sup>1</sup> Growth rates are also affected by lice treatments, and we use the estimates by Walde et al. (2022) to reflect that the growth rate is reduced due to treatment. For temperatures, we use historical values from the NorKyst800 model (Albretsen et al., 2011).

## 2 Dual formulation under relaxation of integers

We present the dual formulation of the Aquaculture Harvest Planning (AHP) problem under relaxation of integrality in the binary variables  $y^\diamond$  and  $h^\diamond$ . Dual variables are listed as multipliers to all constraints in the primal formulation in the paper. We refer to Rockafellar (1999) for details on dualisation of multistage stochastic programs.

Upon relaxation of binary variables, we also insert the lower bounds

$$0 \leq y^\diamond, h^\diamond, \tag{C.10}$$

while tightening constraints and initial conditions ensure these are never larger than 1.

<sup>1</sup>Skretting (<https://www.skretting.com/>) is a feed production company for aquaculture.

The dual objective to be minimised in each stage  $t$  is stated as:

$$f_t^*(\lambda_t, \xi_t) = N^{\text{boat}}(\kappa_t^{\text{bflx}} + 2\kappa_t^{\text{blmt}}) + \left( B^{\text{cmp}} - \sum_{i \in \bar{\mathcal{A}}_t} w_{it} \tilde{y}_{it} \right) \beta_t^{\text{cmp}} + \sum_{i \in \mathcal{O}_t} B_i^{\text{loc}} \beta_{it}^{\text{loc}} \\ + \mathbb{I}[t = 1] y_{i0} \left( \sum_{i \in \mathcal{A}_t} (\sigma_{it} \phi_{it} + \tau_{it}^{h^\diamond}) + \sum_{i \in \mathcal{O}_t} \tau_{it}^{y^\diamond} \right) \quad (\text{C.11})$$

where  $\mathbb{I}[t = 1]$  is an indicator for whether  $t = 1$  and  $y_{i0}$  is the initial in-going fish count. The first line represents operational costs, and the latter represents the value of the initial in-going biomass. The dual constraints have formulations:

- Harvest quantity

$$\phi_{it} \geq (\pi_{it} r^{\text{hog}} - c^{\text{slaughter}}) w_{it} - \nu_{it}^{h^\diamond} - w_{it} \kappa_{it}^C, \quad (h_{it}) \quad \forall i \in \mathcal{A}_t. \quad (\text{C.12})$$

- Out-going fish count

$$\phi_{it} \geq \mathbb{E}^t [\sigma_{i,t+1} \phi_{i,t+1}] - c^{\text{feed}} \Delta w_{it} - \nu_{it}^{y^\diamond} - w_{it} (\beta_t^{\text{cmp}} + \beta_{it}^{\text{loc}}), \quad (y_{it}) \quad \forall i \in \mathcal{O}_t, \quad (\text{C.13})$$

- Boat count

$$0 \geq \kappa_{it}^{h^\diamond} + C^{\text{boat}} \kappa_{it}^C - (\kappa_t^{\text{bflx}} + \kappa_t^{\text{blmt}}) - c^{\text{boat}}, \quad (b_{it}) \quad \forall i \in \mathcal{A}_t. \quad (\text{C.14})$$

- Active status

$$\tau_{it}^{y^\diamond} \geq M_{it} \nu_{it}^{y^\diamond} - c^{\text{active}} - c^{\text{delouse}} \psi_{it} + \mathbb{E}^t [\tau_{i,t+1}^{h^\diamond}] + \mathbb{E}^t [\tau_{i,t+1}^{y^\diamond}], \quad (y_{it}^\diamond) \quad \forall i \in \mathcal{O}_{t+1}, \quad (\text{C.15a})$$

$$\tau_{it}^{y^\diamond} \geq M_{it} \nu_{it}^{y^\diamond} - c^{\text{active}} - c^{\text{delouse}} \psi_{it} + \mathbb{E}^t [\tau_{i,t+1}^{h^\diamond}], \quad (y_{it}^\diamond) \quad \forall i \in \mathcal{C}_{t+1}. \quad (\text{C.15b})$$

- Harvest status

$$\tau_{it}^{h^\diamond} \geq M_{it} \nu_{it}^{h^\diamond} - \kappa_{it}^{h^\diamond}, \quad (h_{it}^\diamond) \quad \forall i \in \mathcal{A}_t. \quad (\text{C.16})$$

- Dual variable bounds

$$0 \leq \kappa_t^{\text{bflx}} \leq c^{\text{boat}} / 2, \quad (z_t^{\text{bflx}}) \quad (\text{C.17})$$

$$0 \leq \kappa_t^{\text{blmt}} \leq P^{\text{boat}}, \quad (z_t^{\text{blmt}}) \quad (\text{C.18})$$

$$0 \leq \beta_t^{\text{cmp}} \leq P^{\text{cmp}}, \quad (z_t^{\text{cmp}}) \quad (\text{C.19})$$

$$0 \leq \nu_{it}^{y^\diamond}, \nu_{it}^{h^\diamond}, \kappa_{it}^C, \kappa_{it}^{\text{bflx}}, \kappa_{it}^{\text{blmt}}, \kappa_{it}^{h^\diamond}, \beta_{it}^{\text{loc}}. \quad (\text{slack variables}) \quad (\text{C.20})$$

