Navigating trade-offs in green fleet renewal

A multi-criteria decision support framework for strategic fleet management

MT54035-20: MT MSc Thesis

Jelmer Pentinga

Delft University of Technology

Thesis for the degree of MSc in Marine Technology in the specializations of *Maritime Operations and Management*, *Marine Engineering*, *Ship Design*, and *Ship Hydromechanics*.

Navigating trade-offs in green fleet renewal

A multi-criteria decision support framework for strategic fleet management

by

Jelmer Pentinga

Performed at

Port of Rotterdam

This thesis (MT.24/25.051.M) is classified as confidential in accordance with the general conditions for projects performed by the TU Delft.

28 August, 2025

Company Supervisors:
Company supervisor: G.J. (Bob) Madlener

Thesis Exam Committee:
Chair: Dr. ir J.F.J. (Jeroen) Pruyn
Responsible professor: Prof. dr. ir E.B.H.J. (Edwin) van Hassel
Staff Member: Dr. J.M. (Jaap) Vleugel
Company Member: G.J. (Bob) Madlener

Author Details:
Student Number: 4883608

Preface

This thesis marks the conclusion of my journey in Marine Technology at Delft University of Technology. It was written in collaboration with the Port of Rotterdam, whose support and ambition to lead in sustainable port operations inspired the topic and shaped the scope of this work.

I would like to express my sincere gratitude to the members of the Port of Rotterdam fleet renewal team, in particular Bob Madlener, my supervisor, as well as Willem Toet and Bernd Laas, for their openness to discussing my ideas and challenges. Their input was invaluable in delivering an insightful and practical result that can help guide fleet operators in transitioning their fleets in the most efficient manner.

I am also grateful to my academic supervisors, Edwin van Hassel and Jeroen Pruyn, for their guidance and constructive feedback throughout the process. Their support has helped me further develop my skills and knowledge, ultimately leading to this report, the final product of my time in Delft. I hope you enjoy reading it.

Jelmer Pentinga
Delft, August 2025

Abstract

The maritime sector faces mounting pressure to decarbonise in alignment with global climate objectives and regional frameworks such as the EU Fit-for-55 package and the IMO net-zero targets. For fleet operators such as the Port of Rotterdam, which aims to reduce Scope 1 and 2 emissions by 90% in 2030 and sail emissions-free from 2035, this involves navigating complex trade-offs between sustainability, cost-efficiency, and operational readiness. This thesis investigates how the PoR can optimise its fleet renewal strategy to minimise total polluting emissions and transition costs while maintaining functional capacity.

To address this challenge, a hybrid decision-support framework was developed, combining multi - objective optimisation with multi-criteria decision analysis. The optimisation model produced Pareto-optimal strategies, using the ε-constraint method, that balance lifecycle CO_2 emissions, total cost of ownership, and local air pollution. Multi-criteria decision analysis, using TOPSIS, enabled inclusion of stakeholder preferences in the classification of different transition schedules under varying assumptions about fuel types, material choice, and production location. Scenario analyses were performed to assess the robustness of the combined framework against various economic outlooks and environmental choices, such as fuel type, hull material, and production location.

The results show that lifecycle emissions are largely shaped by design decisions such as material and energy source, whereas local pollutants are very sensitive to the replacement schedule. The total cost of ownership shows limited sensitivity to scheduling (1–2% variation), while battery production and dismantling emerge as the dominant drivers of greenhouse gas emissions and financial impact. The inclusion of CO_2 emission depreciation significantly altered the schedules of optimal results, raising ethical and policy considerations. Certain vessel classes demonstrated robust scheduling behaviour, in various strategic choices and economic scenarios, identifying them as low regret alternatives. Other classes were more sensitive to changes in the strategic choices or stakeholder preferences.

The framework successfully supported trade-off navigation, revealing how rankings changed under varying stakeholder preferences and scenario assumptions. However, several simplifications remain. The decoupled class structure limited the ability to model shared infrastructure and battery packs. The cost structures did not reflect strategic procurement differences, and the lifecycle assessment focused solely on CO_2-equivalent emissions, excluding other impact categories such as toxicity or resource depletion. These limitations suggest that future extensions should integrate infrastructure co-optimisation, procurement variation, and broader environmental metrics to fully capture the system-level implications of fleet renewal.

This research contributes a replicable, stakeholder-aligned methodology for sustainable fleet transition planning. It provides the Port of Rotterdam with a transparent and data-driven tool to align its environmental commitments with long-term operational and financial viability, providing critical insights for fleet operators pursuing low-emission transitions.

Contents

List of Figures

List of Tables

Nomenclature

Abbreviations

Abbreviation	Definition
Analytic Hierarchy Process	AHP
Analytic Network Process	ANP
Approximate Dynamic Programming	ADP
Battery Swapping Module	BSM
Blast Oven Furnace	BOF
Capital Expenditure	CAPEX
Carbon dioxide	CO2
Cradle to Gate	CTG
Date Envelopment Analysis	DEA
Decision-Making Unit	DMU
Direct Reduced Iron	DRI
Dynamic Programming	DP
Electric Arc Furnace	EAF
Elimination Et Choix Traduisant la Reality	ELECTRE
End-of-Life	EOL
European Union	EU
Genetic Algorithm	GA
Grave to Cradle	GTC
Greenhouse Gas	GHG
Groene Maze	GM
Hydrotreated Vegetable Oil	HVO
Incident Response Vessel	IRV
Integer Linear Programming	ILP
International Organization for Standardization	ISO
International Maritime Organization	IMO
Life Cycle Assessment	LCA
Life Cycle Impact	LCI
Linear Programming	LiP
Marine Diesel Oil	MDO
Maritime Fleet Size and Mix Problem	MFSMP
Maritime Fleet Renewal Problem	MFRP
Mixed Integer Linear Programming	MILP
Multi-Criteria Decision Analysis	MCDA
Multi-Objective Optimisation	MOO
Net Present Value	NPV
Nieuwe Maze	NM
Non-dominated Sorting Genetic Algorithm	NSGA
Operating and Maintenance	O&M
Operational Expenditure	OPEX
Particulate Matter	PM
Patrol Vessel	PV
Port of Rotterdam	PoR
Probabilistic Dynamic Programming	PDP
Preference Ranking Organisation Method for Enrichment of Evaluations	PROMETHEE
Rigid Hull Inflatable Boat	RHIB

Abbreviation	Definition
Shorepower	SP
Stochastic Programming	StP
Strength Pareto Evolutionary Algorithm 2	SPEA2
Surveyor Vessel	SV
Sustainable Development Goal	SDG
Small Patrol Vessel	sPV
Tank-to-Wake	TTW
Technique for Order of Preference by Similarity to Ideal Solution	TOPSIS
Total Cost of Ownership	TCO
United Nations	UN
Well-to-Tank	WTT
Well-to-Wake	WTW

1 Introduction

The maritime industry faces increasing pressure to reduce greenhouse gas (GHG) emissions in accordance with international climate objectives and regional policy instruments. This is driving fleet operators to transition toward more sustainable fleets. One such operator is the Port of Rotterdam (PoR), which must navigate complex trade-offs between decarbonisation, local pollution (LP) reduction, cost efficiency, and operational continuity.

This thesis investigates how fleet operators can gain strategic insight into multidimensional fleet renewal through a case study conducted at the PoR. This chapter introduces the broader context of the problem, outlines the research objective and questions, defines the scope, and provides an overview of the thesis structure.

1.1. Background and motivation

The impact of GHG emissions on climate change is undeniable (Stocker et al., 2014), and the devastating consequences of global warming are becoming increasingly evident (Intergovernmental Panel on Climate Change, 2022). This drives the urgency to reduce polluting emissions globally, in all industries. In 2015, the United Nations (UN) formulated the Sustainable Development Goals (SDGs), as a blueprint for a peaceful and prosperous future (United Nations, 2015). Goal 13 aims to take action against climate change and mitigate its impact. To do so, 107 countries have adopted net zero pledges for their GHG emissions and more than 9000 companies are part of *The Race to Zero*, to achieve an emission reduction of more than 50% by 2030 (United Nations, 2024). Furthermore, different regulatory bodies have developed strategies, such as the International Maritime Organization (IMO), which aims to reduce CO_2 emissions by 40% in 2030, compared to 2008 (International Maritime Organization, 2023). In addition, it has approved net zero regulations for global shipping, pricing emissions above a certain threshold, of which the proceeds are used to support innovation on low-emission shipping (International Maritime Organization, 2025). The European Union (EU) strives to be the first climate-neutral continent, for which it has developed *The Green Deal* (European Commision, 2019), implemented in legislation with *Fit for 55* (European Commision, 2023).

To be able to reach these emission reduction goals, many changes are needed within society. The distribution of GHG emissions in the EU showed that in 2019, 25.8% of the CO_2 emissions were emitted by the transport sector, which predominantly uses fossil fuels as an energy carrier (Eurostat, 2019). For the maritime sector, an investigation into the configuration of the power plant onboard ships worldwide by Eirik Ovrum et al. (2024) showed that 98% were still sailing on conventional fuels in 2024. With the average age of the international fleet being 12.6 years in 2023 (Clarksons, 2024), a large part of the current fleet will either need to be replaced before they reach their technical end-of-life (EOL), or undergo an expensive refit to be able to achieve the targeted emission reductions on time. The conventional drivetrain systems will need to be replaced by systems using non-conventional energy carriers or include emission capturing technologies to fulfil the pledged reductions. These technologies remain relatively unproven and include additional technical and financial challenges, leading to a difficult situation where fleet owners must navigate the trade-off between fulfilling sustainability pledges and ensuring that the fleet can operate economically viable. In this transition, battery-electric vessels are gaining increasing interest, resulting in a growing need for shore power (SP) infrastructure to enable regular battery charging. In operational contexts where continuous service is required and charging time must be minimised, battery swapping modules (BSMs) offer a practical solution by allowing rapid replacement of depleted battery packs.

Beyond emissions produced during vessel operation, a significant portion of the environmental impact of the maritime sector is derived from the production and salvaging of vessels. These upstream and downstream phases can contribute substantially to the total environmental footprint of a vessel. Material choices play a key role in this, as the production of metals such as steel and aluminium is energy intensive. There is growing interest in the use of scrap steel and recycled aluminium, reflecting broader efforts to reduce embedded emissions within industrial supply chains. This extended lifecycle perspective aligns with Goal 12 of the SDGs, which promotes sustainable consumption and production patterns (United Nations, 2015). Moreover, technological innovation in manufacturing and recycling processes is expected to reduce the carbon intensity of materials over time. This raises the question whether all lifecycle emissions should be treated equally across time horizons or whether future emissions can be discounted in light of expected efficiency improvements.

In addition to the global environmental impact, there is also a significant local impact, with shipping contributing LP in the form of NO_x and particulate matter (PM) emissions. These pollutants are directly associated with adverse health effects (Aardenne et al., 2013; World Health Organization, 2016). Reducing local ship air pollution aligns with Goal 3 of the SDGs, ensuring healthy lives and strengthening well-being, while at the same time promoting sustainable cities and communities (Goal 11), highlighting the multifaceted nature and importance of transitioning to a sustainable fleet.

To support fleet owners in the decision making process for fleet renewal, the modelling of ideal fleet compositions has been an area of research for more than 50 years (Pantuso et al., 2014). Within the modelling of maritime fleets, two types of problem are distinguished. The Maritime Fleet Size and Mix Problem (MFSMP) consists of problems focused on determining the optimal fleet composition for a single time step. It typically involves strategic decisions related to the optimal fleet composition and the optimal deployment of individual vessels, in a static operational environment for a single time step. The Maritime Fleet Renewal Problem (MFRP) extends the MFSMP by introducing a multi-period planning horizon, where the fleet evolves dynamically over time in response to market changes, technological advancements, or policy changes. Instead of optimising the composition of the fleet for a single period, MFRP models consider the timing of acquisitions and retirements to maintain long-term profitability, while also fulfilling demand in the multiple single time steps.

The transition of the PoR fleet falls under the MFRP. The PoR wants to become more sustainable and has increased its efforts to reduce polluting emissions. Specifically, the company aims to reduce its emissions in scope 1 and 2 by 90% in 2030 and sail emissions-free in 2035, compared to the emissions of 2019. Approximately 70% of the emissions in scope 1 that year were due to the fleet of the company (Port of Rotterdam, 2023). According to regional emissions data, PoR-operated vessels are among the 500 most polluting ships operating within city borders (Scholten et al., 2022). The fleet consists of multiple classes: Patrol vessels (PVs), small patrol vessels (sPVs), incident response vessels (IRVs), surveyors (SVs), rigid hull inflatable boats (RHIBs) and a presentation yacht called the *Nieuwe Maze* (NM), which will be replaced by a new yacht called the *Groene Maze* (GM), these two yachts fall under the NM class (VesselFinder, 2024). The PVs and IRVs are not completely homogeneous but consist of multiple series of vessels within the two classes. To be able to achieve the mentioned emission reduction goals in 2030 and 2035, a renewal programme was considered necessary for the current fleet. This programme was set up by the PoR and is currently working on determining suitable replacement vessels. By transitioning to a cleaner fleet, the PoR is actively working on SDGs 3, 11, 12 and 13.

To ensure that the fleet renewal programme is effective and aligned with the sustainability and operational goals of the company in a cost-effective manner, a deeper understanding of the key factors that influence the decision-making process of the fleet renewal problem is required. This includes identifying the relevant operational factors and evaluating the economic and environmental impact that influence the timing of the fleet renewal process. These challenges frame the objective of this research.

1.2. Objective

The objective of this research is to generate insight into how fleet operators, specifically the PoR, can strategically schedule the replacement of their vessels to reduce both the emissions of GHG and LP, while minimising the transition cost and maintaining operational capabilities.

To meet this objective, this study explores current MFRP practices, as well as multi-objective optimisation (MOO) and multi-criteria decision analysis (MCDA) methodologies, and sets up a decision support framework to generate the insight needed for fleet operators.

1.3. Research questions

To achieve the objective, the following main research question has been defined:

How can the Port of Rotterdam optimise the fleet renewal process to minimise the total polluting emissions and transition cost, while ensuring that operational capacity is maintained?

The main research question can be divided into four keywords: *optimise*, *emissions*, *costs*, and *process*. For each of these keywords, a sub-question has been defined.

For *process*, the following sub-question has been defined:

What are the key decision factors and fleet operator's interests that influence the timing of vessel replacement?

This explores the main aspects of the fleet renewal decision-making process, identified in part through the literature review and in part by interpreting the results of various scenarios that are simulated within the decision support framework.

For *costs*, the following research question has been defined:

What is the total cost of ownership (TCO) of the vessels in the fleet?

This covers capital expenditure (CAPEX) and operational expenditure (OPEX) over the lifetime of the vessel. Costs are determined through PoR stakeholder interviews and included within the framework based on common practices identified in the literature review. This results in the evaluation of the economic impact of fleet transition being included within the framework.

For *emissions*, the following research question has been defined:

What is the life cycle impact (LCI) of the vessels on the environment?

This considers the environmental impact of fleet and infrastructure emissions, including global CO_2-equivalent emissions across all lifecycle phases and local pollutants such as NO_x and PM during operation. The most suitable methods identified in literature will be incorporated for the emission assessment in the framework.

For the word *optimise*, the following question has been defined:

How can the economic, operational and sustainability factors be combined in a decision support framework?

This addresses the integration of multiple decision criteria into a suitable structure for the PoR case study. Existing fleet renewal modelling approaches are reviewed to select and adapt an appropriate structure.

By answering these sub-questions, the framework integrates relevant aspects of fleet renewal and is applied to simulate alternative economic and strategic environmental scenarios. Strategic pathways reflect different assumptions about fuel types, material use, and production practices, while economic scenarios vary key financial parameters. Together, these explore the sensitivity and robustness of fleet renewal strategies under varying long-term conditions.

1.4. Scope

This study focusses on providing strategic decision support for the fleet renewal programme at the PoR. Its primary objective is to generate insight into the driving factors of the fleet renewal process under multiple objectives, including minimising lifecycle emissions, ensuring cost efficiency, and maintaining operational continuity. The analysis excludes detailed vessel design, as the focus is on operational replacement rather than technical development. Workforce planning is also excluded, as it is assumed that crew size and composition remain constant across the old and new fleets. Replacement vessels are assumed to fall within existing operational class categories and serve functions equivalent to those of their predecessors.

The analysis adopts a strategic horizon of 25 years, corresponding to the expected life expectancy of the vessel, beginning in 2026. Within this horizon, operational ageing and scheduled replacements are considered. Vessel usage intensity, degradation rates, and unexpected failure events are not modelled. These exclusions are intended to maintain a high-level strategic focus and avoid introducing short-term operational variability that is outside the scope of this research.

The configuration of the new fleet is based on the choices made by the PoR fleet renewal team. It includes a fixed number of vessels and a predetermined number of charging locations. In accordance with these choices, the new vessels are assumed to be battery-electric, with the exception of the RHIB, which is assumed to sail on hydrotreated vegetable oil (HVO), due to range and speed restrictions. The IRV, PV, and NM vessel classes are assumed to operate using BSMs, to enable continuous service. The new vessels for the sPV and SV classes are assumed to charge overnight using SP infrastructure. These infrastructural and technological choices were made by the PoR and are treated as fixed inputs in the analysis. CAPEX and OPEX are included for the vessels, the batteries, and the supporting infrastructure. Fuel costs are considered separately from general OPEX to distinguish between actively operating vessels and those in reserve. The design and construction times for the vessels, battery systems, and infrastructure are excluded to maintain a narrow focus on strategic replacement decisions.

The environmental analysis is based on a life-cycle assessment (LCA) perspective. CO_2-equivalent emissions from the processing, production and dismantling stages of batteries, vessels, and infrastructure are included. The emissions resulting from the production of BSM cranes are taken into account, while the emissions associated with the preparatory work at the infrastructure sites are excluded. During the operational phase, NO_x and PM emissions are also considered to capture local air quality impacts. Other environmental impact categories such as toxicity, resource depletion, and noise pollution are excluded due to limited availability of reliable data and to avoid excessive uncertainty that could hinder the interpretation of the results. The environmental inputs assume the use of both new and recycled steel and aluminium, reflecting current industrial production processes. However, the broader concept of circularity, defined as systematic reuse, remanufacturing, or closed-loop material tracking, is not explicitly modelled. Environmental assessment is limited to emission-based indicators and does not account for resource recovery or EOL reuse scenarios.

To evaluate the implications of different strategic decisions, a series of scenario simulations is conducted. These include five strategic pathways, each representing different assumptions about material choices, fuel types, and production locations. Within each pathway, the CAPEX and OPEX values are kept constant to isolate the effects of strategic environmental decisions. One pathway also figures as the standard economic scenario, and two other economic scenarios are defined by varying the costs, the inflation, and depreciation rates, while the same strategic configuration is maintained. Finally, a separate scenario includes the depreciation of CO_2 emissions during the cradle-to-gate (CTG) and grave-to-cradle (GTC) phases. This scenario captures the effects of technological advancement and grid decarbonisation on the processing, production, and dismantling stages by discounting future CTG and GTC emissions over time. All scenarios are evaluated using the proposed decision support framework to determine their influence on the optimal fleet replacement strategy.

1.5. Thesis outline

The thesis is structured into two main parts. Part I consists of the review of the existing literature, to establish the conceptual and methodological basis for the decision support framework. First, the methodology of the literature review is discussed, as well as the identified literature gaps and a detailed description of the focus area of the performed literature review in chapter 2. In chapter 3 the relevant factors of the fleet renewal problem in current studies are discussed in depth, divided into the operational aspect of fleet renewal, the sustainability aspect, and the economic aspect, to support answering the first three sub-questions. In chapter 4 different fleet renewal modelling methodologies and decision support frameworks of current MFRP research are explored, to support answering the fourth sub-question and determine a suitable framework for the case study at the PoR. In conclusion Part I, chapter 5 describes the proposed framework and how the various aspects of the MFRP are integrated. This sets the foundation for the implementation described in the next part.

Part II, details the architecture of the decision support framework in chapter 6. The input parameters of the framework are established in chapter 7. The results of the application of the framework for the case study at the PoR are provided in chapter 8. Chapter 9 interprets the results and addresses the sub-questions, along with the framework's limitations and recommendations for future studies. The thesis concludes by answering the main research question in chapter 10.

Part I

Literature review

2

Literature review methodology

This chapter outlines the methodology used to conduct a systematic and comprehensive review of existing research on fleet renewal. The objective is to identify and classify relevant studies that address the various dimensions of the MFRP, as well as the modelling methodologies used to support decision making. The chapter explains the structured approach adopted to collect and analyse the literature, highlights the depth and focus of existing research, and identifies the research gaps this thesis aims to address.

The literature collection and screening strategy is presented in section 2.1, followed by the classification of studies and the identification of research gaps in section 2.2. The structure of the subsequent literature review chapters is described in section 2.3. The chapter concludes with a brief summary in section 2.4.

2.1. Approach

SCOPUS was selected as the primary database for this literature review due to its extensive repository of scientific publications. Known for its comprehensive inclusion of peer-reviewed journals, conference proceedings, and books, SCOPUS serves as an excellent resource for collecting a diverse array of scientific research on fleet renewal. Its sophisticated search functionalities facilitated a comprehensive and efficient search (TU Delft, n.d.). Although Web of Science and Google Scholar were considered, they were excluded because of a significant overlap with SCOPUS. To find relevant studies that are not available on SCOPUS, a backward check was performed on the articles that were deemed useful from SCOPUS.

Seven distinct queries were formulated, each targeting a specific aspect of the MFRP. Query A focused on modelling and decision-making processes. Query B addressed economic considerations. Query C captured stakeholder-related and non-economic factors. Queries D and E aimed to identify studies in which operational capacity was required to be maintained during the renewal process. Query F addressed maintenance considerations, while Query G incorporated the sustainability dimension.

The term *fleet replacement* was used as a synonym for *fleet renewal* in all queries. Furthermore, for Query A, *decision-making* was used as a synonym for *decision analysis*.

The SCOPUS queries are summarised in Table 2.1.

Query	Search term
A	”fleet renewal” AND multi-objective OR multi-criteria OR ”decision analysis”
B	”fleet renewal” AND economic* OR cost*
C	”fleet renewal” AND stakeholder* OR factor*
D	”fleet renewal” AND availability
E	”fleet renewal” AND operational AND capacity OR requirement
F	”fleet renewal” AND maintenance
G	”fleet renewal” AND emission*

Table 2.1: Used search terms in SCOPUS.

The resulting papers from the queries entered into SCOPUS were screened in three steps. First, duplicate results were removed from the total results of subsequent queries. This was followed by the initial screening. In the initial screening, the titles and abstracts of the articles were reviewed to exclude non-relevant studies. Articles that were not focused on fleet renewal or were not scientific in nature were removed at this stage.

Secondary screening involved a more detailed examination of the abstract and a quick scan of the full text of the articles. During this stage, the relevance of the articles was assessed to the key aspects of fleet renewal, such as decision analysis processes, economic factors, sustainability considerations, and operational factors. Articles that focused solely on new technology aspects without addressing the broader renewal process were excluded. This screening process ensured that only the most relevant studies were included in the literature review.

In addition to the articles identified through the initial search queries, 19 additional articles were discovered by backward checking the articles that passed the secondary screening. The distribution of the articles found with each query in the screening process is given in Table 2.2.

Query	Total results	New results	After title & abstract screening	Final selection
A	36	36	20	8
B	187	161	48	17
C	80	35	10	3
D	16	5	1	0
E	8	2	0	0
F	38	5	0	0
G	139	43	4	2
Added	-	-	-	19
Total	**504**	**287**	**83**	**49**

Table 2.2: Search hit distribution.

The 49 resulting articles were categorised to provide a structured overview of the current research landscape in fleet renewal. This categorisation is based on the main focus of each article, allowing a systematic analysis of key themes and trends in the literature. The 49 articles are shown in Appendix A, as well as the source article of the 19 articles that were found by backward checking. The final set of 49 articles provides a structured and broad foundation on which this literature review is built to identify methodological directions and research gaps.

2.2. Research gap

The literature was categorised into four distinct categories based on the main focus of the research: *Decision analysis*, *economic*, *sustainability*, and *operational capacity*. This categorisation, performed during the secondary screening phase, enabled a systematic review of the current research landscape and revealed the depth of exploration in each domain. The results of this classification are presented in Table 2.3.

Category	Description	Number of papers
Decision analysis	All articles that provide decision analysis support by comparative analysis or objective optimisation	39
Economic	All articles that include an economic objective or analysis for fleet renewal	37
Sustainability	All articles that include sustainability objectives or impact analysis	13
Operational	All articles that focus on maintaining or improving the operational availability during the fleet renewal process	4

Table 2.3: Research categories.

The findings show that 25 of the articles focused solely on economic evaluations as the main objective or optimisation criteria, using either the TCO or the net present value (NPV) methodology. In 13 of these articles, operational aspects were included by including a forecasted demand or a one-on-one replacement strategy for the vessels. In seven articles, environmental constraints were also used to restrict the solution space, either by using emission limits (Loennechen et al., 2024; Martin et al., 2024; Patricksson et al., 2015; Sønnervik et al., 2024; Zhao et al., 2021, 2024) or determining a yearly electrification target (Pelletier et al., 2019). Only three studies also included maintenance downtime to determine optimal strategies, excluding vessels undergoing maintenance from the operational fleet that needed to meet a predetermined demand (Meng & Wang, 2010; Meng et al., 2015; Patricksson et al., 2015).

Coppola et al. (2023), Du and Kommalapati (2021), and Giordano et al. (2018) combined economic and environmental evaluations as the main objectives to optimise, all three using the LCA methodology for CO_2 emissions. For the economic evaluation, the former used NPV and the latter two used TCO. The studies evaluated different strategies on both objectives and compared them using MCDA. Castillo and Álvarez (2023) used the classification of emission scopes, combined with TCO in a MOO framework. Aiello et al. (2024) and Moreno Sader et al. (2025) both used MCDA to compare the environmental impact, assessed with the well-to-wake (WTW) framework, together with the economic impact, assessed using TCO. These studies did not adequately account for operational factors such as fleet availability, maintenance schedules, or infrastructure capacity during the transition phases.

For Ali et al. (2023), Fee et al. (2019), and Turan et al. (2020, 2022), the main objective was operational availability, accounting for unavailability due to maintenance downtime, as well as the availability of the supporting infrastructure and the sufficient staff to crew the ships. The final article used MOO to evaluate various fleet combinations. The first three articles used simulation to simulate the workforce transition, with the fleet replacement optimised using MOO and evaluated with MCDA.

The categorisation of the reviewed literature reveals several notable research gaps. First, LP emissions are almost entirely absent as an optimisation criterion, despite its relevance for direct health and urban impact. Second, while some operational constraints, such as maintenance downtime or crew availability, were considered in a handful of studies, they are rarely integrated into models that also account for environmental performance. Likewise, environmental factors are often missing from operationally focused models. LCA focused solely on CO_2-equivalent emissions, not including other impact categories.

This thesis seeks to address these gaps by developing an integrated decision support framework that incorporates economic, environmental, and operational dimensions. Inspired by the boundary definitions of the existing literature, new variables are introduced to expand the relevance and realism of the framework. These include explicit modelling of LP emissions, specifically NO_x and PM emissions, maintenance-related unavailability, and supporting infrastructure. In addition, the framework introduces future-orientated considerations such as the discounting of CO_2 emissions to reflect expected technological improvements in material production. The structure of this framework and the methodological steps leading to its development are described in the following section.

2.3. Structure

To fill the research gap, the operational, economic, and environmental aspects of fleet renewal are first reviewed separately in chapter 3. This chapter is focussed on providing insights for answering the first three sub-questions. This is done by first going over how operational aspects were included in previous research, both as a key variable and as a way to limit the solution space. This is followed by a detailed investigation on how the environmental impact is quantified in previous studies and incorporated together with economic considerations. After that, the different economic frameworks to assess the economic impact of fleet renewal are discussed. Finally, the findings of this chapter are used to determine the way forward to include these factors in the case study.

This is followed by chapter 4, focussing on the fourth sub-question, by evaluating the different modelling methodologies used for the MFRP, particularly the integration of multiple objectives or criteria. This is done by first detailing the historical development of (maritime) fleet renewal modelling, followed by the discussion of the different methods that were used to combine multiple objectives or criteria in previous studies. By providing a comparative analysis of these methods, the appropriate structure of the framework is determined.

Finally, chapter 5 summarises the key findings and limitations identified in the literature.

2.4. Conclusion

The literature review methodology provides a systematic and structured exploration of existing research in the domain of fleet renewal. Through a targeted SCOPUS-based search strategy and a classification between the different focusses of the articles, the review identifies the main foundations of the literature on fleet renewal, as well as aspects where research is still lacking.

In particular, while many studies have been optimised for economic return, there is a visible research gap in incorporating operational factors, especially outside of the military domain, together with sustainability objectives in a unified framework. This literature gap, especially the lack of integrated approaches that combine economic objectives with environmental and operational dimensions, underscores the novelty and relevance of this thesis. By addressing this gap, the proposed research aims to provide a more comprehensive and balanced approach to modern fleet renewal decision making.

The findings of this literature review lay the groundwork for future research and practical applications in fleet renewal. By filling the identified research gap, this study has the potential to significantly advance the field, offering new insights and methodologies that can be applied to real-world fleet renewal strategies.

3

Fleet renewal

A successful fleet renewal programme requires careful consideration of multiple interdependent factors. First, it is essential that the fleet is able to meet its functional objectives throughout the transition period, which can be to transport cargo, patrol, provide emergency response, or perform other specific duties. section 3.1 details how current research ensures that fleet capacity is sufficient to perform required tasks, for the duration of the transition period.

In addition, sustainability commitments and regulations have become a major factor in fleet renewal. The environmental impact of both old and new fleets must be carefully considered, along with compliance with current and future regulations. The different methods used in current studies to ensure compliance with emission regulation and account for their negative impact are explored in section 3.2. Not only is the polluting effect of the fleet during its operational usage of relevance for the environment but also the impact of the construction and decommissioning of the fleet.

Lastly, fleet renewal is a costly process that requires significant investments from the fleet operator. Therefore, an economic analysis is necessary to recognise the associated expenses. The ways in which these analyses have been performed in previous studies are detailed in section 3.3.

A summary of the findings of this chapter and the way forward to include these three different aspects within the framework for the PoR are discussed in section 3.4.

3.1. Operational factors

To ensure that the fleet can meet its objective, several operational factors come into play. The various operational factors that have been employed in the researched articles have been identified and divided into four main categories, which are discussed individually. First, *operational capacity* is discussed, detailing how previous research has ensured that the fleet has enough capacity to perform its task. This is followed by *infrastructure*, which focusses on the how the required infrastructure changes are included, inherent in switching to new energy carriers, so that fleet operators can support the operation of new vessels. *Maintenance* includes the various manners in which fleet maintenance requirements have been taken into account, which is necessary if fleet operators want to perform their tasks for any significant amount of time. Finally, *crew* details how previous studies simulated the need for manned vessels.

3.1.1. Operational capacity

Operational capacity is a critical aspect of fleet renewal, ensuring that the fleet has the capacity to perform its tasks. Various methods are used to include and optimise operational capacity, each with its own benefits and disadvantages.

The most convenient way to maintain the same level of operational capacity during fleet renewal is to use a one-on-one replacement strategy for the vessels, as demonstrated by Sadeghpour et al. (2019). This approach involves replacing each old vessel one-on-one with a new vessel, ensuring that the size of the fleet remains constant throughout the planning horizon. The simplicity of this method makes it easy to implement, as it avoids the need for adjustments to the size or composition of the fleet and allows the decision maker to focus solely on determining the best moment of replacement for the individual vessels. However, this strategy lacks the flexibility to adapt to changing demand or the new vessels having a different operational capacity than the old vessels. This makes the strategy primarily suitable for fleet owners that know with a high degree of certainty that their future fleet composition should remain exactly the same and changes in the vessel design do not significantly influence the operational abilities with regard to the fleet's task.

One common method to include different future scenarios is the use of demand satisfaction constraints, which ensure that the total capacity of the fleet can meet the predetermined demand for every time period. This method is typically expressed through mathematical models that include constraints to ensure that demand is satisfied (Castillo Campo & Álvarez Fernández, 2023; Pantuso et al., 2014; Winkelmann et al., 2024). The benefits of this approach include its simplicity and reliability in ensuring that the fleet can meet demand. In addition, future changes in demand are accounted for by specifying the demand per time step. However, unexpected changes in demand can greatly disrupt the accuracy of the model, making it important to take into account the uncertainty. This method is most applicable to fleet owners who want to model their fleet composition based on the future outlook of demand.

Another method is to employ route-based demand constraints, where vessels are linked to certain routes to model liner fleets. Constraints are used to ensure that enough vessels are assigned to specific routes to meet the specified demand for that route (Meng & Wang, 2010; Meng et al., 2015). This approach optimises the use of vessels by allocating them to specific routes, which allows the fleet owner to specify the types of vessel that sail on the different routes. This allows for the incorporation of restrictions on vessel requirements on certain routes, such as Zhao et al. (2024) employed on the power requirements of vessels in a certain operating area and Zhao et al. (2021) on sulphur emission requirements, to model low-emission zones. This grants the fleet owner the ability to optimise the fleet renewal process for multiple routes simultaneously, as well as to optimise the usage of non-homogenous vessels in the fleet. However, this comes with the need to accurately predict not only the total demand but also specify the demand over the different routes. Also, the non-homogenous vessels may not be able to sail other routes, which can lead to a fleet composition that is optimised for a certain scenario but would perform poorly in other scenarios. This methodology is mainly suited to liner fleet operators, where demand on different routes can be predicted with a high degree of certainty. It is also suited to be applied to public transit buses, as Pelletier et al. (2019) has done.

Penalty functions introduce a penalty cost for solutions that fall below a minimum availability threshold, encouraging the operational capacity to be above that threshold. Fee et al. (2019) and Turan et al. (2022) used penalty functions to encourage high availability for military vessels. The main advantage is that the importance of operational capacity can be balanced with other penalty functions that contain factors such as the cost or duration of replacement. By choosing the different penalty costs, the preference of the fleet owner can be included to optimise the fleet renewal to the owners' wishes. Because this approach requires the assignment of subjective penalty weights, its applicability depends on the decision maker's ability to quantify trade-offs. This method is mostly useful for fleet owners that want to make a balanced trade-off between different objectives based on their own preferences.

Availability maximisation defines the availability of the fleet as the main objective and aims to maximise it over time. Turan et al. (2020) applied this to a military fleet to account for the different maintenance schedules and the limited resources. Ali et al. (2023) employed the same method, but added an additional step of dividing the availability with the planning horizon, resulting in a deployment score. The benefit of this method is that the average availability is maximised, making effective use of the current asset pool and their various states during the fleet renewal process. However, this does not make it the most cost-effective option, since it does not account for excess capacity. Furthermore, while the total availability might be maximised, that does not necessarily mean that there is sufficient capacity at all single timesteps. For certain fleet owners, a certain minimum capacity might be more important. This method is most applicable to fleet owners with vessels in multiple stages of their life and limited resources, aiming to have maximum availability, such as navies. It might be necessary to combine this with additional limits on a minimum availability to reach a desirable solution for every time step.

3.1.2. Infrastructure

With the transition of vessels to new non-fossil energy carriers, the energy infrastructure must be adjusted to support fleet renewal. This adjustment is critical to ensure that the new fleet can operate efficiently and carry out its tasks. Various methods are used in the literature, each with its own benefits and disadvantages.

One of the key considerations in the adjustment of the supporting infrastructure is the different technical capabilities. Aiello et al. (2024) included the range and refuelling time as performance criteria for an evaluation of three different technologies for fleet renewal, namely conventional, electric, and hydrogen vehicles. Besides the range and refuel time, the cost and emissions are also included as performance criteria for the decision analysis. This approach ensures that the infrastructure limitations of various technologies are included in the considerations for different technologies. This method is mainly of importance for fleet owners that want to compare different energy carrier technologies and their subsequent requirements with the supporting infrastructure.

The acquisition cost of the necessary infrastructure is another critical consideration in evaluating the total cost of fleet renewal. Coppola et al. (2023), Moreno Sader et al. (2025), and Winkelmann et al. (2024) accounted for the acquisition cost of the necessary infrastructure in their studies. This provides additional information on the total cost of the fleet renewal programme for the vessels and the necessary infrastructure. For fleet owners who provide their own infrastructure for their fleet, these costs are necessary to include to assess the cost of switching to another energy carrier. In addition to acquisition costs, the operating and maintenance (O&M) costs of the infrastructure can be included to provide more accurate overviews of the investments required over the planning horizon. Islam and Lownes (2019) employed this for a public transit bus fleet renewal programme. To fully account for infrastructure costs, Alp et al. (2022), Castillo and Álvarez (2023), and Castillo Campo and Álvarez Fernández (2023) included it in the TCO analysis, where besides the acquisition and O&M costs, also the residual value is also included at the end of the planning horizon. This comprehensive view spans the entire asset lifecycle, but requires robust cost projections and salvage value estimates, which may be uncertain for emerging technologies and long-term planning horizons. This highlights the need to consider not only the vessels themselves, but also the surrounding system in which they operate.

3.1.3. Maintenance

The maintenance of vessels is vitally important to ensure viable operations during their lifetime. However, it is difficult to predict the specific amount of maintenance that vessels need, as certain components will fail unexpectedly. Maintenance costs are included in the O&M cost of the vessel. These costs are often based on historical data, the technology employed, the acquisition cost, and the type of vessel. This amount scales with time, making it more expensive to maintain older vessels than newer ones. This creates a trade-off between when it is more economically beneficial to purchase a new vessel with a high acquisition cost but lower O&M costs. This approach has been used by Castillo Campo and Álvarez Fernández (2023), Coppola et al. (2023), Pantuso et al. (2015), Pelletier et al. (2019), and Sønnervik et al. (2024). Often, a maximum service life is determined, with vessels needing to be removed from the fleet at the latest at the EOL. After the EOL, life-extending maintenance is required, increasing the associated cost and introducing additional uncertainty in the total amount of O&M cost.

Although scaling the O&M cost with age is a simple way to make a base assumption about the O&M cost and accounting for the ageing effects, it does not account for the actual usage of the asset. Castillo and Álvarez (2023) and Winkelmann et al. (2024) determined that maintenance costs depend on the cumulative use of the asset. This ensures that the state of the asset is taken into account more accurately. However, it also requires tracking of the usage of different assets over time, which can bring additional difficulties, especially for large fleet owners. Aiello et al. (2024) combined both age and mileage as factors for the growing O&M costs, providing a more complete view of maintenance needs.

Meng and Wang (2010), Meng et al. (2015), and Zhao et al. (2021) went further than accounting for the cost, including the unavailability of the vessels while maintenance is performed on them. This was done by determining an age-dependent maintenance time that the vessels were required to undergo periodically. This causes the size of the fleet to grow to still have the operational capacity available to fulfil its at all times. This is a more realistic method, since SOLAS requirements state that merchant vessels must undergo a comprehensive study of the hull in a dry dock twice in a five-year period (International Maritime Organization, 1974).

Fee et al. (2019), Patricksson et al. (2015), Sønnervik et al. (2024), and Turan et al. (2020) not only accounted for maintenance cost and downtime, but also for maintenance facility workload, assuming only a limited number of ships could be maintained at the facility at any time, ensuring that fleet maintenance is spread over the planning horizon. This is most relevant for fleet owners that perform their own maintenance.

3.1.4. Crew

Vessels (still) need to be manned, as this crew is needed to operate the vessels. The cost associated with the crewing of the vessels is included in the O&M costs. Several studies have separately included the importance of crew management in maintaining operational efficiency. Fee et al. (2019) penalised the number of crew outside a predefined range, emphasising the need to maintain an optimal crew size to ensure that the fleet composition matches the crew composition. Turan et al. (2020) included the availability of the crew as a constraint, ensuring that the fleet renewal process can be sustained by the available crew. Turan et al. (2022) introduced an additional cost for not having enough personnel, ensuring that the crew is maintained at optimal levels to reduce the risk of operational disruption.

3.2. Sustainability

Since climate change and related regulations are the main drivers of the transition to cleaner technologies during the fleet renewal process, environmental factors are often included in the latest research. There are multiple ways to assess the impact of the environmental impact. In this section, first, the ways to monetise the environmental impact are discussed, followed by an individual discussion of the different quantification methods of emissions that have been employed in previous research.

3.2.1. External cost

External costs are costs that fall outside of the market price of the goods or services that cause the cost and are imposed on a third party (CE Delft, 2019). There are several types of external costs, with the cost of climate change the most frequently included. The exact cost may be difficult to quantify since it is difficult to determine the cost of emissions. To approximate the external cost of climate change, studies often use the carbon price in emission trading systems as a proxy for market valuation, ranging between 50-100 € per ton of equivalent CO2 over the last three years for the emission trading system of the EU (Trading Economics, 2025). The cost of removing a ton of CO2 using direct air capture is estimated to be between 100 - 1300 $ per ton of CO2 equivalent (Young et al., 2023). These wide ranges in estimates illustrate the uncertainty fleet operators face when attempting to internalise climate costs into decision making.

Alp et al. (2022) and Winkelmann et al. (2024) accounted for the external cost of global warming, caused by the emissions of GHG by the fleet. These costs are penalised by a carbon tax; this internalises the external costs and combines them with the market price, providing incentives for companies to move to cleaner alternatives in order to minimise the costs. Coppola et al. (2023) and Zhou et al. (2023) included the external cost due to air pollution. Since the air pollution causes negative health effects for the local population. Islam and Lownes (2019) included the external social cost, which is composed of the external cost of global warming, air pollution, and noise pollution.

3.2.2. Scope 1, 2, and 3 emissions

For an company to internalise the external cost of emissions, a company must first know what emissions occur. These emissions can be divided into three scopes. Scope 1 covers direct GHG emissions by assets controlled or owned by the company, such as the combustion of fuel by company cars and vessels. Scope 2 covers indirect GHG emissions associated with the purchase of electricity, steam, cooling, and heating for the use of the company. Scope 3 covers indirect GHG emissions due to activities from assets not owned by the company but affected by its value chain. Figure 3.1 provides a structured overview of scope 1, 2, and 3 emissions across the value chain of a company, helping to visualise the direct and indirect impacts of fleet operations. Scope 3 includes 16 distinct categories. Their relevance depends on the structure and activities of the company's value chain.

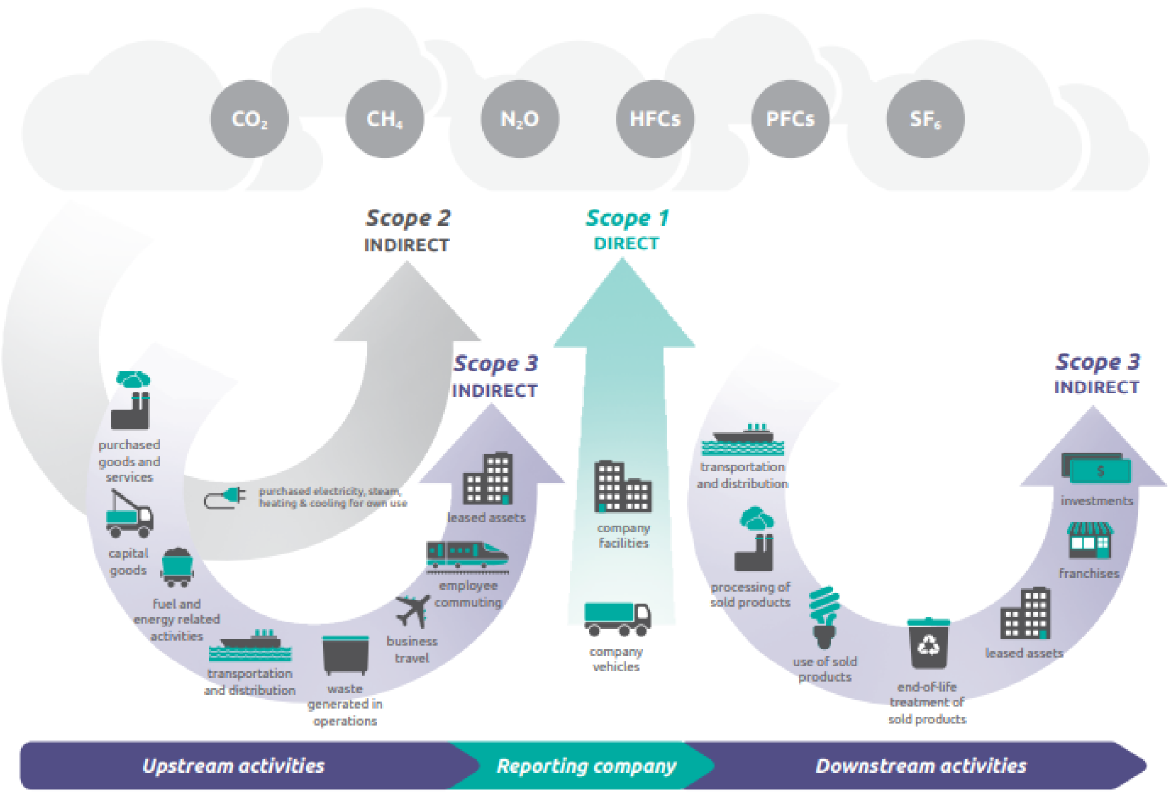


Figure 3.1: Overview GHG Protocol scopes and emissions (GHG Protocol, 2013).