### 3 Problem instances

The case study data is obtained by a combination of public data, insights from proprietary data, and simulation from statistical models fitted to real data. We derive synthetic operational data so that we can insert our decision model at any point in time. We start out with all real sites along the Norwegian coastline (those active in September 2021) using their real temperatures within the years 2012—2022, and assume all sites are active throughout this time span. These interact spatially by emission of parasitic lice between sites. Production cycles are assumed to have a length of 78 weeks (18 months) with fallowing (closing down) 8 weeks between cycles. By random initialisation times, this infers all production cycles. The statistical lice and treatment models are initialised at the first time step with zero lice and simulated in a single run to obtain synthetic historical data. Weight is initialised at 0.3 kg at the start of each cycle and forwarded conditional on temperature and loss in weight gain due to lice treatments. Each production cycle starts with 200 000 fish, reduced by simulated mortality rates conditional on lice treatments. Site capacity is set to their real number, and a cage is assumed to have a capacity of 790 tonnes; thus, the number of cages per site can be inferred. Biomass limits for sites are set to their real numbers, while the company-wide biomass limit is set to 60% of the combined site capacities. To construct a starting point for the stochastic optimisation model, we must infer how much has already been harvested, but historical harvest plans are proprietary data. We generate historical harvest actions by assuming farmer keeps the fish until the biomass limit is reached, at which point they harvest the largest fish first in chunks of size according to well-boat capacity. This rule-of-thumb for harvest planning is common in the industry.

### 4 K-means clustering on trees

The minimum transportation distance approach to scenario reduction can, for random variables, be solved heuristically through clustering (Rujeerapaiboon et al., 2022). However, for stochastic process we must also account for information structure; i.e., time-development of information states represented through branching in a scenario tree. The essence of the nested distance (Pflug & Pichler, 2012) is to incorporate causality in measures of distance between stochastic processes by prioritising distances in earlier stages first. We customise the k-means algorithm (Lloyd, 1982) to account for time structure in stochastic processes and for causality in distance metrics.

First, we relax the nested distance in part by using the distance metric

$$D(\omega, \omega') = \sum_{t \in \mathcal{T}} d_t(\omega, \omega') e^{-\rho(t-1)}, \quad (\text{C.21})$$

between outcomes  $\omega$  and  $\omega'$ , where the metric  $d_t(\omega, \omega')$  is defined in the paper. This still prioritises distances in one stage over the next but only by a ratio of  $e^\rho$  instead of absolute priority. For comparing distances, (C.21) corresponds to the nested distance when letting  $\rho \rightarrow \infty$ . For our purposes, we let  $\rho = 1$ .

Assume that the branching structure of the scenario tree is provided and that the values in each node are initially randomised. Let us then assume we have a large number of sampled paths  $\tilde{\Omega}$ , on which we may perform clustering using the distance metric  $D(\omega, \omega')$ . Clustering provides a partitioning of  $\tilde{\Omega}$  where each partition is associated to a path in the scenario tree. The next step in the k-means algorithm would be to insert the average of each partition as the centre of each cluster. Instead, we insert the *conditional* average

in each node of the scenario tree to account for its branching structure. This procedure is repeated from the step of partitioning  $\tilde{\Omega}$  until the improvement is low. This converges reasonably well, but is nonetheless a heuristic.

## References

- Albretsen, J., Sperrevik, A. K., Staalstrøm, A., Sandvik, A. D., Vikebø, F., & Asplin, L. (2011). NorKyst-800 Rapport nr. 1 : Brukermanual og tekniske beskrivelser. 48 s. Retrieved June 22, 2021, from <https://imr.brage.unit.no/imr-xmlui/handle/11250/113865>  
Accepted: 2011-07-13T11:53:51Z.
- Brockwell, P. J., & Davis, R. A. (1991). *Time Series: Theory and Methods*. Retrieved May 10, 2022, from <https://link.springer.com/book/10.1007/978-1-4419-0320-4>
- Jones, M. C., & Pewsey, A. (2009). Sinh-arcsinh distributions. *Biometrika*, 96(4), 761–780. Retrieved October 10, 2022, from <https://www.jstor.org/stable/27798865>
- Lloyd, S. (1982). Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2), 129–137. <https://doi.org/10.1109/TIT.1982.1056489>
- Narum, B. S., & Berentsen, G. D. (2024). Joint Forecasting of Salmon Lice and Treatment Interventions in Aquaculture Operations  
Under review.
- Pflug, G. C., & Pichler, A. (2012). A Distance For Multistage Stochastic Optimization Models. *SIAM Journal on Optimization*, 22(1), 1–23. <https://doi.org/10.1137/110825054>
- Rockafellar, R. (1999). Duality and optimality in multistage stochastic programming. *Annals of Operations Research*, 85(0), 1–19. <https://doi.org/10.1023/A:1018909508556>
- Rujeerapaiboon, N., Schindler, K., Kuhn, D., & Wiesemann, W. (2022). Scenario reduction revisited: Fundamental limits and guarantees. *Mathematical Programming*, 191(1), 207–242. <https://doi.org/10.1007/s10107-018-1269-1>
- Walde, C. S., Jensen, B. B., Pettersen, J. M., & Stormoen, M. (2021). Estimating cage-level mortality distributions following different delousing treatments of Atlantic salmon (*salmo salar*) in Norway. *Journal of Fish Diseases*, 44(7), 899–912. <https://doi.org/10.1111/jfd.13348>
- Walde, C. S., Stormoen, M., Pettersen, J. M., Persson, D., Røsæg, M. V., & Bang Jensen, B. (2022). How delousing affects the short-term growth of Atlantic salmon (*Salmo salar*). *Aquaculture*, 561, 738720. <https://doi.org/10.1016/j.aquaculture.2022.738720>