Alp et al. (2022), combined an electrification target for the fleet with a carbon tax on emissions in scope 1, accounting for the fuel consumption of the fleet. Loennechen et al. (2024) and Zhao et al. (2024) included emission limits for scope 1, to model regulated emission limits, but did not include any external cost. Filák et al. (2021) and Sønnervik et al. (2024) used the scope 1 emission limits, but added a carbon tax, providing an additional economic incentive to switch to cleaner fuels. Winkelmann et al. (2024) used both emission limits and carbon pricing in all three scopes, accounting for fleet emissions throughout the value chain. Castillo and Álvarez (2023) accounted for the emissions in the three scopes and set the model objective to minimise the sum of the emissions in the three scopes.

The main benefit of measuring the emissions in the different scopes is to generate insight for a company, where the emissions in their value chain take place and whether they fall under their direct control or if they should work with upstream and downstream partners to mitigate them, but to gain more insight into the composition of the direct and indirect emissions, other methods are needed. These include LCA and WTW assessments, both of which provide greater resolution with respect to emission sources and timing.

3.2.3. Well-to-wake

The well-to-wake or well-to-wheel framework allows the fleet operator to gain insight into emissions with respect to fuel consumption. It consists of two segments: Well-to-tank (WTT) and tank-to-wake (TTW). The WTT accounts for the GHG emissions emitted to produce, transport, and refine a primary fuel to the fuel bunkered on board. For conventional vessels, this consists of the crude oil extraction, refinement, and transportation of the end-product, which is bunkered on board. For electric vessels, it consists of the production and transportation of the required electricity (WWF, 2017). The TTW phase accounts for emissions released during the actual use of the fuel to propel the vessel. For an electric vessel, these are zero, but for conventional vessels, these are the emissions coming out of the exhaust.

The emissions coming from the exhaust of a vessel fall under scope 1 emissions, while the emissions from the use of electricity fall under scope 2. The emissions from the extraction, refinement and transport of the energy carrier fall under the scope 3, category 3, *fuel and energy related activities*. Figure 3.2 shows the WTW emission pathway, distinguishing between fuel sourcing (WTT) and fuel usage (TTW).

WTW assessments isolate emissions related to the sourcing and use of fuels, providing fleet operators with clarity in comparing energy carriers.

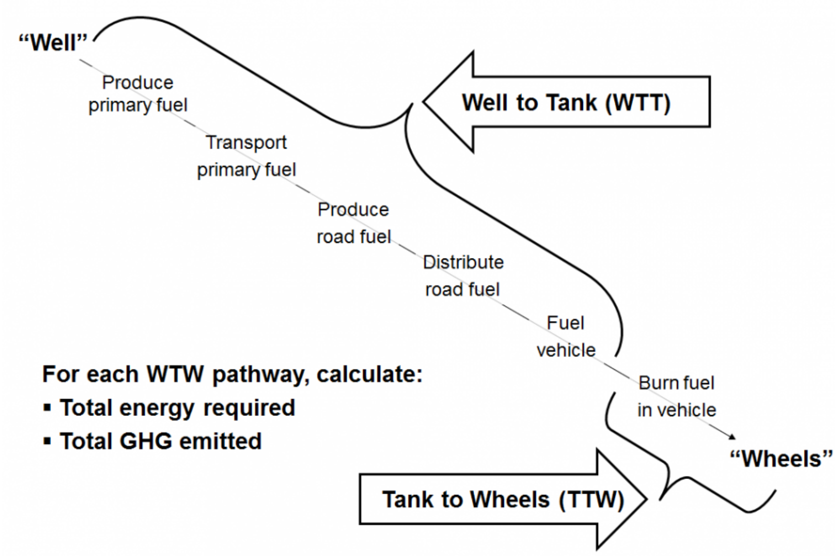


Figure 3.2: Well-to-wheel emissions (European Commision, 2016).

The WTW emissions have been used by Aiello et al. (2024) as an objective to minimise and by Zhou et al. (2023) to account for the external cost of global warming and air pollution.

3.2.4. Life Cycle Assessment

LCA evaluates the impact of a product throughout its life cycle. It consists of five stages: extraction of raw materials, production and processing of the product, transportation of the product, use of the product, and finally disposal of the product (Liu et al., 2024). Assessment can be performed at multiple levels, from *cradle-to-gate*, focused solely on the production emissions, to *cradle-to-cradle*, which includes production, transportation, and use phase emissions, as well as emissions from the conversion of waste products back into useful resources that are used in the manufacturing process, creating a closed-loop resource cycle. Figure 3.3 illustrates the main lifecycle stages, offering a complete view of potential emissions boundaries.

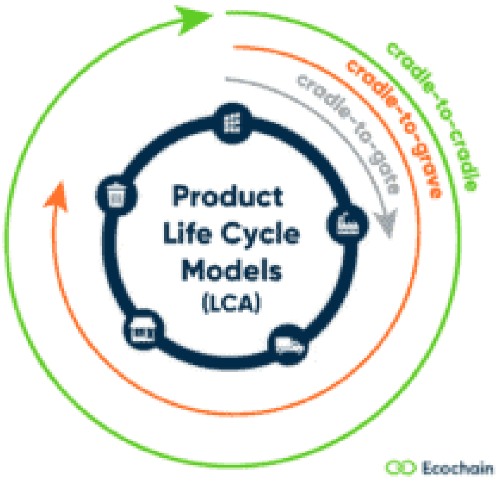


Figure 3.3: Life cycle assessment stages (Ecochain, 2025).

The LCA consists of two main stages. The first is *inventory analysis*, which captures data on material and energy flows throughout all phases of the product life cycle, including extraction, manufacturing, transportation, use, and end-of-life processing. This is followed by *impact assessment*, in which these flows are evaluated in up to 16 standardised environmental impact categories (European Commission, n.d.).

Desantes et al. (2020), Du and Kommalapati (2021), and Giordano et al. (2018) incorporated lifecycle emissions for vehicle fleets. They used the GREET tool, developed by the US Department of Energy, which estimates vehicle emissions per mile. Coppola et al. (2023) included both lifecycle GHG emissions and direct local polluting emissions as criteria to analyse different fleet renewal options. To determine the emissions per kilometre, values were used from Nordelöf et al. (2019) and complemented by reduction factors for battery recycling, which was estimated to save 21 kg of CO2 per kWh of the recycled battery pack.

For fleet renewal decisions, the most critical impact categories include climate change, driven by the global warming potential of GHG emissions. The formation of PM, due to its adverse effects on human health. The use of resources, which accounts for the depletion of finite natural materials. Although advanced technologies can reduce environmental impacts in one category, they can simultaneously increase burdens in another or shift impacts geographically. For example, electric powertrains do not produce direct fuel-related emissions during operation, thus improving local air quality. However, the electricity used may originate from fossil fuel-based sources, such as coal, resulting in upstream emissions in a different region. In addition, battery production relies on materials such as lithium, cobalt, and nickel, whose extraction and processing can result in substantial local emissions and raise ethical concerns about poor labour conditions and environmental degradation (Nature, 2021). This results in difficult ethical considerations, where trade-offs between different benefits and harmful consequences need to be made.

The LCA assessment is mainly useful to gain insight into the total emissions related to the use of a certain product, allowing for the comparison of different materials and recycle procedures, as well as emission benefits or penalties in emissions to extend the technical life of products.

3.3. Economic analysis

Economic viability remains a decisive factor in fleet renewal, and profitability is of vital importance for companies. There are two main methods that have been used in previous studies to evaluate the economic aspect: the TCO framework and the NPV methodology. The TCO framework is the most dominant, being used by 31 out of the 37 papers that make use of an economic evaluation, and the other papers mainly using the NPV methodology.

3.3.1. Total cost of ownership

The TCO framework is a comprehensive financial metric that captures the full cost profile of a vessel throughout its entire operational life. It encompasses CAPEX, associated with asset acquisition and disposal expenses and profits, and OPEX, encompassing the running costs of assets, such as crew wages, fuel cost, maintenance cost, and insurance cost. The residual value that would be gained if the asset is sold at that moment in time can be included in the CAPEX at the end of the planning horizon, to account for the value of the assets at the end of the timeline. External costs can also be included by accounting for the cost of emissions.

TCO is often used as a basis for comparing different fleet configurations or technologies in a cost-minimising framework. For example, Castillo Campo and Álvarez Fernández (2023) evaluated the TCO of three types of electric vans with conventional vans in an MOO model to determine the most economically optimal transition pathway under emission constraints. Similarly, Aiello et al. (2024) used TCO in an MCDA framework to assess trade-offs between cost, sustainability, and technical capacity for road transport.

In total, 14 articles used the TCO standard evaluation, while 17 articles discounted the TCO. This was done by discounting the future cost based on a set discount rate, to include the time value of money. This was done using various discount rates, with the minimum annual rate of 2% being used by Castillo Campo and Álvarez Fernández (2023) and the maximum annual rate of 10% by Alp et al. (2022) and Giordano et al. (2018), highlighting the uncertainty of what should be used as the discount rate.

By discounting the cost, the time value of money is included in the total cost. This also provides the foundation for the NPV, which is discussed in the next section.

3.3.2. Net present value

NPV is a financial evaluation method that evaluates the profitability of an investment by discounting all future costs and revenues to their present value. The standard formulation of NPV aggregates annual net cash flows over a planning horizon, each adjusted by a chosen discount rate that reflects the cost of capital or the required rate of return.

In the context of maritime fleet optimisation, NPV is often used to determine the optimal replacement schedule for ageing vessels. A positive NPV indicates that the investment yields net financial benefits over its lifetime, while a negative NPV suggests that the strategy is unable to meet the required rate of return on the investment. Meng and Wang (2011) and Zhao et al. (2024) used the NPV to determine the optimal fleet renewal strategy, with the goal of maximising profits over time. Compared to TCO, NPV integrates the return on investment. However, future revenues might be difficult to predict, and not all vessels have a direct earning capacity, such as those at the PoR, where the vessels do not transport any cargo or people. If no revenues are generated, it results in the NPV having a value below zero.

3.4. Conclusion

This chapter has outlined the diverse nature of the fleet renewal problem, highlighting the operational, environmental, and economic drivers that shape long-term strategic decisions. Operational continuity is essential during the transition, which requires careful scheduling of vessel availability and accounting for maintenance downtime. Maintenance strategies and crew availability have been incorporated into fleet planning models, although primarily in military contexts, showing potential for broader adoption in civilian settings. Infrastructure constraints, especially those related to alternative energy carriers, introduce additional planning complexities and costs.

Environmental sustainability has increasingly shaped fleet renewal, with emissions classified in scopes for company-wide assessments, WTW assessments for fuel-related research, and LCA for product-related assessments. Although external costs may be internalised, the cost is difficult to quantify and can be very different from the market price of emissions. Certain technologies may change the locations of emissions or introduce additional consequences, such as pour labour conditions for the miners of the needed materials for batteries. This introduces ethical dilemmas where choices for one technology benefit the environment in one area at the cost of another.

From an economic perspective, the TCO provides a widely used and comprehensive metric that captures capital and operating expenses and can include environmental externalities such as carbon pricing. The NPV methodology is particularly suited for long-term financial planning, with assets that generate direct revenue, allowing for comparisons of investment strategies across a multi-period horizon under uncertainty.

For the PoR, the different operational factors are used to constrain the solution space. The operational capacity is included in three different ways. For the PV and IRV classes, which have 24/7 availability, a minimum operational and reserve amount of vessels is always required. For the SV, NM and RHIB class, a minimum operational availability is defined per year, since there is flexibility in when the vessels are deployed to perform their task. Maintenance is included in the form of using the current maintenance schedule for the existing vessels and assuming a periodic maintenance interval for the new vessels, accounting for both maintenance downtime and cost. The required infrastructure is included in the cost estimations, with each unit of infrastructure able to support a limited number of vessels, enabling a match of infrastructure capacity with the fleet configuration. The crewing situation is assumed constant and is not included in the framework, as crew configurations are determined by the PoR and fall outside the scope of this thesis.

With global and local emission reduction of the fleet being two of the main drivers behind the research, both are defined as the main objectives within the framework. For global emissions, the LCA method is used to determine CTG and GTC emissions for vessels, batteries, and infrastructure, including WTW CO_2 emissions. The LP of NO_x and PM emissions is assessed separately with the WTW method. By doing so, the origin of the emissions becomes clearer and the impact of different scenarios can be better assessed.

For economic analysis, since vessels do not generate direct revenues, the TCO method is incorporated as an objective in the optimisation framework. This includes capital expenditures such as acquisition and salvage costs, as well as operational costs associated with vessels, batteries, and infrastructure, including insurance and maintenance. Fuel costs are modelled separately and excluded from the general OPEX category to allow for the differentiation between vessels that are in reserve and vessels that are operational and sailing.

Costs are not discounted in this analysis to retain a time-neutral comparison between strategies. In the absence of revenue streams or financial return calculations, the primary objective is to minimise the absolute lifecycle cost rather than to perform a financial valuation. Therefore, the TCO approach provides a straightforward and consistent means of comparing cost outcomes over time without the influence of a discount factor.

The LCA, TCO, and LP indicators are intended to be optimised, while operational constraints and infrastructure limitations define the feasible solution space. These components inform the design of a decision-support framework that balances operational feasibility, environmental sustainability, and economic viability, a balance further guided by the modelling strategies reviewed in the next chapter.

4

Fleet renewal modelling

Fleet renewal plays a crucial role in maritime operations by maintaining vessel efficiency, cost effectiveness, and compliance with evolving regulatory and operational standards. This chapter describes the different methodologies that have been used in previous studies to model the maritime fleet renewal problem and provide decision support to fleet operators. First, the historical development of maritime fleet renewal modelling is described in section 4.1. The different methods used to combine multiple objectives or criteria for decision support are discussed in section 4.2. These methods are analysed comparatively in section 4.3 to determine the most suitable method for PoR. The chapter concludes with a synthesis of the insights in section 4.4.

4.1. Development of fleet renewal modelling

Fleet renewal represents a specialised subset of the broader equipment replacement problem, a topic extensively studied within Operations Research for more than a century. The foundations were laid by Taylor (1923) and Hotelling (1925), who illustrated how depreciation can be statistically accounted for in the economic evaluations of machines. These studies assumed fixed costs, static operational conditions, and no technological innovation in their replacements. Bellman (1955) used the concept of depreciation to create a mathematical formulation of the equipment replacement problem. The mathematical formulation was dependent on the price of a new machine and the output, maintenance, and resale value of the old machine. The old and new machines were assumed to be exactly the same, except for their respective age. In order to solve the problem, Bellman introduced the concept of dynamic programming (DP), a methodology in which a complex problem is broken down into simpler subproblems. Dreyfus (1960) continued on the formulation of Bellman, but instead of assuming the replacement equipment was equal, he added the effect of technological innovation to the equipment replacement problem, making the assumption that the revenue a future machine will be able to generate shall be greater than the revenue of current machines.

Nicholson and Pullen (1971) were the first to adapt the generalised equipment replacement model to the maritime industry, formulating a two-stage DP model that optimised vessel replacement decisions over time, applied to a shipping company that wanted to reduce its number of vessels, based on a ten-year future outlook. In order to save computation time, a heuristic calculation was performed to determine the priority replacement order of the ships in the first stage of the model. In the second stage, DP was used to determine the optimal level of chartering corresponding to the priority replacement order. One of the imperfections of this method was the assumption of one-to-one replacement of vessels, according to Wijsmuller and Beumee (1979), who stated that future ships shall be larger in size and capable of carrying more cargo. To account for this, they developed a linear programming (LiP) model with the objective of maximising the NPV of the assets of a company at the end of the planning horizon.

A key limitation of LiP is its inability to enforce discrete outputs, which can be problematic in real-world fleet planning where decisions such as vessel counts must be integers. To account for this, Cho and Perakis (1996) developed a mixed-integer linear programming (MILP) model, where they combined MFSMP with MFRP for a liner fleet. The objective was to minimise the total cost of the necessary operations while meeting current and future demand. The model decided which routes the ships should sail, which ships should be laid up, and which ships should be purchased to achieve this objective.

Although MILP allows for integer outputs, the computation time, however, also grows. In order to mitigate the issues of using LiP or MILP, Xie et al. (2000) proposed a hybrid model, where LiP was combined with DP to balance computational efficiency and include realistic constraints. The solutions to the fleet deployment problems were solved for a single time step using LiP and fleet development across multiple time steps was addressed using DP. Their approach provided a structured way to make fleet renewal decisions while considering operational dependencies across multiple time periods.

Until now, the discussed articles all assumed set values for their parameters, with no room for uncertainty, Alvarez et al. (2011) changed that by accounting for the uncertainty in the purchase and selling prices of dry bulk ships, by designing an MILP model combined with robust optimisation, for a multi-period fleet sizing and deployment problem. This allowed for the evaluation of varying degrees of risk tolerance with regard to the decisions in selling and purchasing ships, by choosing conservative values for the purchase and selling prices of ships. Meng and Wang (2010) proposed another way to deal with uncertainty, this time with uncertainty in the demand for cargo shipments during a single time step. Upon assuming that the demand is normally distributed, an integer linear programming (ILP) model with chance constraints is developed, where the demand constraint must be met with a set probability. Continuing on this, Meng and Wang (2011) included a hybrid structure similar to that used by Xie et al. (2000). They developed a DP-ILP model for a multi-period liner fleet planning problem. The uncertainty is captured by determining multiple possible scenarios which were solved using DP with the shortest path algorithm. This was built upon multiple ILPs that capture the fleet deployment problem that is solved for each time step. Sønnervik et al. (2024) minimised the total operational and renewal cost, while complying with the emission targets of the Norwegian fishing fleet, to determine the optimal decarbonisation strategy. This was done by introducing low- and zero-emission propulsion systems. The impact of changes in energy cost and emission taxes was included through a sensitivity analysis.

With increased computational power, more advanced approaches to modelling the MFRP have emerged. For example, Bakkehaug et al. (2014) accounted for uncertainty in vessel prices, freight rates, and demand by formulating a multistage stochastic programming (StP) model aimed at minimising total cost. Scenarios were used to represent market volatility, offering a structured basis for planning under uncertainty.Meng et al. (2015) focused on the multi-period fleet renewal problem of liner ships, incorporating the stochastic dependence of the demand for random and period-dependent container shipment. It is formulated as a multistage StP, with different demand scenarios in multiple stages. The uncertainty in demand was assumed to depend on the previous demand. A two-stage model determined the fleet deployment based on expected demand, with the objective of maximising profit. In the second stage, the demand became known, depending on the demand of the previous stage and a random probability. This was used to optimise the number of containers each ship carried, since the deployment itself was already set. After calculating the expected earnings for the different scenarios, dual relaxation and the Lagrangian method were used to find the optimal path over the multiple scenarios. Pantuso et al. (2015) investigated the effect of including uncertainty on the quality of the results, comparing it to a deterministic model using expected values. Both models were applied to the case of a liner shipping company, with the StP model showing substantial benefits compared to the deterministic model. Patricksson et al. (2015) applied StP to the MFRP faced by a liner shipping company, including regional limitations in the form of emission control areas. The objective of the study was to minimise the total expected cost of servicing the given demand, with the vessels generating a negative cost (income). The uncertainty of fuel prices was taken into account in the stochastic model.

Arslan and Papageorgiou (2017) were the first to apply StP to bulk ship fleet renewal, accounting for uncertainty in demand, time charter cost, and voyage charter rate. The objective of the research was to minimise the total cost over the planning horizon. Zhao et al. (2021) applied StP with robust optimisation to formulate a decision plan for a liner fleet on three sulphur reduction technologies, namely fuel-switching, scrubber, and duel fuel engines. The uncertainty of freight demand, charter rate, and fuel price was included, and also the retrofit time was assumed to follow demand, assuming that a higher demand equals greater economic output, resulting in faster retrofit times. Zhao et al. (2024) integrated a two-stage StP model for the renewal of the short-sea liner fleet with conditional value at risk. This model was the first to integrate the financial risks associated with investments in carbon reduction technologies into the StP model. This allowed the model to be used as a framework that balances risk with profitability and environmental compliance. Loennechen et al. (2024) studied the MFRP in a fleet of Supramax bulk carriers, using StP to account for uncertain fuel and carbon prices. Based on different scenarios for emission reduction targets, the favourable power system changes. These findings highlight the importance of well-defined policy trajectories to enable effective long-term planning.

Turan et al. (2020) developed a simulation optimisation framework tailored to address the issue of the mix of military fleets, with the aim of transitioning older vessels while keeping costs low and maintaining operational readiness. The study integrated an improved genetic algorithm (GA) with a capability simulation model to enable a more dynamic assessment of fleet modernisation strategies. Turan et al. (2022) combined the optimisation of strategic workforce planning with fleet renewal, also using a hybrid model combination of GA and system dynamics to generate solutions. The objective function included both total cost and an unavailability penalty to incentivise high fleet availability.

Fee et al. (2019) applied MOO, where GA was used to generate solutions that were evaluated on operational capacity, maintenance availability, crew availability, and ship age. This was applied to five different transition scenarios for the replacement of a frigate class. Turan et al. (2021) developed a hybrid model of MOO and MCDA, including risk assessment. The following objectives were integrated and optimised using MOO: Workforce cost, capability gap, and capital & sustainment cost. To solve this complex problem, the study introduces a hybrid approach that combines a non-dominated sorting genetic algorithm (NSGA) with a system dynamics simulation model. This framework allows for an iterative evaluation of fleet transition strategies. In addition, conditional value at risk was applied to consider different levels of risk tolerance. The generated solutions of the MOO were evaluated using MCDA, ranking the selection of Pareto-optimal solutions, offering a more comprehensive decision support framework.

Ali et al. (2023) expanded fleet renewal modelling to incorporate a more comprehensive framework that integrates planning, scheduling, and operational feasibility, capturing long-term uncertainties in fleet reliability, mission readiness, and budget fluctuations. To manage complexity and long-term uncertainty in mission readiness and budgets, approximate dynamic programming (ADP) was proposed. By approximating near-optimal solutions rather than exact optima, ADP enables a broader exploration of feasible strategies.

The fleet renewal problem is not specifically applied to ships, but is also of importance to other groups of transport assets such as, but not limited to, buses, aircraft, and vehicles. These assets are often economically interdependent and operate in parallel, which means that replacement decisions cannot be determined independently, a challenge known as the parallel replacement problem (Hartman & Tan, 2014). Hsu et al. (2011) studied the impact of stochastic demand on airline replacement strategies, using probabilistic dynamic programming (PDP), to incorporate different scenarios and certain probabilities for each scenario being reached. Parthanadee et al. (2012) studied the commonly applied replacement rules for parallel fleet replacement. Concluding that purchasing only new vehicles was found to be highly cost-inefficient, where as other widely used rules such as the one-purchase choice for each period, older-vehicles-selling first and no-splitting-in-selling are cost-effective rules even though they are not optimal. Fan et al. (2014) applied PDP to optimise the replacement of the vehicle fleet of the Texas Department of Transportation, with stochastically modelled vehicle usage, defining multiple vehicle usage scenarios.

Pelletier et al. (2019) modelled the fleet replacement model as an ILP for a bus fleet. They introduced the cost of installing and using chargers to the objective function, accounting for the cost of the entire transition, including infrastructure investments. Islam and Lownes (2019) combined the economic and environmental factors in a single MILP that minimises life cycle cost, including the social cost of CO2 emissions. Alp et al. (2022) investigated the effect of congestion at charging stations, taking into account the loss of productivity and costs due to the additional waiting time. For this, they used ILP. Castillo Campo and Álvarez Fernández (2023) applied MILP to compare different electric powertrain technologies for delivery vans, including battery electric, fuel cell electric, and hybrid technologies. These options were evaluated based on their operational limitations, such as range, recharge time, and infrastructure requirements, as well as the maturity of the technique and economic feasibility. Zhou et al. (2023) applied ILP to determine when to purchase or salvage a bus and when to deploy a charger, depending on travel demand, charger demand, budget requirements, and cost involved, which included external cost and external health cost, as well as battery recycling costs/profits. Martin et al. (2024) looked into the effect of sustainability commitments by determining the cost-optimal investments using MILP in fossil and renewable fuel technologies for Norwegian transport operators. The findings show that for truck operators, the sustainability commitments incur minimal additional cost, however for the ship and air plane operators, significant additional cost are due to the sustainability commitments. This showed the need for additional policies on fuel cost and carbon pricing.

Sadeghpour et al. (2019) used ADP, solving the problem using GA. The use of ADP allows for relatively quick coverage of a large feasible region to find a near-optimal solution. Winkelmann et al. (2024) formulated fleet renewal as a sequential optimisation problem, considering multiple technologies and operational clusters. APD was proposed to calculate fleet renewal policies to achieve emission goals while optimising TCO.

Giordano et al. (2018) applied MCDA to compare diesel and battery electric delivery vans on emissions and economic performance, assessing the total cost and total emissions over the lifecycle of a van. Du and Kommalapati (2021) looked at the replacement of the public transport fleet, shifting from conventional to electric and diesel-electric powered buses. These technologies were compared using MCDA in terms of lifecycle emissions and TCO. Coppola et al. (2023) focussed on the economic and ecological transformation of the local public transport bus fleets. An LCA of cost and environmental impacts is proposed to identify pathways for the renewal of existing buses, which are compared using a multi-criteria decision matrix.

Aiello et al. (2024) created a MCDA framework that combines TCO with WTW emissions for a fleet of zero-emission vehicles. Moreno Sader et al. (2025) used the same methodology to perform a costing and emission analysis for long-haul battery electric trucks, with overnight charging in the USA. With the current electricity grids composition, the study showed no emission benefit by switching to electric compared to diesel. This showcases the importance of the underlying infrastructure for the energy transition. Castillo and Álvarez (2023) used MOO to optimise the TCO and cumulative emissions in all three scopes, for the replacement of conventional vans with certain types of electric vans.

As summarised in Table 4.1, fleet renewal modelling has progressed markedly over time, evolving from deterministic, single-objective optimisation approaches toward more sophisticated stochastic, approximate, and multi-objective frameworks. Earlier models primarily emphasised economic optimisation using techniques such as LiP, DP, and MILP under fixed input assumptions. However, increasing sustainability expectations, volatile regulatory conditions, infrastructure dependencies, and complex operational interrelations have necessitated more adaptable and integrated modelling strategies.

In light of these growing complexities, recent studies have increasingly adopted MCDA and MOO approaches to address the inherently multi-criteria nature of fleet renewal decisions. These methodologies offer structured frameworks for navigating trade-offs between competing objectives, particularly economic performance, environmental impact, and operational feasibility. For example, Aiello et al. (2024), Coppola et al. (2023), and Moreno Sader et al. (2025) employed MCDA to evaluate alternative transport technology based on financial and environmental indicators, while Castillo and Álvarez (2023) and Turan et al. (2021) used MOO frameworks to identify Pareto-optimal fleet transition strategies across multiple objectives.

Within the context of the PoR, where strategic fleet renewal requires balancing economic viability, emissions reductions, and operational resilience, MCDA and MOO emerge as particularly well-suited methodologies. Their capacity to support scenario-based decision making under uncertainty, while incorporating stakeholder preferences and multiple performance dimensions, forms a strong basis for the proposed framework. The next section outlines the fundamental structure, advantages, and application considerations of MCDA and MOO techniques as used in fleet renewal literature. This general overview provides the methodological foundation upon which the subsequent comparative analysis for the PoR case is built.

Table 4.1: Overview of reviewed literature on fleet renewal modelling.

Reference	Uncertainty	Algorithm Design Approach	Algorithm Design Accuracy
Nicholson and Pullen (1971)	determ.	DP	exact
Wijsmuller and Beumee (1979)	determ.	LiP	exact
Cho and Perakis (1996)	determ.	MILP	exact
Xie et al. (2000)	determ.	LiP + DP	exact
Meng and Wang (2010)	determ.	ILP + CCP	exact
Meng and Wang (2011)	stoch.	ILP + PDP	exact
Alvarez et al. (2011)	determ.	MILP + RO	exact
Hsu et al. (2011)	stoch.	PDP	exact
Parthanadee et al. (2012)	determ.	ILP	exact
Bakkehaug et al. (2014)	stoch.	StP	exact
Fan et al. (2014)	stoch.	PDP	exact
Meng et al. (2015)	stoch.	StP	exact
Pantuso et al. (2015)	stoch.	StP	exact
Patricksson et al. (2015)	stoch.	StP	exact
Arslan and Papageorgiou (2017)	stoch.	StP	exact
Giordano et al. (2018)	determ.	MCDA	exact
Pelletier et al. (2019)	determ.	ILP	exact
Sadeghpour et al. (2019)	stoch.	ADP	approx.
Islam and Lownes (2019)	determ.	MILP	exact
Fee et al. (2019)	stoch.	MOO	approx.
Turan et al. (2020)	stoch.	MH	approx.
Turan et al. (2021)	stoch.	MOO + MCDA	approx.
Zhao et al. (2021)	stoch.	StP	approx.
Du and Kommalapati (2021)	determ.	MCDA	exact
Turan et al. (2022)	stoch.	MH	approx.
Alp et al. (2022)	determ.	ILP	exact
Castillo and Álvarez (2023)	determ.	MOO	exact
Ali et al. (2023)	stoch.	ADP	approx.
Coppola et al. (2023)	determ.	MCDA	exact
Castillo Campo and Álvarez Fernández (2023)	determ.	MILP	exact
Zhou et al. (2023)	determ.	MILP	exact
Zhao et al. (2024)	stoch.	StP	exact
Aiello et al. (2024)	determ.	MCDA	exact
Winkelmann et al. (2024)	stoch.	ADP	approx.
Sønnervik et al. (2024)	determ.	ILP	exact
Martin et al. (2024)	determ.	MILP	exact
Loennechen et al. (2024)	stoch.	StP	approx.
Moreno Sader et al. (2025)	determ.	MCDA	exact

DP - Dynamic programming, **LiP** - Linear programming, **MILP** - Mixed integer linear programming, **ILP** - Integer linear programming, **CCP** - Chance Constrained Programming, **PDP** - Probabilistic dynamic programming, **RO** - Robust Optimisation, **StP** - Stochastic programming, **MCDA** - Multi-Criteria Decision Analysis, **ADP** - Approximate dynamic programming, **MOO** - Multi-objective optimisation **MH** - Metaheuristics.

4.2. Multi-criteria and multi-objective optimisation techniques

Historically, fleet renewal programmes have focused mainly on reducing costs or maximising profits. However, as outlined in chapter 1, many organisations now also commit to minimise their environmental impact. Consequently, fleet renewal strategies must increasingly be assessed through multiple criteria or optimised across multiple, often conflicting, objectives. The evaluation under multiple criteria falls under the MCDA approach, to provide a framework for the decision maker to make informed decisions. Different methods for employing this are elaborated in subsection 4.2.1. Optimisation of multiple objectives falls under the MOO approach, where the problem is solved mathematically. The methods for employing MOO are discussed in subsection 4.2.2.

4.2.1. Multi-criteria decision analysis

MCDA is a branch of decision analysis designed to support rational decision making in contexts with multiple, often conflicting, criteria and significant uncertainty (Hillier & Lieberman, 2021). Unlike standard decision analysis, which evaluates options based on a single pay-off function, MCDA requires weighing and combining several objectives, such as cost, emissions, and operational availability, to arrive at a preferred choice. A key step in MCDA is determining the relative importance of the criteria, often through stakeholder input or expert elicitation.

Different methodologies were examined to apply MCDA to identify the approaches most suitable for the context of fleet renewal. TOPSIS emerged directly from previous fleet renewal studies, where it was successfully applied to classify technology or policy alternatives under multiple criteria. Other relevant MCDA methods were selected for consideration based on their prevalence in the decision science literature, as reported in the article by Taherdoost and Madanchian (2023b).

This selection ensures that the MCDA methodology aligns both with practices in fleet management literature and with broader trends in multi-criteria decision support.

Analytic Hierarchy Process

The *Analytic Hierarchy Process* (AHP) is designed to assign weights to different criteria. It consists of three stages: Complexity structuring, measurement, and synthesis. In the first stage, the hierarchical structure of the problem is displayed, with the overall goal at the top level, the criteria at the second level, and the alternative decisions at the final level. An overview of the hierarchical structure of AHP is given in Figure 4.1. In the measurement stage, the relative importance of the different criteria is assessed, as well as the performance of each alternative to meet the different criteria. This is done using pairwise comparisons between each of the different (sub)criteria (Kana, 2024). The different alternatives are also compared pairwise for each of the criteria. Comparisons are often made by multiple people and the mean result is taken. In the synthesis stage, the results of pairwise comparisons are combined in matrices, depending on the comparison that was performed. By calculating the eigenvector of these matrices, the weights of the criteria and the performance of the solutions for each criterion are determined. Combining these two leads to the optimal alternative (Hillier & Lieberman, 2021). The benefits of AHP include its structured approach, which provides a clear hierarchical framework to evaluate complex decisions with multiple criteria and alternatives. It allows for the integration of both quantitative and qualitative criteria, making it versatile for various decision-making scenarios. In addition, the hierarchical structure and pairwise comparisons make the decision-making process transparent and easy to understand. However, AHP can be time-consuming and labour intensive due to the process of setting up the hierarchy and performing pairwise comparisons. The results are also sensitive to subjective judgments from decision makers, which can introduce bias. Furthermore, AHP may not be suitable for very large or very small decision problems due to the complexity involved.

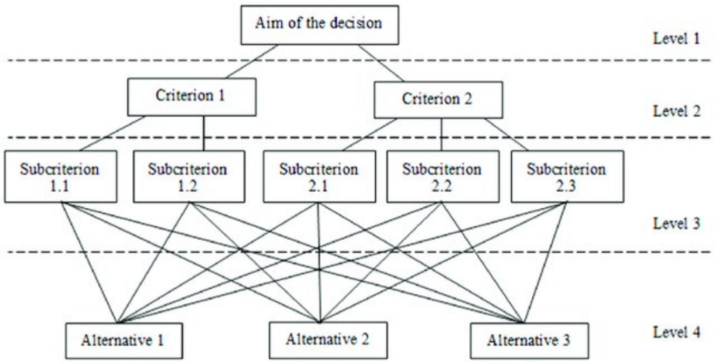


Figure 4.1: Analytical Hierarchy Process structure (Watróbski et al., 2016).

Analytic Network Process

The *Analytic Network Process* (ANP) is an extended version of AHP that enables feedback and interaction between clusters, making it more comprehensive. The structure follows the structure of an AHP, with the problem represented as a network in the first stage. In the second stage, the same comparisons are carried out; however, the interdependencies are also examined, so the impact from one element on another can be displayed by an eigenvector. In the third step, everything is composed in a super-matrix which is then weighted and raised to exponential powers until the elements of the super-matrix are identical. Finally, the different elements' priorities are determined by normalising the clusters of the final matrix. Using the alternatives column in the normalised super-matrix, the priority weights are found, and the alternative with the highest weight is the optimal decision (Taherdoost & Madanchian, 2023c). The benefits of ANP include its ability to capture the interdependencies between criteria and alternatives, providing a more comprehensive analysis compared to AHP. It is useful for complex decision making scenarios with interdependent factors. However, ANP is more complex and time-consuming than AHP due to the additional steps involved in capturing interdependencies. It requires more computational effort and expertise to implement and may be overkill for simpler decision making problems where interdependencies are not significant.

TOPSIS

TOPSIS stands for *Technique for Order Preference by Similarity to Ideal Solution*, and is a highly adopted method in the field of MCDA (Chakraborty, 2022). The main concept of TOPSIS is that the optimal solution is closest to the ideal solution and farthest from the negative ideal solution. It is applied to a decision matrix in six steps. The decision matrix consists of alternatives and criteria. In the first step, the decision matrix is normalised. In the second step, the weighted normalised matrix is calculated. In the third step, the ideal positive and negative solutions are determined. The fourth step calculates the distance of the alternatives from the positive and negative ideal solutions, often using the Euclidean distance. The fifth step calculates the relative proximity factor to these points. Finally, the relative proximity factors are ranked from best to worst in the final step to find the optimal solution (Madanchian & Taherdoost, 2023). The benefits of TOPSIS include its simplicity and straightforward implementation, making it accessible to decision makers with varying levels of expertise. It produces a clear ranking of alternatives based on their proximity to the ideal and non-ideal solution, enhancing interpretability. Moreover, TOPSIS is a versatile tool, suitable for decision-making scenarios involving numerous criteria and alternatives. However, the results of TOPSIS are sensitive to the normalisation method used, which can affect the ranking of alternatives. It does not provide insight into the relative importance of criteria, which may be crucial for some decision making scenarios. Furthermore, TOPSIS may not be suitable for very large or very small decision problems due to its simplicity.

ELECTRE

ELECTRE stands for *Elimination Et Choix Traduisant la Reality*, which allows for the direct comparison of alternatives based on criteria and accounts for the preference and importance of decision makers, generating a ranking of relative strengths and weaknesses. This is done by first creating a decision matrix and determining the weights of the criteria. The decision matrix is then normalised, followed by creating a weighted normalised matrix. The alternatives are compared pairwise into two separate subsets, *Concordance* and *Discordance*, noting which of the alternatives performs better on which criteria. In the next step, the concordance matrix is built, taking the sum values of the weights associated with the concordance of each alternative. The *discordance matrix* is built by taking the max discordance between two alternatives and dividing this by the max discordance over all criteria. The higher this value, the less favourable the alternative. These matrices are indexed by taking a threshold value, often 0.7 for concordance and 0.3 for discordance. The matrices are combined element-wise to form the *aggregate dominance matrix*. If there is one in the column, it shows that the alternative is dominated by another and that the alternative can be discarded (Taherdoost & Madanchian, 2023d). The benefits of ELECTRE include its ability to provide a clear ranking of alternatives based on concordance and discordance, making it easy to interpret. It is useful for decision making scenarios with multiple criteria and alternatives and accounts for the preferences and importance of decision makers. However, ELECTRE is more complex and time-consuming than TOPSIS due to the additional steps involved in calculating concordance and discordance. It requires more computational effort and expertise to implement and may not be suitable for very large or very small decision problems due to its complexity.

PROMETHEE

PROMETHEE is short for *Preference Ranking Organisation Method for Enrichment of Evaluations* and consists of two stages (Brans et al., 1986). In the first stage, a general decision matrix of alternatives and criteria is developed and the weights between the different criteria are determined. In the second stage, the decision matrix is used to calculate the outranking relation of the alternatives, where the preference of one alternative over another is given by a number between zero and one. With zero corresponding to non-preference and one corresponding to a strict preference. This is followed by determining the aggregated preference by taking the sum of the outranking relations between different alternatives over all weighted criteria. This leads to a global preference for one alternative over another, which is then used to determine the out-ranking flows. The second stage of the method consists of taking the positive and negative outranking flows for a certain alternative by taking the sum of the global preferences over all other alternatives, divided by the number of compared alternatives. By comparing the out-ranking flows, the different alternatives are ranked (Taherdoost & Madanchian, 2023a). The benefits of PROMETHEE include its ability to provide a clear ranking of alternatives based on out-ranking flows, which makes it easy to interpret. It allows for a detailed evaluation of alternatives based on multiple criteria and accounts for the preferences and importance of decision makers. However, PROMETHEE is more complex and time-consuming than other MCDA methods because of the additional steps involved in calculating out-ranking flows. It requires more computational effort and expertise to implement and may not be suitable for very large or very small decision problems due to its complexity.

Data Envelopment Analysis

Data Envelopment Analysis (DEA) is a non-parametric technique used to assess the relative efficiency of alternatives, treating each as a decision-making unit (DMU) that converts inputs into outputs (Charnes et al., 1978). It constructs an empirical efficiency frontier and assigns efficiency scores from 0 to 1, with fully efficient units scoring 1. DEA optimises the input and output weights for each DMU individually, ensuring the most favourable efficiency score under the constraint that no DMU exceeds an efficiency of 1. The advantages of DEA include its ability to handle multiple inputs and outputs simultaneously without assuming a pre-defined trade-off between them. It is particularly useful for benchmarking and identifying best-practice frontiers. However, DEA does not incorporate stakeholder preferences or subjective criteria, which limits its use in value-sensitive decision making.

While MCDA supports structured evaluation of predefined alternatives, MOO methods mathematically generate optimal solutions by modelling multiple objectives. The next subsection details common MOO techniques.

4.2.2. Multi-objective optimisation

Multi-objective programming extends traditional single-objective programming by optimising multiple objectives. Traditionally, different methods can be classified into three categories. In the first category, preferences are made *a priori*, so they can be incorporated into the model. The second category consists of interactive methods in which preferences are progressively articulated. In the third category, preferences are not included in the model and added *a posteriori* by the decision maker after finding the set of solutions (Antunes et al., 2016).

In contrast to single-objective programming, where a single optimal solution is sought, multi-objective programming seeks a set of *non-dominated* (Pareto-optimal) solutions that represent trade-offs between objectives. The solutions in this set cannot be improved for an objective without worsening the result for another objective. The decision space of the problem is mapped in a p-dimensional space, corresponding to the p-number of objectives, called the *objective function space*. Each potential solution is represented by a vector in this space, the components of which are the values of each objective function.

The general MOLP problem is formulated as follows (Antunes et al., 2016):

$$\text{max} \begin{cases} z_1 = f_1(\mathbf{x}) = \mathbf{c}_1\mathbf{x} \\ \vdots \\ z_p = f_p(\mathbf{x}) = \mathbf{c}_p\mathbf{x} \end{cases} \qquad \text{"Max"}\ \mathbf{z} = \mathbf{f}(\mathbf{x}) = \mathbf{C}\mathbf{x} \tag{4.1}$$
$$\text{subject to } \mathbf{x} \in \mathbf{X} = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{A}\mathbf{x} = \mathbf{b}, \mathbf{x} \geq 0\}$$

$\mathbf{A}$ is the matrix of technological coefficients corresponding to the left-hand side of the constraints, $\mathbf{b}$ is the vector containing the right-hand side values of the constraints. $\mathbf{C}$ contains the row vectors $\mathbf{c}_1, \cdots, \mathbf{c}_p$, corresponding to the coefficients of the objective functions.

The non-dominated solutions are placed together in the *Pareto front*. The ideal solution or utopia point is the hypothetical solution in which each objective reaches its optimal value without considering the trade-off and constraints due to the other objectives (Hillier & Lieberman, 2021). The objective function space for a two-dimensional problem is shown in Figure 4.2.

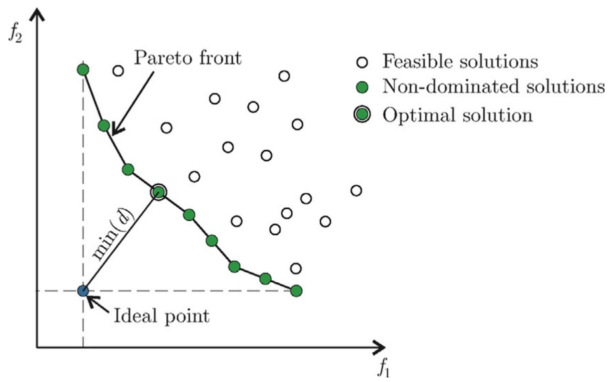


Figure 4.2: Objective function space (Bre & Fachinotti, 2017).

The most common method to compute the non-dominated solutions is by using a scalarization technique, which transforms the multi-objective problem into a single-objective problem that may be solved repeatedly with different parameters. These functions are called *scalarization* functions. The parameters in these functions depend on the preference of the decision maker, making it *a priori* techniques. The benefit of these techniques is that the computational effort should not be too demanding. There are three main scalarization techniques employed, namely, *weighted sum*, *ϵ-constraint* & *reference point* (Antunes et al., 2016).

Weighted sum

The weighted sum method is a commonly used approach to solve the multi-objective optimisation problem. A weight is assigned to all objective functions and the sum of the assigned weights must be one. A single objective function is defined to maximise or minimise the sum of the individual objective functions multiplied by their weight. By using additional constraints in the form of the objective functions, any non-dominated solution can be reached for a (mixed) integer linear programming model. For this method, the preference of the different objectives should be known in order to assign the weights correspondingly (Antunes et al., 2016). The scalarization function is shown in Equation 4.2. This formulation is straightforward to implement and computationally efficient, allowing it to be solved as a single-objective (integer) linear programme. It has been used by Castillo and Álvarez (2023) to combine the objectives of minimising the TCO and lifecycle emissions of the fleet. The benefits of the weighted sum method include its simplicity and straightforward implementation. Provides a clear objective function that combines multiple objectives into a single optimisation problem. In addition, it allows for the integration of preferences by assigning weights to different objectives. However, the weighted sum method is only suitable for problems with convex Pareto fronts, as it cannot find non-convex Pareto optimal solutions. It requires pre-defined weights for each objective, which may not always be available or easy to determine. Furthermore, it does not allow for interactive feedback from the decision maker during the optimisation process.

$$
\begin{aligned}
\max \; & \sum_{k=1}^{p} \lambda_k f_k(\mathbf{x}) \\
\text{s.t.} \quad & f_k(\mathbf{x}) \geq e_k, \quad k = 1, \dots, p \\
& \mathbf{x} \in X
\end{aligned}
\tag{4.2}
$$

ϵ-constraint

The ϵ-constraint method selects one of the objective functions to optimise and considers the other objective functions as constraints. This is done similarly to the weighted sum method by assigning a weight of one to the primary objective function and zero or a very small value to the other functions. By changing the value ϵ_k for the constraint, the objective region is scanned (Antunes et al., 2016). The scalarization function is shown in Equation 4.3. The benefits include its flexibility in allowing for the exploration of the Pareto front by adjusting the ϵ values. Provides a clear objective function that focusses on one primary objective while considering others as constraints. In addition, it enables a comprehensive evaluation of the trade-offs between the objectives. However, the method is more complex and time-consuming than the weighted sum method because of the need to solve multiple optimisation problems. It requires more computational effort and expertise to implement. Furthermore, it requires pre-defined ϵ values for the constraints, which may not always be available or easy to determine.

$$
\begin{aligned}
\max \quad & f_i(\mathbf{x}) + \rho \sum_{\substack{k=1, \\ k \neq i}}^{p} f_k(\mathbf{x}) \\
\text{s.t.} \quad & f_k(\mathbf{x}) \geq e_k, \quad k = 1, \dots, p, k \neq i \\
& \mathbf{x} \in X
\end{aligned}
\tag{4.3}
$$

Reference point

In the reference point method, the ideal solution is used as a reference point. The distance between the solution and the ideal solution can be calculated in multiple ways, for example, using the *city block*, *Euclidean*, or *Chebyshev* distance. The distance to the ideal point is multiplied by the weight of the objective (Antunes et al., 2016). The scalarization function is given in Equation 4.4. The benefits of the reference point method include its flexibility in allowing for the use of different reference points, providing flexibility in the optimisation process. Provides a clear objective function that minimises the distance to the ideal solution. In addition, it enables a comprehensive evaluation of the trade-offs between the objectives. However, the reference point method is more complex and time-consuming than other scalarization techniques due to the need to define and calculate distances to the reference point. It requires more computational effort and expertise to implement. Furthermore, it requires pre-defined reference points, which may not always be available or easy to determine.

$$\begin{aligned} &\min \left(\max_{k=1,\dots,p} \left\{ \lambda_k \left(z_k^+ - f_k(\mathbf{x}) \right) \right\} \right) \\ &\text{s.t.} \quad \mathbf{x} \in X \end{aligned} \tag{4.4}$$

Another way to determine the optimal solutions is by generating solutions and keeping track of their performance. This can be categorised into two main classes: *evolutionary* and *swarm-based* techniques. These techniques fall into the *a posteriori* methods. Where swarm-based techniques are mainly focused on continuous variables, evolutionary algorithms can handle both (Clerc, 2006). For this study, due to the discrete nature of fleet renewal, the focus will continue on evolutionary algorithms.

Evolutionary algorithms

Make use of the concept of natural evolution to find the optimal set of solutions. There are three types of algorithms: *dominance-based* algorithms, *indicator-based* algorithms, and *decomposition-based* algorithms. Dominance-based algorithms assign fitness based on Pareto dominance of the solution. Indicator-based algorithms use indicators to determine the selection of individuals. Decomposition-based algorithms decompose the problem into smaller sub-problems using scalarization, and these are solved collaboratively. Whereas the first type mainly deals with fewer objectives, the latter two types are more suitable for problems with a higher number of objectives (Sharma & Kumar, 2022).

Evolutionary algorithms face significant challenges when dealing with constraints, so in order to ensure that the solutions are feasible, the use of constraint handling techniques, such as penalty functions, constraint dominance, and feasibility rules, is employed. These techniques can increase the complexity and computational effort required, as the algorithm must repeatedly check the feasibility of solutions and apply constraint handling techniques.

Fitness functions are applied to evaluate the generated solutions and play a crucial role to guide the selection process. In the context of constrained optimisation, fitness functions must not only evaluate the objective functions but also account for the feasibility of solutions. Penalty-based fitness functions, for example, add a penalty to the fitness value of infeasible solutions, making them less likely to be selected. Constraint dominance prioritises feasible solutions over infeasible ones, ensuring that the algorithm focusses on finding feasible solutions before optimising the objective functions. Adaptive fitness functions can dynamically adjust the evaluation criteria during the optimisation process, allowing the algorithm to adapt to the changing landscape of the solution space.

General form

Evolutionary algorithms work by defining a population consisting of individuals. The individual represents a potential solution. The initial population is generated at random. This population is then exposed to natural selection, where the individuals are evaluated on their fitness. Individuals with a high fitness are selected to become *parents*, who develop offspring called *children*. Parents and children become the new population, and the other individuals are removed to maintain a constant population size. To ensure that there is variation in the solutions, two concepts are applied. *Crossover* ensures that part of the properties of the parents are represented in the children, while *mutation* is applied probabilistically to individuals, introducing random changes that diversify the offspring beyond the direct traits of their parents. This ensures that the algorithm is spread over the objective function space. After the new population has been generated, the fitness is again assessed, and the process continues until the termination criteria are met. The benefits of evolutionary algorithms include their flexibility in handling both continuous and discrete variables, making them suitable for a wide range of problems. They provide robust solutions by exploring a large solution space and are adaptable to changes in the problem formulation or objectives. However, evolutionary algorithms require significant computational effort and expertise to implement. They are more complex and time-consuming than other optimisation methods. In addition, the results are sensitive to the choice of parameters, which can affect the quality of the solutions (Selçuklu, 2023).

Non-Dominated Sorting Genetic Algorithm-II

NSGA-II, developed by Deb et al. (2000), is one of the most influential and widely adopted multi-objective evolutionary algorithms. NSGA-II introduces an efficient non-dominated sorting procedure and a crowding distance mechanism to ensure diversity in the population. Individuals are ranked according to Pareto dominance in successive fronts, with the best front assigned the highest rank. Within each front, a crowding distance is computed based on the relative spacing between neighbouring solutions in the objective space. This helps guide the selection toward both convergence and diversity. The algorithm implements elitism by combining the parent and offspring populations before sorting and selecting, thereby ensuring the survival of the best individuals. NSGA-II is particularly effective for problems with two or three objectives and is computationally efficient with a complexity of $O(M * N^2)$, where M is the number of objectives and N the size of the population. Its main disadvantages are the reduced performance in problems with many objectives (often more than three), where dominance-based methods tend to lose resolution and diversity. Nevertheless, its simplicity, robustness, and wide availability in software libraries make it a default choice for many real-world multi-objective optimisation problems.

Strength Pareto Evolutionary Algorithm 2

The Strength Pareto Evolutionary Algorithm 2 (SPEA2) is an advanced multi-objective evolutionary algorithm introduced by Zitzler et al. (2001) as an improvement over the original SPEA. It addresses two critical limitations of its predecessor: lack of elitism and insufficient preservation of diversity. SPEA2 maintains an external archive of non-dominated solutions, which directly contributes to elitism and supports convergence to the Pareto front. Each individual in the combined population (archive + current generation) is evaluated using a fitness function that combines two components: *Strength*, which quantifies how many individuals are dominated by a given solution, and a *density estimate* based on the distance to the k-th nearest neighbour in objective space. This dual mechanism allows SPEA2 to achieve both convergence and diversity among solutions. However, SPEA2 can be computationally more expensive than other evolutionary algorithms due to archive management and distance-based calculations. Furthermore, the quality of the results depends on the choice of archive size and neighbourhood size k. Despite these challenges, SPEA2 has been shown to outperform earlier algorithms in maintaining a well-distributed Pareto front across a wide range of test problems and is widely used in engineering and operations research contexts.

Conclusion

Within the fields of MOO and MCDA, various methods have been developed to address complex decision problems involving multiple, often conflicting objectives. MOO focusses on identifying a set of efficient solutions that represent optimal trade-offs across objectives, while MCDA provides a structured framework to evaluate these solutions in light of context-specific preferences and qualitative considerations.

Together, they offer complementary capabilities: MOO generates the technical trade-off space and MCDA translates that space into actionable priorities for decision makers. This complementarity underpins the comparative analysis that follows, which evaluates their suitability to guide sustainable fleet renewal decisions in the PoR context.

4.3. Comparative analysis

This section presents a comparative evaluation of modelling approaches relevant to strategic fleet renewal under sustainability constraints. Two methodological domains are considered: MOO techniques, which are used to generate a range of trade-off solutions, and MCDA techniques, which are used to evaluate and prioritise these solutions in line with stakeholder preferences. The comparison aims to identify the strengths and limitations of each method in relation to criteria such as transparency, ease of use, constraint handling, trade-off exploration, and stakeholder alignment. The goal of this comparison is to critically assess the methods and identify those most appropriate for the fleet renewal case study.

4.3.1. Multi-objective optimisation comparison

Within the context of MOO, three scalarization methods and two evolutionary algorithms have been reviewed for their applicability to the fleet renewal problem: the weighted sum method, the ε-constraint method, the reference point method, NSGA-II and SPEA2. These methods are compared according to four grouped criteria: *usability*, *technical capability*, *decision relevance*, and *operational factors*. The comparison results are presented in Table 4.2.

Criterion	**Weighted Sum**	**Reference Point**	**ε-Constraint**	**NSGA-II / SPEA2**
Usability				
Ease of use	High	Medium	High	Medium
Transparency	High	Medium	High	Low
Technical capability				
Constraint handling	High	High	High	Low
Trade-off exploration	Low	Medium	High	High
Decision relevance				
Preference inclusion	Yes	Yes	No	No
Operational				
Computation time	Low	Medium	High	High
Result reproducibility	High	High	High	Low

Table 4.2: Comparison of multi-objective optimisation methods.

The ε-constraint method aligns most closely with the specific requirements of the PoR case study. It systematically explores the Pareto front by optimising one objective while treating the others as constraints. This provides a high level of transparency and ensures constraints satisfaction, which is crucial for operational planning. One drawback is the increased computational burden, due to repeated single-objective solves.

The weighted-sum method offers simplicity and fast computation, but is limited in its ability to fully explore trade-offs. The predetermined weights introduce subjectivity and limit the discovery of non-convex portions of the Pareto front. The reference point method addresses this by using aspiration levels for each objective, though it requires a priori input and may not generate full front coverage.

NSGA-II and SPEA2, two evolutionary algorithms, perform well in generating well-distributed Pareto fronts and are widely used in academic research. NSGA-II applies non-dominated sorting and crowding-distance metrics, while SPEA2 enhances convergence through elitism and strength ranking. However, both methods rely on stochastic mechanisms, resulting in non-deterministic outputs, and have limited capability to strictly enforce constraints. These drawbacks, combined with their lower transparency and high computational load, make them less suitable for stakeholder-facing, policy-constrained applications such as the PoR fleet renewal challenge.

In conclusion, while each MOO method offers valuable features, the ε-constraint method strikes the most effective balance between transparency, reproducibility, and technical robustness for the case study.

4.3.2. Multi-criteria decision analysis comparison

To evaluate and rank the non-dominated alternatives obtained from the optimisation stage, a range of MCDA techniques are considered. These methods serve to help decision makers translate multidimensional outcomes into actionable priorities under different stakeholder views. The following methods are evaluated: AHP, ANP, TOPSIS, PROMETHEE, ELECTRE, and DEA. Each method is assessed according to four core criteria: transparency, ease of use, dependency handling, and inclusion of preferences. These results are summarised in Table 4.3.

Criterion	AHP	ANP	TOPSIS	PROMETHEE	ELECTRE	DEA
Transparency	Medium	Low	High	Medium	Low	Medium
Ease of Use	Medium	Low	High	Medium	Medium	Medium
Dependency Handling	No	Yes	No	Limited	No	Implicit
Preference Inclusion	Yes	Yes	Yes	Yes	Yes	No

Table 4.3: Comparison of multi-criteria decision analysis methods.

AHP is a structured method widely used in MCDA applications, particularly when transparency is important and the number of criteria is moderate. However, it becomes less tractable as the number of criteria and alternatives grows due to the exponential increase in pairwise comparisons. ANP extends AHP by incorporating interdependencies among criteria, improving expressiveness but reducing interpretability and ease of use.

While PROMETHEE enables nuanced preference modelling and detailed ranking, the interpretability of its results may pose challenges for non-technical stakeholders. ELECTRE is suitable for eliminating dominated alternatives, but does not yield a full ranking and requires subjective concordance and discordance thresholds.

DEA represents a distinct approach among the MCDA methods discussed. It evaluates relative efficiency by comparing alternatives as decision-making units that transform multiple quantitative inputs into multiple outputs. DEA avoids subjective weighting by constructing an empirical efficiency frontier. However, this strength is also a limitation: DEA cannot reflect explicit stakeholder preferences and assumes consistent quantitative data structures across all alternatives. Its applicability is strongest when the inputs and outputs are strictly numerical and directly comparable, which may not hold in complex multi-dimensional trade-off scenarios like public-sector fleet renewal.

In general, the choice of the MCDA method involves a trade-off between interpretability, analytical robustness, and preference expressiveness. TOPSIS was ultimately selected for this thesis due to its intuitive structure, computational efficiency, and suitability for transparent stakeholder engagement within the PoR context.

4.3.3. Synthesis of comparative results

The comparative analysis of MOO and MCDA techniques reveals complementary strengths that can be strategically combined to support sustainable fleet renewal decisions at the PoR. The ε-constraint method demonstrated the highest alignment with case-specific requirements among MOO techniques, due to its transparency, strong constraint handling capacity, and suitability for systematic trade-off exploration. It enables decision makers to construct a Pareto front without requiring a priori weighting of objectives, thus maintaining neutrality in the exploration of conflicting goals such as life-cycle emissions and cost.

Among the MCDA methods, TOPSIS was identified as particularly suitable for evaluating the non-dominated solutions generated by MOO. Its conceptual simplicity, visual interpretability, and low computational burden make it accessible to a wide range of stakeholders. In contrast to efficiency-based techniques like DEA, which evaluate performance based on objective input-output relationships, TOPSIS enables ranking alternatives based on explicitly defined stakeholder preferences, which is essential in this value-sensitive context.

The combination of the ε-constraint method for solution generation and TOPSIS for solution selection allows for a transparent and modular two-stage decision support framework. This hybrid structure aligns with the operational and strategic needs of the PoR by enabling a clear separation between technical optimisation and stakeholder-driven decision making. It supports both exploratory insight and actionable guidance and forms the basis for the modelling approach proposed in the next chapter.

4.4. Conclusion

Fleet renewal modelling has evolved significantly from deterministic linear optimisation approaches to more advanced techniques, including stochastic optimisation, MOO, and approximate methods. This evolution reflects the increasing complexity of decision making in fleet management, where multiple objectives, such as cost minimisation and sustainability, must be balanced against various constraints, including budget limitations, regulatory compliance, and operational feasibility.

LiP and MILP remain fundamental in deterministic formulations, offering high accuracy and interpretability. However, these methods often struggle with real-world complexities, such as uncertainty and non-linear relationships. To address these challenges, extensions such as chance-constrained programming and robust optimisation have been developed, enabling models to account for risk sensitivity and uncertainty margins.

StP has emerged as a powerful tool for capturing uncertainty over time by defining multiple scenarios, allowing for the identification of optimal solutions that perform best on average under uncertain conditions. ADP further enhances decision making by finding near-optimal solutions, reducing computation time while covering a broad feasible region. These advances are particularly valuable in complex, real-world applications where traditional methods fall short.

Building on deterministic methods such as LiP and MILP, MOO has emerged as a robust framework for navigating trade-offs inherent in fleet renewal strategies. MOO methods aim to generate a diverse set of non-dominated solutions, offering a comprehensive view of the solution space. Two common families of MOO techniques are scalarization methods, such as the weighted sum or ϵ-constraint, which convert the multi-objective problem into a series of single-objective ones; and evolutionary algorithms, such as NSGA-II or SPEA2, which simulate natural selection processes to iteratively improve solution sets.

MCDA has gained prominence as a means to formalise and integrate stakeholder preferences within multi-criteria decision environments. Methods like AHP, TOPSIS, and ELECTRE offer structured approaches to evaluating alternatives based on multiple criteria, incorporating subjective preferences and uncertainties into the decision-making process.

Although there is a wide range of methods for addressing problems with multiple objectives and decision criteria, each comes with trade-offs in complexity, transparency, and alignment with stakeholder engagement. In the context of the PoR, where transparency, traceability, and ease of use are paramount, the selected hybrid approach offers a balanced solution. The use of MOO enables the generation of Pareto-optimal solutions without requiring predefined preferences. To do so, the ϵ-constraint method is employed, which allows one primary objective to be optimised while the other is transformed into a constraint. By changing the bounds of this constraint, the trade-offs are explored throughout the solution space.

Once a diverse set of Pareto-optimal solutions is generated, MCDA is applied a posteriori to guide the selection of preferred alternatives. The chosen technique, TOPSIS, ranks alternatives based on their geometric proximity to an ideal and anti-ideal point across all criteria. This method is well-suited to contexts where decision makers wish to balance conflicting objectives and require intuitive, interpretable outputs. The combination of ε-constraint optimisation and TOPSIS-guided evaluation results in a decision support framework that is both operationally viable and methodologically transparent. This hybrid structure forms the foundation for the framework introduced in chapter 6.

5 Conclusion

This chapter concludes the literature review by summarising the key insights and limitations identified in the literature on fleet renewal. Although substantial work has been done in areas such as cost modelling and emissions evaluation, existing approaches often treat these aspects in isolation and lack mechanisms for balancing trade-offs transparently, especially in combination with a focus on the operational capacity during the fleet renewal.

The reviewed literature also reveals significant blind spots in the inclusion of operational capacity. Operational aspects such as maintenance-induced downtime and recertification requirements are often omitted, despite their real-world relevance in determining vessel availability and long-term planning feasibility, as well as the inclusion of sufficient skilled personnel and facilities to support fleet renewal. Likewise, while many models assign a monetary cost to emissions through carbon pricing or taxation, the correct amount to fully internalise the external cost is difficult to determine, with market prices for CO_2 emissions far below the estimated cost of removing CO_2 from the air. Different frameworks are detailed for determining the amount of emissions, together with multiple impact categories.

Beyond technical and operational factors, this review also acknowledges the ethical complexities associated with emerging fleet technologies. The sourcing of battery materials, such as lithium, cobalt, and nickel, raises issues related to labour rights, environmental degradation, and geopolitical dependencies. Mining activities may also cause significant pollution emissions in the local area. The trade-off between less pollution in one location, at the cost of more pollution and poor labour conditions at another location, raises serious ethical questions. This underlines the importance of a transparent module, where trade-offs can be mapped, to provide the needed decision support to the decision maker.

For decision support, mathematical optimisation has been used to generate optimal solutions or MCDA to compare different alternatives based on multiple criteria. These findings collectively justify the development of a two-stage decision support model, combining MOO with MCDA. This structure allows the exploration of trade-offs via non-dominated solution sets and the incorporation of stakeholder preferences a posteriori. The main objectives to optimise are the global CO_2 emissions, assessed by the LCA method for the CTG and GTC stages, together with LP in terms of CO_2, NO_x and PM, assessed with the WTW framework. The economic objective is the minimisation of the TCO. By analysing different scenarios and strategic pathways, multiple future perspectives can be investigated in terms of pricing and emission consequences. The next chapter outlines the detailed methodology of this framework, structured to address the complex, multi-dimensional nature of fleet renewal in a transparent and adaptable way.

Part II

Model

6 Framework implementation

To support the PoR in its goal of renewing its fleet in an environmentally and economically sustainable manner, a decision support framework has been developed. This chapter presents the architecture and methodology of the framework in section 6.1, which encompasses its layered structure, mathematical formulation, and decision support mechanisms. By integrating economic and environmental criteria within a hybrid MOO and MCDA framework, the framework generates renewal schedules that balance TCO, lifecycle CO_2 and LP emissions under realistic operational constraints. The verification of the framework is discussed in section 6.2. A range of strategic sustainability pathways and economic scenarios are defined to provide information on possible policy and investment choices, the development of these pathways and scenarios is discussed in section 6.3 and section 6.4. The chapter is concluded with a summary in section 6.5.

6.1. Architecture

The framework is structured into three integrated layers. As discussed in chapter 4, a combination of ε-constraint and TOPSIS allows the generation of non-dominated alternatives, which can then be evaluated based on the preferences of the stakeholders. Before the generation of alternatives, the raw input data is consolidated. This is done in the first layer, referred to as the preprocessing layer, translating the data into structured economic and environmental indicators relevant to the selected strategic pathway or scenario. These indicators include both economic (e.g., fuel cost, CAPEX and OPEX) and environmental indicators (e.g., CTG, GTC and WTW). The second layer contains of the MOO model, which solves the problem using the data provided by the preprocessing layer. It evaluates strategies that simultaneously minimise TCO, lifecycle CO_2 emissions and LP NO_x emissions, using the ε-constraint method to generate the Pareto front of non-dominated solutions. Finally, the third layer comprises the decision support system, which applies the TOPSIS method to rank Pareto-optimal solutions based on stakeholder-defined preferences. This supports transparent and strategic selection among trade-off solutions. An overview of the complete framework is visualised in Figure B.1.

6.1.1. Preprocessing layer

The preprocessing layer is responsible for aggregating, transforming and structuring technical, environmental, and economic input data into a suitable form for optimisation. This is done for the various assets simultaneously.

Figure 6.1 presents the internal architecture of this layer. The blue components represent scenario-specific input for the case study. The input for the different assets is derived from the PoR data. The green components reflect emission factors drawn from empirical data and literature. These are used to calculate the emissions and fuel cost per asset (dark blue). The resulting outputs (orange) form the left-hand side input for the MOO model. For these assets, costs are assumed to be exogenous to strategic choices such as production location or material composition.

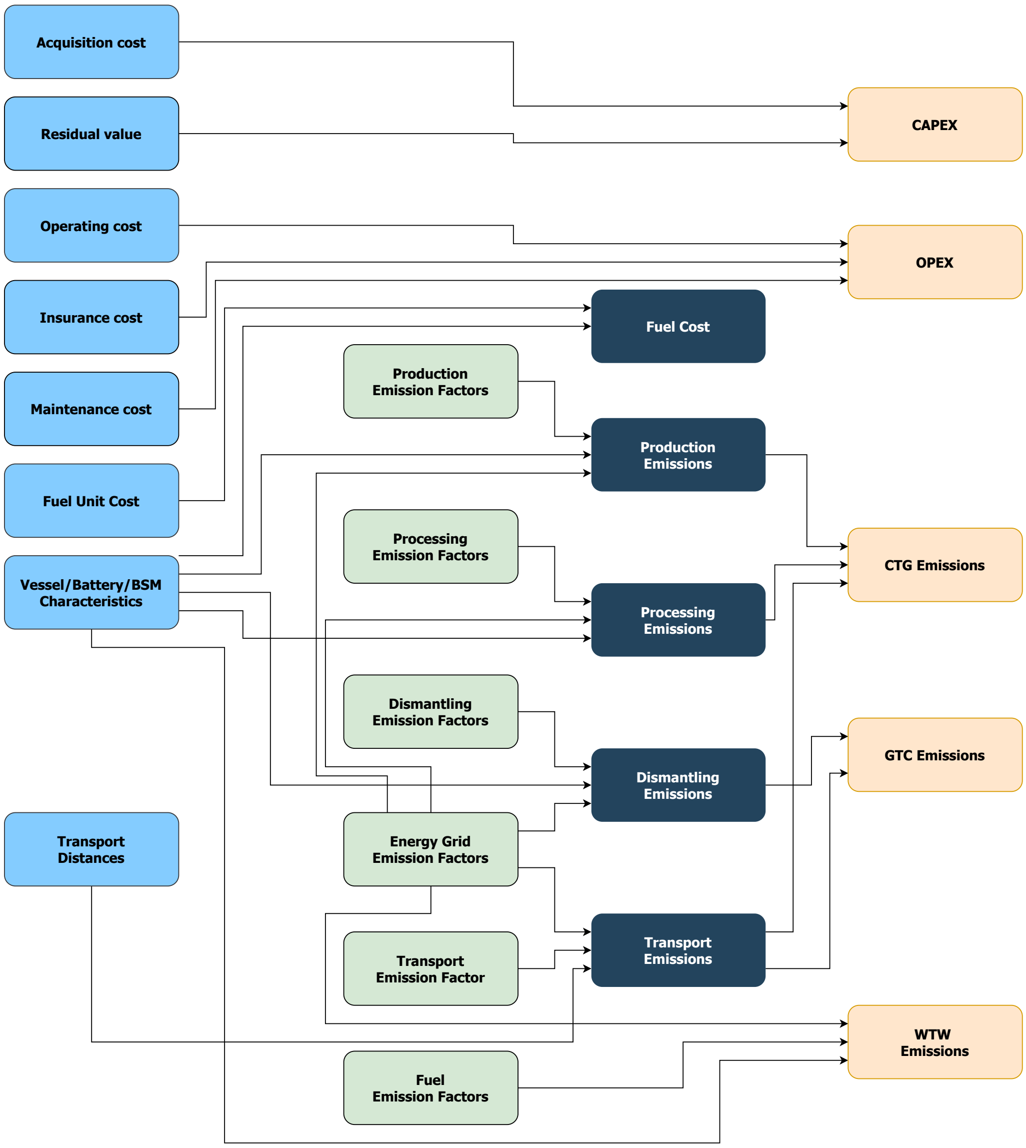
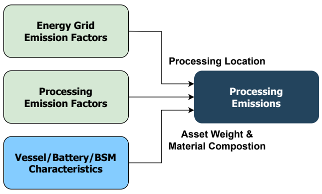


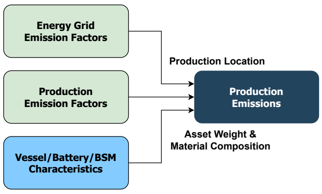
Figure 6.1: Preprocessing layer architecture.

Within this layer, parameters are calculated for vessels, infrastructure, batteries, and energy carriers. Two types of infrastructure are distinguished, BSM and SP. Emissions from BSM infrastructure include crane CTG emissions, whereas emissions from the preparation of the infrastructure location are not included. The associated cost of constructing the charging locations is included. The battery cost parameters apply exclusively to external battery packs used in the IRV, PV, and NM classes. For the sPV and SV vessel classes, internal batteries are considered part of the overall vessel cost, with embedded emissions allocated to the vessel's CTG and GTC emissions.

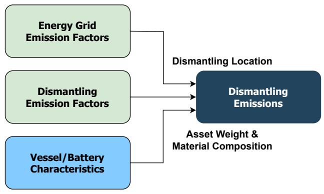
Lifecycle emissions are assessed in three stages, processing, production, and dismantling. Processing emissions result from the extraction and refinement of raw materials. Production emissions reflect the transformation of these materials into final components, and dismantling emissions account for the controlled breakdown of assets to enable material recovery. Each stage includes both direct emissions from physical operations and indirect emissions from electricity consumption, which depends on the regional grid mix. Emissions are calculated separately for vessels, batteries, and BSM infrastructure. The GTC emissions of the BSM are excluded, as its EOL lies beyond the planning horizon. The stages are visualised in Figure 6.2.



(a) Processing factors.



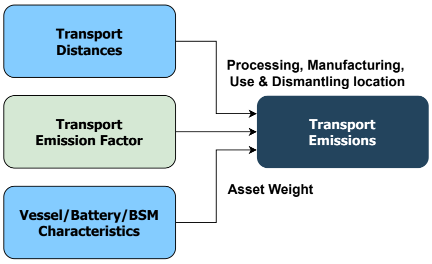
(b) Production factors.



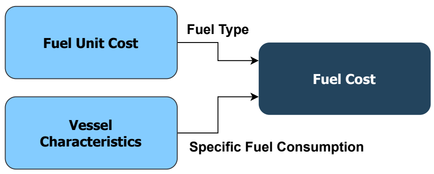
(c) Dismantling factors.

Figure 6.2: Lifecycle emission phases: Processing, production, and dismantling.

Transport emissions are assessed in three stages, the transport of materials to production sites, delivery of finished products to Rotterdam, and post-decommissioning transport of dismantled assets to material recovery locations. The dismantled materials are assumed to be reused locally without further transport. These emissions are calculated based on asset weight, origin-destination distances, and fuel-specific emission factors. The cost of fuel consumption is calculated using vessel-specific fuel consumption and the type of fuel used. This can be conventional marine diesel oil (MDO), HVO, or electricity (grey or green). These processes are represented in Figure 6.3.



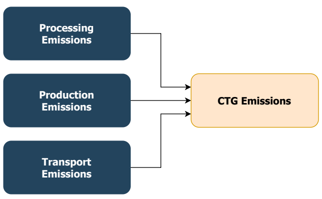
(a) Transport factors.



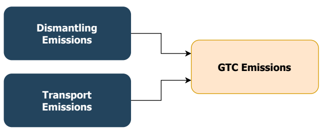
(b) Fuel cost factors.

Figure 6.3: Transport emissions and fuel cost estimation.

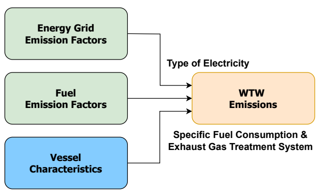
Emissions are categorised into three key lifecycle metrics. CTG emissions are calculated by aggregating the emissions from processing, production, and upstream transport. GTC emissions include dismantling emissions and any post-use transport. WTW emissions cover operational emissions from fuel combustion or electricity consumption and are influenced by the type of fuel and the vessel-specific exhaust gas treatment systems. The respective categories are depicted in Figure 6.4.



(a) Cradle-to-gate factors.



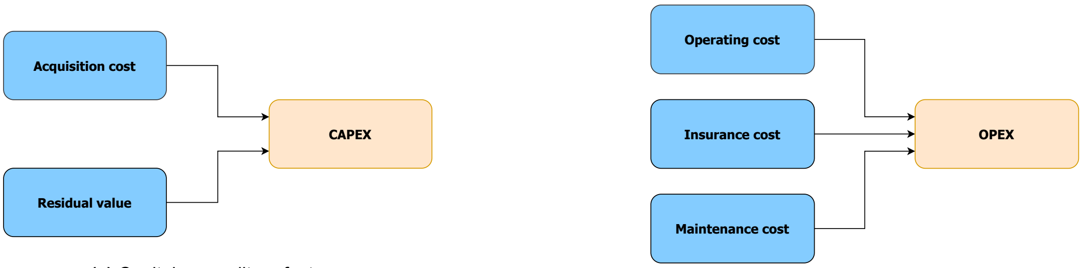
(b) Grave-to-cradle factors.



(c) Well-to-wake factors.

Figure 6.4: Life cycle emission categories.

The economic components of the preprocessing layer consist of calculating CAPEX and OPEX for all assets. The CAPEX consists of the sum of all purchases and salvages of the assets at each time step. The OPEX consists of the sum of the insurance cost of the vessels, the operating cost of the various assets, and the large maintenance of the vessels at each time step. The values are derived from internal historical data and expert assumptions provided by the PoR Asset Management team, responsible for the maintenance of the current fleet. Costs are adjusted for inflation using a cumulative quarterly rate to account for higher prices if purchases occur at later timesteps. The residual value of the company asset at the moment of salvage is determined by depreciating the current book value over time, using the prime-cost method. The cost architecture is shown in Figure 6.5. The fuel cost is kept separate of the OPEX, to differentiate between vessels that are in reserve or operational at different timesteps.



(a) Capital expenditure factors.

(b) Operational expenditure factors.

Figure 6.5: Economic indicators: Capital and operational expenditures.

By consolidating technical specifications, economic parameters, and environmental performance metrics, the preprocessing layer produces a consistent dataset that supports multi-criteria analysis. This ensures that cost and emissions are assessed with full lifecycle awareness and remain comparable across scenarios. The preprocessing layer serves as the foundation of the decision support model. It is responsible for gathering all the required input data and translating it into the economic and environmental indicators necessary for the optimisation layer. This involves a structured transformation of scenario inputs into numerical outputs that represent capital and operational costs, as well as life cycle emissions associated with each asset configuration.

The code of the preprocessing layer is divided into four parts, corresponding to the colours of the blocks. The code is included in Appendix C.

6.1.2. Multi-objective optimisation layer

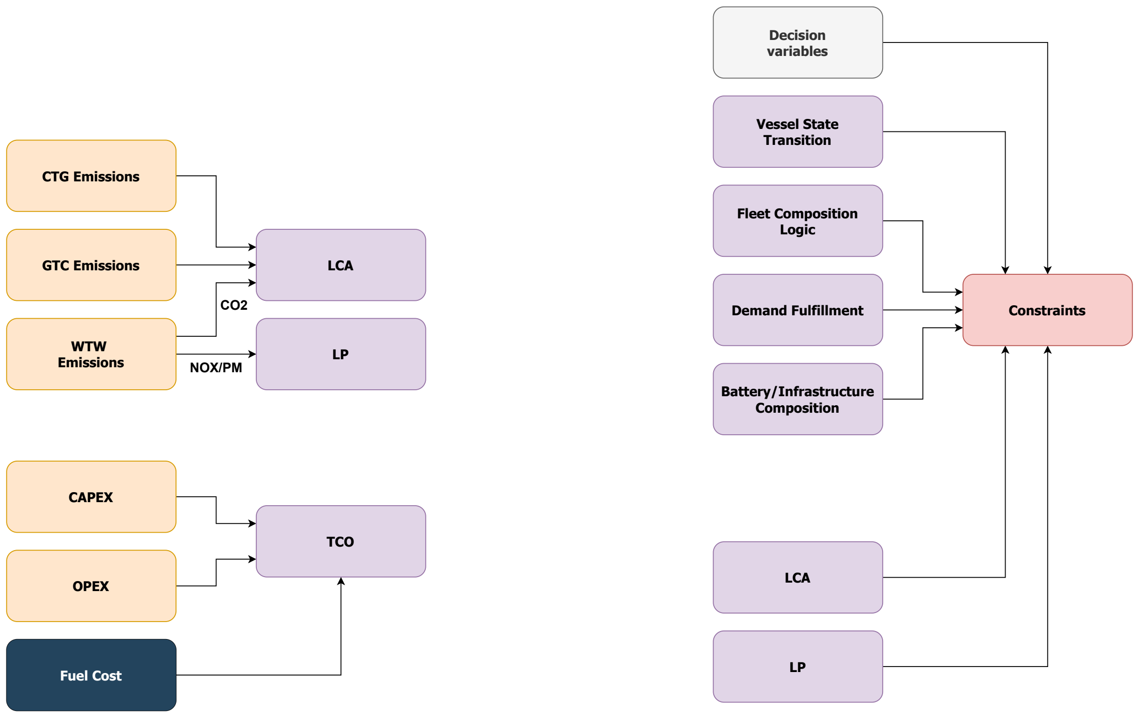
The MOO layer determines the optimal fleet renewal strategy, constrained by the inputs and parameters generated in the preprocessing layer. To reduce computational complexity, the optimisation is performed separately for each vessel class. Although this class-wise separation limits the ability to capture infrastructure synergies between vessel types, it offers significant gains in the required computation time. The inputs provided by the preprocessor layer are illustrated in Figure 6.6a.

To address the conflicting objectives, the ε-constraint method is applied. In this formulation, TCO is optimised as the primary objective, while LCA emissions and LP are incorporated as constraints. The LCA component includes CO_2 emissions in the CTG, GTC and WTW stages. In contrast, the LP criterion reflects the impacts of NO_x and PM. By systematically varying the admissible values of LCA and LP, using thresholds obtained from single-objective baseline runs, the model constructs a Pareto frontier. This frontier captures the trade-offs between economic and environmental performance and enables the identification of efficient, non-dominated solutions for further analysis.

The model formulation incorporates several categories of constraints to ensure that the optimisation reflects operational and technical realities. *Vessel state transition* constraints govern how vessels age, are maintained, and eventually replaced. *Fleet composition* logic enforces the exclusivity rules and guarantees adequate coverage of operational needs. *Demand fulfilment* constraints match the available fleet capacity with service requirements, while *battery composition* constraints ensure that sufficient battery packs are available for uninterrupted operation. Finally, *infrastructure composition* constraints capture the charging requirements for the different vessel classes. The structured input and interaction of these constraints are visualised in Figure 6.6b, illustrating how they collectively define the feasible solution space for the model.

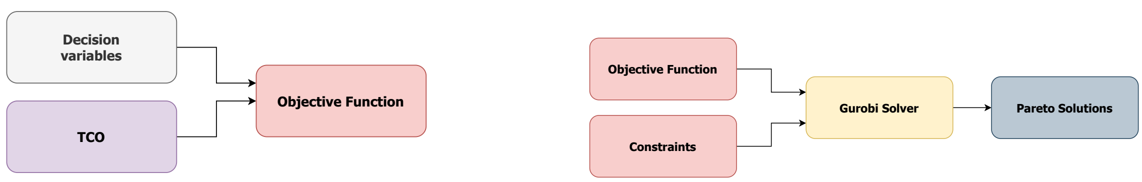
The mathematical implementation of the MOO model is carried out in Python using the Gurobi, as shown in Figure 6.7. Gurobi was selected for its robust performance in the implementation of large-scale MILPs. Gurobi's advanced branch-and-bound algorithms, presolve and warm-start techniques, and multithreaded optimisation capabilities make it well suited to handle the computational demands of the ε-constraint approach, which requires repeated solves under the varying ε-constraint thresholds. Its Python API further facilitates seamless integration with the pre- and post-processing layers of the decision support framework. By looping through the various ε-constraints, each optimisation run delivers a set of class-specific Pareto-optimal solutions, which are then combined and are passed on to the decision support layer for MCDA.

The MOO layer code consists of five scripts as documented in Appendix D. One script sets the correct model configuration depending on the class, a second script determines the class-specific parameters such as the required demand and batteries. The third script builds the Gurobi model. The fourth script combines the previous three scripts and employs them to use the ϵ-constraint method. The final scripts run all the previous scripts and extract the data from the various runs. For the data extraction and visualisation, three support scripts are used, which are documented in Appendix E.



(a) Preprocessed input. **(b)** Multi-objective optimisation constraints.

Figure 6.6: Overview of preprocessed input and constraint structures for the multi-objective optimisation layer.



(a) Objective definition. **(b)** Solver integration.

Figure 6.7: Implementation of the multi-objective model and solver.

Mathematical notation

The mathematical notation of the MOO problem, consisting of the used sets, parameters, and decision variables, is provided in Table 6.1.

Symbol	Description
Sets	
A	Set of vessels
J	Set of possible ages (in quarters)
T	Set of discrete time periods (in quarters)
E	Set of emission types (CO_2, NO_x, PM)
H	Set of infrastructure types (BSM & SP)
Parameters	
$b_{\text{eol\_a}}$	End-of-life age for batteries
$b_{\text{acq\_c}}$	Acquisition cost of battery in time t
$b_{\text{res\_c}}$	Residual value of battery at end-of-life
$b_{\text{om\_c}}$	Operation and maintenance cost of battery in time t
$b_{\text{ctg\_e}}$	Cradle-to-gate GHG emissions of battery asset in time t
$b_{\text{gtc\_e}}$	Grave-to-cradle GHG emissions of battery asset in time t
b_d	Battery demand per vessel
d_{req}	Required number of operational vessels per time period
d_{res}	Required number of reserve vessels per time period
f_c	Fuel cost for vessel i in time t
f_e	Emission vector per unit fuel for vessel i in time t: $[\text{CO}_2, \text{NO}_x, \text{PM}]$
$i_{\text{acq\_c}}$	Infrastructure acquisition cost of type h in time t
$i_{\text{om\_c}}$	Infrastructure maintenance cost of type h in time t
$i_{\text{res\_c}}$	Residual value of infrastructure of type h at end-of-horizon
$i_{\text{ctg\_e}}$	Cradle-to-gate GHG emissions of infrastructure asset of type h in time t
i_{d}	Amount of vessels supported by a single infrastructure asset of type h
$v_{\text{i\_a}}$	Initial age of current vessel a
$v_{\text{eol\_a}}$	End-of-life age for each vessel a
$v_{\text{acq\_c}}$	Acquisition cost of vessel a in time t
$v_{\text{maint\_c}}$	Maintenance cost for vessel a at age j in time t
$v_{\text{res\_c}}$	Residual value of vessel a at age j
$v_{\text{om\_c}}$	Operation cost of vessel a in time t
$v_{\text{ins\_c}}$	Insurance cost of vessel a in time t
$v_{\text{ctg\_e}}$	Cradle-to-gate GHG emissions of vessel a in time t
$v_{\text{gtc\_e}}$	Grave-to-cradle GHG emissions of vessel a in time t
Decision variables	
O_{ajt}	1 if vessel a operates at age j in time t
R_{ajt}	1 if vessel a is in reserve at age j in time t
U_{ajt}	1 if vessel a is under maintenance at age j in time t
S_{ajt}	1 if vessel a is decommissioned at age j in time t
P_{at}	1 if vessel a is purchased in time t
Q_t	Number of new batteries acquired in time t
W_{jt}	Number of batteries used at age j and time t
V_{jt}	Number of batteries in reserve at age j and time t
X_{jt}	Number of salvaged batteries at end-of-life age j in time t
Y_{ht}	Number of infrastructure units of type h acquired in time t
Z_{hjt}	Number of infrastructure units of type h installed of age j in time t

Table 6.1: Model sets, parameters, and decision variables.

Objective functions

The fleet renewal model incorporates three objectives that represent economic and environmental concerns. These are evaluated using the ε-constraint method, with the TCO as the primary objective and the LCA and LP as bounded constraints.

To ensure cost realism, insurance costs are considered annually regardless of vessel operational status, including reserve and maintenance. Similarly, operating expenses are applied to all vessels that are active, in service or in reserve, reflecting ongoing maintenance and other fixed annual costs.

Total cost of ownership: The TCO includes the CAPEX and OPEX for vessels, batteries, and infrastructure, as well as fuel costs.

$$
\begin{aligned}
\min f_1(\mathbf{x}) = & \underbrace{\sum_{a,t} P_{at} \cdot v_{at}^{\text{acq\_c}} - \sum_{a,j,t} S_{ajt} \cdot v_{a,t-j,t}^{\text{res\_c}}}_{\textbf{Vessel CAPEX}} + \underbrace{\sum_{a,j,t} O_{ajt} \cdot \left(v_{at}^{\text{om\_c}} + v_{a,t-j,t}^{\text{ins\_c}} + v_{ajt}^{\text{maint\_c}}\right)}_{\textbf{Vessel OPEX (Operational)}} \\
& + \underbrace{\sum_{a,j,t} R_{ajt} \cdot \left(v_{at}^{\text{om\_c}} + v_{a,t-j,t}^{\text{ins\_c}} + v_{ajt}^{\text{maint\_c}}\right)}_{\textbf{Vessel OPEX (Reserve)}} + \underbrace{\sum_{a,j,t} U_{ajt} \cdot \left(v_{at}^{\text{om\_c}} + v_{a,t-j,t}^{\text{ins\_c}} + v_{ajt}^{\text{maint\_c}}\right)}_{\textbf{Vessel OPEX (Maintenance)}} \\
& + \underbrace{\sum_{t} Q_t \cdot b_t^{\text{acq\_c}} - \sum_{j,t} X_{jt} \cdot b_{t-j,j}^{\text{res\_c}}}_{\textbf{Battery CAPEX}} + \underbrace{\sum_{j,t} W_{jt} \cdot b_t^{\text{om\_c}}}_{\textbf{Battery OPEX}} \\
& + \underbrace{\sum_{h,t} Y_{ht} \cdot i_t^{\text{acq\_c}}}_{\textbf{Infrastructure CAPEX}} + \underbrace{\sum_{h,t} Z_{hjt} \cdot i_t^{\text{om\_c}}}_{\textbf{Infrastructure OPEX}} + \underbrace{\sum_{a,t} O_{sjt} \cdot f_{st}^{\text{c}}}_{\textbf{Fuel Costs}}
\end{aligned}
\tag{6.1}
$$

Life cycle emissions: The LCA objective aggregates CO_2 equivalent emissions from the CTG, GTC, and WTW perspectives for all assets:

$$
\begin{aligned}
\min f_2(\mathbf{x}) = & \underbrace{\sum_{a,t} P_{at} \cdot v_{it}^{\text{ctg\_e}}}_{\textbf{Vessel CTG emissions}} + \underbrace{\sum_{a,t} S_{ajt} \cdot v_{it}^{\text{gtc\_e}}}_{\textbf{Vessel GTC emissions}} + \underbrace{\sum_{a,t} O_{ajt} \cdot f_{i,\text{CO}_2,t}^{\text{e}}}_{\textbf{Operational WTW emissions}} \\
& + \underbrace{\sum_{h,t} Y_{ht} \cdot i_t^{\text{ctg\_e}}}_{\textbf{Infrastructure CTG emissions}} + \underbrace{\sum_{t} Q_t \cdot b_t^{\text{ctg\_e}}}_{\textbf{Battery CTG emissions}} + \underbrace{\sum_{t} X_{jt} \cdot b_t^{\text{gtc\_e}}}_{\textbf{Battery GTC emissions}}
\end{aligned}
\tag{6.2}
$$

Local pollution: The third objective minimises NO_x emissions, which are correlated with PM. Only operational vessels contribute to these emissions:

$$
\min f_3(\mathbf{x}) = \underbrace{\sum_{a,t} O_{ajt} \cdot f_{a,\text{NO}_x}^{\text{e}}}_{\textbf{Operational NO}_\textbf{x}\textbf{ emissions}} \tag{6.3}
$$

Constraints

The optimisation model is governed by a comprehensive set of constraints that ensure logical consistency, physical feasibility, and operational feasibility throughout the vessel fleet, batteries, and infrastructure components. These constraints are grouped into five functional categories: Vessel state transition, fleet composition logic, demand fulfilment, battery composition, and infrastructure composition.

Vessel state transition The fleet evolves through four mutually exclusive operational states: operational (O), reserve (R), maintenance (U), or salvaged (S). Existing vessels are assigned exactly one state in the initial period (Equation 6.4), and all vessels can only be in one state per time step (Equation 6.5). As vessels age, their state transitions across periods (Equation 6.6). New vessels enter at zero age one period after purchase (Equation 6.7).

$$O_{iv_{ia}0} + R_{iv_{ia}0} + U_{iv_{ia}0} + S_{iv_{ia}0} = 1 \quad \forall i \text{ with } v_{ia} \text{ known} \tag{6.4}$$

$$O_{ijt} + R_{ijt} + U_{ijt} + S_{ijt} \leq 1 \quad \forall i, j, t \tag{6.5}$$

$$O_{i,j+1,t+1} + R_{i,j+1,t+1} + U_{i,j+1,t+1} + S_{i,j+1,t+1} = O_{ijt} + R_{ijt} + U_{ijt} + S_{ijt} - S_{ijt} \quad \forall i, j, t \tag{6.6}$$

$$O_{i,0,t+1} + R_{i,0,t+1} + U_{i,0,t+1} + S_{i,0,t+1} = P_{it} \quad \forall i \in I_{\text{new}}, t \tag{6.7}$$

Fleet composition logic Each vessel may be purchased at most once (Equation 6.8) and must be salvaged at most once (Equation 6.9). A new vessel may only be active after its purchase (Equation 6.10). If a vessel reaches a defined maintenance age, it must be maintained or have been salvaged before (Equation 6.11, Equation 6.12). Salvage is enforced when the EOL threshold is reached (Equation 6.13, Equation 6.14).

In these constraints, the symbol τ denotes an index over the set of time periods. It is used as a reference period in the summations, relative to *t*. τ < *t* indicates the summation of all periods prior to *t* Expressions such as τ include the current period as well. When τ appears as a free index, it indicates that the constraint is applied for each possible purchase period.

$$\sum_{t} P_{it} \leq 1 \quad \forall i \in I_{\text{new}} \tag{6.8}$$

$$\sum_{j,t} S_{ijt} \leq 1 \quad \forall i \tag{6.9}$$

$$O_{ijt} + R_{ijt} + U_{ijt} + S_{ijt} \leq \sum_{\tau<t} P_{i\tau} \quad \forall i \in I_{\text{new}}, j, t \tag{6.10}$$

$$S_{ijt} + \sum_{\tau \in T:\tau \leq t} U_{ij\tau} = 1 \quad \text{for maintainable } j, t, \ \forall i \in I_{\text{existing}} \tag{6.11}$$

$$\sum_{t \in T:t \leq \tau} U_{ijt} = P_{i\tau} \quad \text{for maintainable } j, \ \forall i \in I_{\text{new}}, \tau \tag{6.12}$$

$$\sum_{\substack{j,t \\ t \leq T_i^{\text{eol}}}} S_{ijt} = 1 \quad \forall i \in I_{\text{existing}} \tag{6.13}$$

$$\sum_{\substack{j,t \\ t \leq T_{i,\tau}^{\text{eol}}}} S_{ijt} = P_{i\tau} \quad \forall i \in I_{\text{new}}, \tau \tag{6.14}$$

Demand fulfilment To ensure minimum system availability, a minimum number of operational and reserve vessels is required in each period. These constraints can be applied quarterly or annually. In the quarterly form shown below, the operational vessels must exceed d^{req} (Equation 6.15) and the reserve vessels must exceed d^{res} (Equation 6.16).

$$\sum_{i,j} O_{ijt} \geq d^{\text{req}} \quad \forall t \tag{6.15}$$

$$\sum_{i,j} R_{ijt} \geq d^{\text{res}} \quad \forall t \tag{6.16}$$

Battery composition Batteries enter the system at zero age upon purchase (Equation 6.17) and age incrementally over time (Equation 6.18). Battery usage is tied to vessel operations: Operational vessels require two sets of battery packs, as they need to be changed during their shifts. Reserve vessels require one set, because if they become operational and another vessel goes in reserve, the second set of that vessel will be taken over (Equation 6.19). At the EOL, the salvage of battery units is enforced (Equation 6.20).

$$V_{0t} + W_{0t} = Q_t \quad \forall t \tag{6.17}$$

$$V_{jt} + W_{jt} = V_{j-1,t-1} + W_{j-1,t-1} \quad \forall j \geq 1, t \geq 1 \tag{6.18}$$

$$\sum_{j} W_{jt} = \sum_{i \in I_{\text{new}}, j} (2b_d \cdot O_{ijt} + b_d \cdot R_{ijt}) \quad \forall t \tag{6.19}$$

$$X_{b^{\text{eol}},t} = Q_{t-b^{\text{eol}}} \quad \forall t \geq b^{\text{eol}} \tag{6.20}$$

Infrastructure composition The cumulative installed infrastructure must be equal to the sum of previous purchases (Equation 6.21), and must be sufficient to serve operational vessels (Equation 6.22). The ageing of infrastructure is modelled in (Equation 6.23).

$$\sum_{j} Z_{hjt} = \sum_{\tau \leq t} Y_{h\tau} \quad \forall t \tag{6.21}$$

$$\sum_{j} Z_{hjt} \geq i_d \cdot \sum_{i \in I_{\text{new}}, j} O_{ijt} \quad \forall t \tag{6.22}$$

$$Z_{hjt} = Z_{h,j-1,t-1} \quad \forall j \geq 1,\ t \geq 1 \tag{6.23}$$

6.1.3. Decision Support Layer

The decision support layer constitutes the final analytical component of the fleet renewal decision support framework. It interprets the set of non-dominated solutions produced by the MOO layer and selects a recommended strategy that aligns with stakeholder preferences.

Given the trade-off nature of MOO, the results consist of a Pareto front comprising multiple equally optimal solutions. However, from a strategic planning perspective, decision makers require a single actionable recommendation. To bridge this gap, an MCDA framework is employed, as shown in Figure 6.8. This framework enables stakeholders to articulate their relative preferences by assigning weights to each criterion, which are then used to rank and select the most desirable solution from the Pareto front. The script used to perform TOPSIS is supplied in Appendix F.

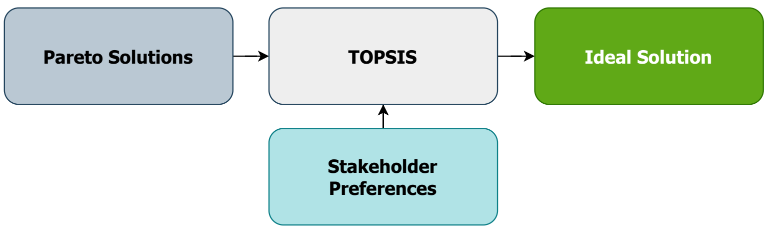


Figure 6.8: Illustration of the TOPSIS decision framework.

TOPSIS methodology

TOPSIS ranks alternatives based on their geometric proximity to an ideal (best-case) solution and their distance from a nadir (worst-case) solution. In this context, each Pareto-optimal solution is evaluated on three criteria: The TCO, the LCA and the LP, all of which need to be minimised.

The layer performs the following steps:

Let $\mathcal{S}$ be the set of Pareto optimal solutions, and let each solution $s \in \mathcal{S}$ have an associated performance vector:

$$\mathbf{y}_s = (f_1(s), f_2(s), f_3(s))$$

where f_1, f_2, and f_3 correspond to LCA, TCO_{res}, and LP, respectively.

Normalisation: Each criterion is normalised using Min-Max scaling to ensure comparability.

$$\hat{f}_j(s) = \frac{f_j^{\text{max}} - f_j(s)}{f_j^{\text{max}} - f_j^{\text{min}}}, \quad j \in \{1, 2, 3\}$$

This transformation reverses the scale, since all criteria are to be minimised. After normalisation, all values are in the interval $[0, 1]$, where higher values are preferred.

Ideal and Nadir solutions: Define the ideal solution as follows:

$$\mathbf{y}^+ = \left(\max_{s \in \mathcal{S}} \hat{f}_1(s), \max_{s \in \mathcal{S}} \hat{f}_2(s), \max_{s \in \mathcal{S}} \hat{f}_3(s) \right)$$

and the nadir solution as:

$$\mathbf{y}^- = \left(\min_{s \in \mathcal{S}} \hat{f}_1(s), \min_{s \in \mathcal{S}} \hat{f}_2(s), \min_{s \in \mathcal{S}} \hat{f}_3(s) \right)$$

Distance calculation: For each solution s, compute the weighted Euclidean distance to the ideal and the nadir:

$$D_s^+ = \sqrt{\sum_{j=1}^{3} w_j \cdot \left(\hat{f}_j(s) - \hat{f}_j^+ \right)^2} \tag{6.24}$$

$$D_s^- = \sqrt{\sum_{j=1}^{3} w_j \cdot \left(\hat{f}_j(s) - \hat{f}_j^- \right)^2} \tag{6.25}$$

where w_j denotes the relative importance (weight) of criterion j, such that:

$$\sum_{j=1}^{3} w_j = 1 \quad \text{and} \quad w_j \geq 0$$

TOPSIS score: The relative proximity to the ideal solution is calculated as follows:

$$C_s = \frac{D_s^-}{D_s^+ + D_s^-}$$

A higher C_s indicates a better compromise solution.

Recommendation: The final decision is the solution with the highest TOPSIS score:

$$s^* = \arg\max_{s \in \mathcal{S}} C_s$$

This selection represents the strategy that best aligns with the preferences provided by the decision maker in terms of climate, economic, and local environmental objectives. The TOPSIS approach provides transparency and repeatability in how these trade-offs are operationalised.

6.2. Model verification

To verify the logical correctness, internal consistency, and reliability of the proposed fleet renewal decision support model, the verification process evaluates all three key layers of the model architecture. In addition, several targeted constraint relaxation tests were conducted to validate the operational integrity and realism of the internal mechanics of the model.

6.2.1. Pre-processing

The preprocessing layer is responsible for translating technical, economic and environmental input data into structured model-ready parameters. To verify this process, emission pathways were cross-validated to ensure that CTG emissions consistently equalled the sum of processing, production, and transport-related emissions. Likewise, GTC emissions were verified to include dismantling emissions and EOL transport impacts. The consistency of WTW emissions was checked with the type of fuel and the energy source of each vessel, adjusting for country-specific electricity grid mixes where appropriate. Unit consistency across emissions, energy, mass, and cost was systematically verified. A verification run was performed for the PV fleet, where the cost and emissions of the assets in the first two timesteps were confirmed by hand calculations. The fleet schedule and asset allocation for this run are provided in Figure G.1 and Figure G.2. The cost verification calculations are provided in Figure G.3 and the emission verification calculations in Figure G.4.

6.2.2. Multi-objective optimisation

The MOO layer was tested by observing the model's behaviour under different fleet compositions, acquisition timings, and emissions profiles. The exclusivity of the states of the vessels was confirmed, ensuring that a vessel could occupy only one state, operational, reserve, under maintenance, or salvaged, at any point in time. Temporal transitions between vessel ages followed the expected logic, with vessels progressing through their lifecycle unless salvaged. No vessel was operated prior to acquisition and no salvaged vessel was reactivated in subsequent time periods.

Demand satisfaction was verified by ensuring that the required number of operational and reserve vessels was maintained consistently throughout the planning horizon. The availability of batteries and infrastructure was always consistent with acquisition schedules. The number of batteries in use never exceeded those acquired and the usage of infrastructure never exceeded the cumulative installations.

The shape of the Pareto front was examined to confirm the existence of trade-offs between TCO, LCA, and LP. As expected, solutions with lower emissions generally involved higher costs, producing a convex Pareto front.

To verify the functionality of the constraints, selected constraints were temporarily disabled or relaxed and compared to a baseline, shown in Figures G.5, G.6, G.7 and G.8, where all constraints were activated. When demand constraints were turned off, the model opted to immediately sell all vessels (Figure G.9), and not acquire new assets. This also resulted in negative costs (Figure G.10) confirming that these constraints governed the size of the fleet.

By disabling battery and infrastructure constraints, no infrastructure or batteries were acquired (Figure G.12). Since battery salvage also no longer is coupled to the acquisition, the maximum battery salvage occurs, limited by variable bounds. Resulting in negative TCO (Figure G.13). This also makes the transition to new vessels more economical, leading to a faster transition (Figure G.11). By disabling salvage and maintenance constraints, fewer vessels are acquired. Furthermore, more vessels are acquired at the the same time steps, as they no longer need to undergo maintenance in the same period (Figure G.14. This also reduces the cost by €100 million (Figure G.15). Finally, disabling ageing constraints shows that the old vessels are no longer in a state at each time-step, while the new vessels can be salvaged continuously at age zero (Figure G.16). The old vessels still maintain the correct ages, while there are gaps in some cases, because the state variables are only defined for realistic time and age combinations, restricting their ageing process (Figure G.17).

6.2.3. Decision support

The decision support layer uses a TOPSIS-based approach to evaluate the set of Pareto-optimal solutions. Verification focused on accuracy scoring and ranking consistency. For test cases, the scoring equation was manually implemented using normalised values in all three objectives. The resulting scores matched those produced by the model, confirming the correct implementation. Figure G.18 shows the hand calculation of the TOPSIS score of two runs with equal weights, with the resulting scores from the Python script provided in Figure G.19.

All calculated scores were limited within the expected range between 0 and 1. When stakeholder weightings were adjusted to favour a particular objective, such as lifecycle emissions, the model responded by selecting cleaner but more expensive fleet configurations. In contrast, assigning 100 percent weight to TCO consistently led the model to select the most cost-effective solution, validating the sensitivity and responsiveness of the ranking mechanism.

6.3. Strategic pathways

To evaluate the influence of upstream and downstream supply chain decisions on fleet sustainability, five strategic pathways are defined. Each pathway represents a coherent scenario composed of interconnected choices regarding geographic locations, hull materials, fuel sources, and circularity strategies. These pathways allow the model to examine the trade-offs between global versus local supply chains, primary versus secondary materials, linear versus circular asset lifecycles, and steel or aluminium vessel hulls.

The first pathway represents a globally distributed configuration with a conventional supply chain. Hull materials are processed in the EU, produced in Vietnam, and dismantled in Turkey. Batteries are produced and dismantled in China. The hull material is basic-oxygen furnace (BOF) steel. Electric vessels use the Dutch mixed grid composition, and conventional vessels use MDO as fuel. This pathway assumes a full linear economy.

The second pathway explores regional manufacturing and dismantling, using primary aluminium as the hull material. The electrical mix for the use phase remains unchanged, while conventional vessels use HVO as fuel. This pathway also assumes a full linear economy.

The third pathway represents a high-sustainability scenario with a local supply chain, clean energy during the use phase, and clean energy during the processing, production, and dismantling of the assets. Secondary aluminium is used for the hull, and batteries are made from new materials. Conventional vessels use HVO as fuel. This pathway assumes a partly circular economy using recycled materials.

The fourth and fifth pathways represent medium-sustainability scenarios with a regional supply chain, where all processes are carried out within Europe. The fourth pathway uses secondary aluminium for the hull, while the fifth pathway uses recycled steel processed by electric arc furnaces (EAF). Vessels use HVO and clean energy, and batteries are produced from new materials.

By evaluating each configuration through the multi-objective model, the analysis offers insights into how strategic design choices impact both economic costs and environmental outcomes.

6.4. Scenario development

To account for financial uncertainty and sensitivity to macroeconomic variables, the model incorporates three economic scenarios: *conservative*, *standard*, and *optimistic*. These scenarios reflect plausible ranges in acquisition costs, residual values, maintenance costs, operating costs, insurance premiums, energy prices, inflation rates, and depreciation rates. Their inclusion enables a robustness analysis of the results under varying economic outlooks, ensuring that recommendations are not overly sensitive to optimistic or pessimistic cost assumptions. The price fluctuations are based on the range that PoR uses in their own cost estimations of 40%.

The *standard* scenario, corresponding to Pathway 4, applies baseline economic parameters that reflect current cost levels and inflation expectations at the time of model development. It assumes a yearly inflation rate of 2% and depreciation rates of 3% for vessels, 8% for batteries, and 2.5% for infrastructure. Fuel and electricity prices are anchored to historical data for 2024.

In the *conservative* scenario, economic conditions are assumed to deteriorate. The cost of new assets and fuel increases by 40%. Inflation is set at 3% annually. The depreciation rates increase by 50%, reflecting a higher asset devaluation.

In contrast, the *optimistic* scenario assumes favourable cost trends. New asset and energy costs are reduced by 40%, and inflation is lowered to 1% annually, indicating a stable price environment. Depreciation rates are halved, reflecting slower asset devaluation and extended economic life.

All adjustments are implemented proportionally to the baseline values for all assets and energy sources. Importantly, economic scenarios do not affect environmental performance assumptions, allowing a decoupled analysis of financial risk versus environmental gain. This enables the model to quantify the trade-offs and co-benefits of pathway selection under realistic financial volatility, offering actionable insights for risk-aware policy and procurement strategy.

In addition to these economic cost variants, a final scenario is introduced to account for environmental discounting practices observed in climate economics and political decision making. The *CO_2 depreciation* retains the economic parameters of the standard case but applies a 4% annual depreciation to CTG and GTC CO_2 emissions. This reflects an emerging approach in literature and policy debates where future emissions are weighted less heavily than the present ones, allowing for differentiated analysis based on time preference in environmental valuation. By including both cost and emission discounting schemes, the model enables consistent sensitivity analysis across perspectives found in both economic planning and climate governance.

6.5. Conclusion

This chapter presented the methodology and implementation of a modular decision support framework tailored for sustainable fleet renewal in the PoR. The model integrates data processing, MOO, and MCDA into a coherent three-layer architecture. The preprocessing layer systematically transforms scenario-specific input into structured economic and environmental indicators, ensuring consistency and comparability across strategic pathways. The optimisation layer applies the ε-constraint method to balance the total cost of ownership, lifecycle CO_2 emissions, and local pollutants, generating a diverse set of non-dominated fleet transition strategies. Gurobi was selected as the solver because of its robustness in handling large-scale MILPs with time-coupled constraints, enabling efficient exploration of the Pareto frontier across multiple vessel classes. Finally, the decision support layer employs the TOPSIS method to translate stakeholder preferences into actionable recommendations, operationalising complex trade-offs into ranked strategies.

By combining rigorous mathematical formulation with transparent decision analysis, the framework provides a practical tool for long-term planning under uncertainty. Its modularity allows for the incorporation of updated data, evolving policy targets, and diverse stakeholder priorities. In addition, the integration of economic scenarios and environmental discounting enables a nuanced assessment of the robustness of different alternatives and preference for time in sustainability decision-making.

In sum, the developed methodology offers a robust platform for structured decision support, enabling the PoR to explore decarbonisation trajectories with both technical precision and strategic flexibility. This foundation supports the input configuration and empirical analysis detailed in chapter 7.

7

Input data

This chapter presents a complete set of quantitative input parameters that underpin the fleet renewal optimisation model. These inputs include technical characteristics, economic assumptions, and environmental factors, all of which influence the outcomes in the strategic and economic scenarios. The input framework adopts a bottom-up approach, incorporating class-differentiated data on vessels, battery systems, infrastructure, and fuel types, to reflect variations in function, size, and energy consumption. Data were sourced from PoR documentation and validated secondary references.

Environmental impacts are evaluated throughout the lifecycle, using emission factors associated with material production, manufacturing, dismantling, and logistics, applied in relation to the mass of the assets and the geographic context. Economic parameters such as CAPEX, OPEX, fuel cost, depreciation, and inflation are included as fixed values and scenario variables, allowing simulation under changing market conditions.

The characteristics and cost assumptions used for the vessels, batteries, infrastructure, and fuel are detailed in section 7.1. The emission factors applied throughout the lifecycle assessment are described in section 7.2. Parameters reflecting long-term strategic planning and macroeconomic uncertainty are provided in section 7.3 and section 7.4.

7.1. Asset characteristics and costs

This section describes the technical and operational parameters used to model the various assets throughout the fleet renewal horizon. It includes both existing and new vessel types across multiple classes, as well as the two types of charging infrastructure, the battery and the fuel aspects.

Vessel class	Battery capacity (MWh)	Battery weight [ton]	Battery demand
IRV	1	5.9	4
PV	1	5.9	4
sPV	2	11.8	1
SV	2	11.8	1
NM	1	5.9	2
RHIB	-	-	-

Table 7.1: Battery capacity and weight per vessel class.

In line with the PoR directives, the new vessels are required to be battery-electric, with the exception of the RHIB. For the sPVs and SVs, the batteries will be placed internally, with the vessels being too small to support the BSM battery packs on deck. For the IRV, PV and NM classes, modular battery packs are used. The weight and capacity of the battery packs for the different vessels class are provided in Table 7.1, as well as the demand in the number of battery packs, depending on the operational requirements. The weight of the battery was estimated on the basis of the total capacity and a specific energy assumption of 170 Wh/kg (Ampherr, n.d.; Fagorederbatt, n.d.). Using the GREET tool, the weight is calculated to be 5.9 tons per MWh (GREET, 2024).

The economic parameters of the vessels, including acquisition cost, book value, operating cost, and insurance premiums, are summarised in Table 7.2. For new vessels, book values are included according to current PoR practice: At the time of purchase, the book value is set at 75% of the acquisition price and then depreciated at 3% per year, reaching zero after 25 years, which corresponds to the technical EOL. The 25% difference between acquisition price and the book value reflects the premium associated with the highly specific requirements of the PoR, which exceed typical market specifications. The book value of the asset is depreciated over time, to determine the residual value cash flow at the moment of salvage, if occurs within the planning horizon. Annual insurance costs are set at 1% of the acquisition price, consistent with current insurance premiums. Acquisition prices are estimated by the company (Port of Rotterdam, 2024c), while operational costs are based on historical data for the current fleet (Port of Rotterdam, 2024b).

Name	Class	Acquisition cost [€1000]	Book value [€1000]	Operational cost [€1000/quarter]	Insurance cost [€1000/quarter]
RPA 10	IRV	-	1500	20	11
RPA 11	IRV	-	1500	18	11
RPA 12	IRV	-	850	18	8
RPA 13	IRV	-	950	15	8
RPA 14	IRV	-	650	15	7
RPA 15	IRV	-	850	20	8
RPA 16	IRV	-	1250	17	10
RPA 30-35	IRV	24 600	18 400	15	61
RPA 6	PV	-	900	14	3
RPA 7	PV	-	875	16	3
RPA 8	PV	-	3500	16	16
RPA 22-24	PV	21 200	15 900	14	53
RPA 1	sPV	-	450	12	4
RPA 2	sPV	-	500	13	4
RPA 21	sPV	11 200	8430	10	28
SV 1	SV	-	350	11	3
SV 2	SV	-	350	10	3
SV 41-42	SV	10 700	8050	10	27
NM	NM	-	2500	33	16
GM	NM	23 400	17 500	25	59
RPA 5	RHIB	-	250	6	1.5
RPA 25-26	RHIB	600	450	4	1.5

Table 7.2: Vessel economic parameters.

The acquisition cost, book value, and OPEX of the battery system are summarised in Table 7.3. Each battery pack has a book value of €570,000, corresponding to 85% of its acquisition price, the difference reflecting the design for integration with the BSM infrastructure. Battery packs are assumed to have a technical lifetime of 10 years, depreciated at 8% per year, leaving a residual value equal to 5% of the acquisition cost at EOL due to the high content of recoverable materials. The annual OPEX is set at 2.5% of CAPEX, based on data from (National Renewable Energy Laboratory, 2025). The acquisition prices were provided by SHIFTR, the developer of the BSM infrastructure (“SHIFTR”, n.d.). The acquisition cost for the infrastructure is based on estimates from the PoR (Port of Rotterdam, 2024c). Given the specific operational requirements at the PoR, the book value is set at 75% of the acquisition price at the time of purchase, consistent with the approach used for the vessels. The infrastructure is assumed to have a useful life of 30 years, depreciated at 2.5% per year. As the EOL falls beyond the planning horizon, no residual value cash flow is included at the moment of salvage within the framework. The annual OPEX is assumed to be 2.5% of CAPEX, as summarised in Table 7.3.

Component	Acquisition cost (€1000)	Book value (€1000)	OPEX (€1000/quarter)
Battery System			
Battery Pack	670	570	4
Infrastructure Types			
BSM	8110	6082	51
Shore Power	3800	2850	24

Table 7.3: Cost parameters for batteries and infrastructure.

The fuel prices are displayed in Table 7.4. These prices follow from the average bunkering cost that the PoR paid last year for HVO and MDO (Port of Rotterdam, 2024a). The electricity price is derived from Eurostat data for non-household consumers in the Netherlands (Eurostat, 2025).

Fuel Type	Price (€1000 per ton or MWh)
MDO	0.92
HVO	1.27
Electricity	0.24

Table 7.4: Fuel prices.

The transport distances used in the CTG and GTC assessments are provided in Table 7.5. For the transport emissions, a factor of 7.9 grammes CO_2-eq per tonne-km is used, corresponding to data from (Sustainable Ships, 2025). Romania and Vietnam are taken as locations, because of the presence of one of the main Dutch shipbuilders in those countries, namely Damen Shipyards, which has also built part of the current fleet. Turkey is chosen as the location because of its abundance of demolition yards, and Norway has many of the resources and facilities required for the production of batteries.

Route	Distance (km)
Romania–Vietnam–Rotterdam	31,800
China–Rotterdam	19,500
Romania–Rotterdam	6,300
Norway–Rotterdam	1,000
Rotterdam–Turkey	6,000

Table 7.5: Transport distances.

Maintenance parameters were established based on technical service schedules. For new vessels, a large maintenance service is assumed every 10 quarters (2.5 years), in accordance with industry norms and current maintenance intervals. During these services, certifications are also renewed. The RHIB undergoes annual maintenance during its scheduled winter downtime. For the current fleet, the PoR has made a future outlook, depending on the current status of the vessels. For new vessels, it is assumed that the vessels undergo a certification service and a conservation service every five years. The average costs of these two services are taken (Port of Rotterdam, 2025). The maintenance interval and the service cost for the new vessels are summarised per class in Table 7.6, while the ages when the current vessels need to be maintained, together with the corresponding cost, are provided in Table 7.7.

Vessel Name	Class	Maintenance interval (quarters)	Cost (€1000)
RPA 30-35	IRV	10	500
RPA 22-24	PV	10	400
RPA 21	sPV	10	275
SV 41-42	SV	10	275
GM	NM	10	400
RPA 25-26	RHIB	4	50

Table 7.6: Maintenance overview new vessels.

Vessel Name	Class	Vessel age [quarters]	Maintenance cost [€1000]
RPA 10	IRV	98, 107, 120	865, 700, 740
RPA 11	IRV	99, 108, 119, 127	865, 700, 740, 525
RPA 12	IRV	107, 117, 127, 136	825, 540, 650, 615
RPA 13	IRV	109, 119	675, 590
RPA 14	IRV	168, 177	1000, 650
RPA 15	IRV	172, 182	1000, 525
RPA 16	IRV	101, 110, 121, 131	765, 900, 615, 650
RPA 6	PV	81, 91, 99	465, 650, 465
RPA 7	PV	91, 101, 111	650, 465, 650
RPA 8	PV	46, 56, 64, 72	75, 200, 75, 200
RPA 1	sPV	101, 111	750, 465
RPA 2	sPV	98, 110, 123	750, 350, 350
SV 1	SV	88	450
SV 2	SV	85, 95	525, 200
NM	NM	130, 134, 138, 142, 146	450, 450, 450, 450, 450
RPA 5	RHIB	27, 31, 35, 39	50, 50, 50, 50

Table 7.7: Maintenance overview current vessels.

Each vessel is characterised by its name, class, initial age, EOL age, hull weight by material, specific fuel consumption (SFC) and pollutant emissions (NO_x, PM). These are summarised in Table 7.8. The material composition is used to calculate the CTG and GTC emissions, while operational emissions are calculated from the specific fuel usage and pollutant emission factors. For the existing fleet, the hull weights are derived from the lightship weight and the hull composition. The hull weight is assumed to be 40% of the lightship weight for full aluminium vessels, 45% for mixed steel-aluminium configurations and 50% for full steel constructions. For new constructions, weight estimates were provided directly by the PoR.

Fuel consumption for the existing fleet is based on empirical measurements from a TNO study conducted on the operational PoR fleet (TNO, 2020). The reported daily energy consumption (in GJ/day) was converted to MWh using the diesel energy content (43 MJ/kg) and an assumed conventional drivetrain efficiency of 40%. For future battery-electric vessels, an electric drivetrain efficiency of 90% was assumed. The final energy requirement per vessel was then scaled proportionally to the size and mission profile of each class and is given per quarter. NO_x and PM emissions are calculated based on the exhaust gas treatment system, corresponding to the CCR0, CCRI, CCRII and stage V diesel engines (Hulskotte, 2018).

Finally, Table 7.9 defines the different rates that are used to adjust some of the cost and emission parameters over time. The CAPEX, OPEX and fuel cost are adjusted using a fixed quarterly inflation rate. This is done over the full planning horizon, except for the book value and the insurance cost. The book value and insurance cost are no longer adjusted for inflation from the moment of acquisition, therefore, in the case of current assets, it stays completely free of inflation. This is because both costs are dependant on the acquisition price.

Name	Class	Age [quarters]	EOL [quarters]	Steel [ton]	Alu [ton]	SFC [ton/MWh]	NO_x [kg/MWh]	PM [kg/MWh]
RPA 10	IRV	95	130	104	–	28.6	4.57	0.186
RPA 11	IRV	95	134	104	–	28.6	4.57	0.186
RPA 12	IRV	99	146	104	–	28.6	12.6	0.6
RPA 13	IRV	99	126	104	–	28.6	12.6	0.6
RPA 14	IRV	163	186	170	–	44.4	12.6	0.6
RPA 15	IRV	163	190	170	–	44.4	12.6	0.6
RPA 16	IRV	95	138	110	–	28.6	12.6	0.6
RPA 30-35	IRV	New	99	*210*	*105*	*152/240*	–	–
RPA 6	PV	79	106	34	4	22.2	10.05	0.54
RPA 7	PV	83	118	34	4	22.2	10.05	0.54
RPA 8	PV	35	82	–	20	14	4.57	0.186
RPA 22-24	PV	New	99	*190*	*95*	*143/227*	–	–
RPA 1	sPV	95	118	20	2.5	17.5	12.6	0.6
RPA 2	sPV	95	134	20	2.5	17.5	12.6	0.6
RPA 21	sPV	New	99	*70*	*35*	*110/174*	–	–
SV 1	SV	79	98	30	3.8	9.4	7.26	0.23
SV 2	SV	79	102	30	3.8	9.4	7.26	0.23
SV 41-42	SV	New	99	*70*	*35*	*51/80*	–	–
NM	NM	127	146	–	60	25	7.26	0.23
GM	NM	New	99	–	60	135	–	–
RPA 5	RHIB	27	42	–	1.6	1.6	7.26	0.23
RPA 25-26	RHIB	New	42	–	1.6	1.6	4.57	0.186

Table 7.8: Vessel characteristics and specific emissions.

The book value of active company assets is depreciated over time using straight-line depreciation rates. Depreciation is applied linearly over an assumed economic lifetime of 25 years for vessels, 10 years for battery systems, and 30 years for infrastructure assets. Depreciation rates are used to determine future residual value cash flows if the asset needs to be salvaged within the planning horizon. Since the EOL of the infrastructure is beyond the planning horizon, no residual value of the infrastructure is included in the TCO. It is however, used to determine the book value of all the active assets at the end of the planning horizon.

CO_2 emissions are assumed to remain constant over time, with the exception of one scenario in which a 4% annual CO_2 depreciation is introduced to reflect time-discounting of emissions, due to technological advancement and grid decarbonisation. This is applied to the CTG and GTC emissions. The inherent emissions due to the combustion process are not depreciated.

Parameter	Rate
Depreciation Rates (per quarter)	
Vessels	0.0075
Batteries	0.0200
Infrastructure	0.00625
CO_2	0
Inflation Adjustment (per quarter)	
Inflation Rate	0.005

Table 7.9: Standard quarterly depreciation and inflation rates.

7.2. Emission Factors

To assess lifecycle emissions, the model incorporates a detailed set of emission factors related to material processing, manufacturing, transport, fuel consumption, and dismantling. These values are used in all stages of the CTG, WTW and GTC lifecycle to estimate the total environmental impact of each renewal pathway of the fleet.

Processing emissions vary by material and production method. Steel is modelled using either the BOF or EAF process, with EAF emissions differentiated by feedstock: recycled scrap or direct reduced iron (DRI). Aluminium inputs are categorised as primary (new) or secondary (recycled). These production routes directly affect both the direct CO_2-equivalent emissions and associated indirect energy use, as shown in Table 7.10.

Steel data is based on global production statistics (*Fact Sheet: The facts about steelmaking*, 2022). The emissions for primary aluminium are taken from Kvande et al. (2022), and for secondary aluminium from Shen and Zhang (2024).

Production-phase emissions, associated with the transformation of raw metals into functional hull components, are assumed to be consistent between materials. The main contributors to energy demand are welding processes and overhead yard systems such as cranes and lighting, following the analysis by Hadžić et al. (2025). These same overhead demands are assumed during dismantling, with energy comparable to that required for reverse welding operations. Emissions from blasting, coating, cutting, and grinding are excluded for dismantling.

Material	Type	Direct CO_2-eq emissions (ton/ton)	Energy (MWh/ton)
Metal processing			
Steel	BOF	1.20	2.22
Steel	DRI–EAF	1.00	0.89
Steel	Scrap–EAF	0.04	0.58
Aluminium	Primary	6.00	14.27
Aluminium	Secondary	0.23	0.03
Metal production			
Steel & Aluminium	—	0.000771	0.566
Metal dismantling			
Steel & Aluminium	—	-	0.518

Table 7.10: Metal emission and energy factors across lifecycle phases.

Battery lifecycle emissions are derived from three primary components. The first is the upstream impact of material sourcing, which is highly dependent on geographic location and mining method. Recycling significantly reduces these material emissions by reusing valuable metals such as lithium and nickel. The second source is the production energy required for the assembly of cells and the integration of modules. These processes are energy-intensive and are assumed to remain constant regardless of the source of raw materials. For both categories, values are based on Kallitsis et al. (2024).

Lastly, EOL emissions arise from dismantling and material separation, which is necessary for battery recycling. This value comes from the results of Li et al. (2023).

Lifecycle Step	Material Source	CO_2 (t/MWh)	Energy
Material contribution	new	56.00	—
Production (assembly)	all	—	60.00 [MWh/MWh]
Dismantling (recycling)	all	—	22.98 [MWh/t]

Table 7.11: Battery lifecycle emissions and energy demand.

Fuel combustion during vessel operation is assessed using static emission factors per tonne of fuel consumed. MDO and HVO are modelled with fixed CO_2-equivalent intensities, while electricity-based emissions vary according to the composition of the regional grid. The applied values are shown in Table 7.12. The emission data for MDO and HVO are based on fuel LCAs, assuming a standard density of 850 kg/L for both fuels (*Brandstoffen voertuigen*, 2024).

Fuel Type	CO_2 (t/ton)
MDO	4.04
HVO	0.41
Electricity	N/A

Table 7.12: Fuel emission factors.

Electricity-based emissions are computed using region-specific grid factors that reflect both direct and upstream emissions associated with power generation. These factors, summarised in Table 7.13, include CO_2, NO_x, and PM. The values of the Dutch grid (grey and mixed) are derived from the national inventory of greenhouse gases (*Emissies naar lucht op Nederlands grondgebied*, 2024) and the energy statistics on total power generation (*Hernieuwbare energie in Nederland 2023*, 2024). The global reference values are based on the regional electricity profiles compiled in (*Carbon intensity of electricity generation, 2000 to 2024*, n.d.), corresponding to the locations mentioned in Table 7.5.

Region	CO_2 (t)	NO_x (kg)	PM (kg)
Clean Energy	0.000	0.000	0.000
Netherlands (Grey)	0.536	0.250	0.0056
Netherlands (Mix)	0.328	0.150	0.0034
Worldwide Average	0.473	—	—
EU Average	0.237	—	—
China	0.560	—	—
Turkey	0.470	—	—
Vietnam	0.472	—	—

Table 7.13: Electricity grid emission factors (per MWh).

These emission coefficients allow for a consistent assessment of operational and indirect energy-related emissions throughout all phases of the lifecycle. Their integration into the model ensures that both direct fuel combustion and upstream electricity emissions are aligned with regional energy characteristics and fuel sourcing assumptions. Combined with material and production emissions, they support scenario-based evaluations that are grounded in realistic and regionally differentiated environmental data.

7.3. Strategic pathway input

Each strategic pathway represents a distinct configuration of global sourcing, energy provision, and circularity strategies. The five different pathways are shown in Table 7.14.

Path	Vessel Sites	Battery Sites	Hull material	Energy source
1	Romania processing, Vietnam production, Turkey dismantling	China production, China dismantling	BOF steel	Mixed electricity / Diesel
2	Romania processing, Romania production, Turkey dismantling	Norway production, Norway Dismantling	Primary aluminium	Mixed electricity / HVO
3	Rotterdam processing, Rotterdam production, Rotterdam dismantling	Norway production, Norway dismantling	Secondary aluminium	Green electricity / HVO
4	Romania processing, Rotterdam production, Rotterdam dismantling	Norway production, Norway dismantling	Secondary aluminium	Green electricity / HVO
5	Romania processing, Rotterdam production, Rotterdam dismantling	Norway production, Norway dismantling	Scrap iron EAF	Green electricity / HVO

Table 7.14: Strategic pathways comparing materials, locations and energy source.

Pathway-specific values are assigned for material sourcing, production, dismantling, electricity mix, and transport logistics. Material emissions differ by both type and production method: BOF steel, EAF steel with scrap iron as input, and primary versus secondary aluminium. These are applied to the hull weights of the vessel provided in Table 7.8 and used to calculate the emissions of CTG and GTC.

Transport emissions are the distances determined provided in Table 7.5, which reflect the international flow of raw materials, finished vessels, and dismantled components.

Electricity-related emissions are applied regionally. During vessel operation, the composition of the electricity grid of the Netherlands is used, for either a mixed grid composition or solely clean electricity. In the CTG and GTC phases, emissions are calculated using the grid mix of the relevant country for manufacturing or dismantling. These emission factors are summarised in Table 7.13.

EOL strategies are also integrated per pathway. These include recycling of vessel and battery components. Recycling benefits are modelled by closing the material loop between dismantling and production. This generates a recycling credit in the CTG phase, accounting for avoided emissions from virgin material production.

All economic cost values are kept the same on the strategic pathways. This allows environmental impacts to be isolated and compared on a consistent financial basis. Variations in environmental outcomes arise from differences in the origin of the material, the mix of electricity, and the transport configuration, rather than from changes in expenditure.

Strategic pathways, as defined by these combined settings, provide a structured means to assess how design and supply chain decisions influence lifecycle emissions. They enable a scenario-based evaluation that is grounded in engineering realism and spatially differentiated emission data.

7.4. Scenario Input

In addition to the structural differences captured by the strategic pathways, the model incorporates a set of economic scenarios to evaluate how changes in market conditions affect the financial viability of fleet renewal strategies. These scenarios allow the model to explore a range of inflation trends, capital costs, depreciation assumptions, and energy prices over time.

The standard scenario corresponds to the fourth strategic pathway, reflecting current market expectations. It uses acquisition and operating costs based on the 2024 values and assumes a yearly inflation rate of 2%. Depreciation is applied linearly using the prime cost method, with rates determined by the expected lifetime of each asset class: 25 years for vessels, 10 years for battery systems, and 30 years for infrastructure. These values are listed in Table 7.9.

The conservative scenario assumes adverse financial conditions. Acquisition costs, operating expenses, and insurance premiums increase by 40%, while fuel and electricity prices increase by 40%. Inflation and depreciation rates increase by 50% compared to the standard scenario.

The optimistic scenario explores the opposite end of the spectrum. Here, capital and operating costs are reduced by 40%, energy prices are decreased by 40%, and the inflation and depreciation rates are decreased by 50% compared to the standard scenario.

Finally, a scenario maintains all standard financial inputs but introduces a 4% annual CO_2 depreciation. This adjustment reflects a time preference for emissions, placing greater value on early reductions in line with climate science and emerging policy approaches.

The input values for the different scenarios are provided in Table 7.15.

Parameter	Conservative	Standard	Optimistic	CO_2 depreciation
Vessel acquisition cost	+40%	Base value	-40%	Base value
Residual value	+40%	Base value	-40%	Base value
Operating cost	+40%	Base value	-40%	Base value
Maintenance cost	+40%	Base value	-40%	Base value
Insurance cost	+40%	Base value	-40%	Base value
Energy prices	+40%	Base value	-40%	Base value
Inflation rate (annual)	3%	2%	1%	2%
Vessel depreciation (annual)	4.5%	3%	1.5%	3%
Battery depreciation (annual)	12%	8%	4%	8%
Infra. depreciation (annual)	3.75%	2.5%	1.25%	2.5%
CO_2 Depreciation (annual)	0%	0%	0%	4%

Table 7.15: Economic and CO_2 scenario assumptions.

Each of these scenarios is applied by making proportional adjustments to the baseline inputs found in Table 7.2, Table 7.3, and Table 7.4. By applying these financial variations to a stable environmental baseline, the model supports a robust sensitivity analysis. The combination of strategic and economic scenarios ensures that both the environmental and financial dimensions of fleet renewal are assessed under a wide range of plausible futures.

7.5. Conclusion

This chapter defined the full set of quantitative inputs that underpin the fleet renewal optimisation model, spanning technical, economic, and environmental domains. Data were compiled through a combination of PoR sources, expert consultation, and literature-based approximations, ensuring contextual relevance and engineering realism.

Emissions are accounted for throughout the CTG, WTW, and GTC stages, allowing the model to reflect embedded carbon, operational emissions, and EOL recovery in a consistent framework. Technical asset characteristics, such as battery configuration, vessel hull composition, and infrastructure typology, are defined at a high resolution, supporting both class-specific modelling and full-system accounting.

In addition to static parameters, a set of five strategic pathways and four economic scenarios was defined. Strategic pathways explore how material sourcing, production locations, and energy mixes influence environmental performance. In contrast, economic scenarios isolate the effects of market volatility, financial risk, and time-discounting of emissions. Together, they allow the model to simulate a broad and plausible range of future configurations.

By capturing both spatial differentiation and temporal uncertainty, the input layer ensures that the optimisation results are not only technically sound but also responsive to long-term strategic planning and sustainability goals. These foundations now support the transition from parameter definition to multi-objective trade-off analysis, presented in chapter 8.

8 Results and interpretation

This chapter presents and interprets the results of the fleet renewal framework for five transition pathways, the economic and CO_2 depreciation scenarios, resulting from the input provided in chapter 7. The Pareto fronts for the eight resulting configurations are first individually analysed to explore trade-offs between TCO, lifecycle CO_2-equivalent (LCA) emissions, and NO_x (LP) emissions. For each configuration, the TOPSIS preferred alternative is identified using equal weights ([1,1, 1]), giving each objective the same relative importance. The effect of each configuration on its preferred alternative is then assessed by examining the corresponding fleet transition schedule, emission and cost breakdowns, and the allocation of battery packs and charging infrastructure.

Following the individual analyses, the objective values of all generated alternatives for the various pathways are combined in a single figure, with the same being done to all generated alternatives for the different scenarios. In doing so, general trends between the different configurations can be highlighted and the overall impact of the configurations can be assessed. The pathway results are presented in section 8.1, followed by the scenario results in section 8.2, with general conclusions in section 8.3.

In all Pareto front figures within this chapter, both emission categories are plotted on the axes, with alternative solutions colour-coded by relative TCO. Additional Pareto fronts showing TCO versus LCA or LP are included in Appendix I. The top five TOPSIS-ranked alternatives for each configuration are provided in Appendix J. The fleet transition schedules and cumulative TCO and LCA curves for each TOPSIS preferred alternative are presented within the chapter, while detailed emission and cost breakdowns, as well as battery and infrastructure allocations, are shown in Appendix K.

8.1. Pathway results and analysis

First the Pareto front and TOPSIS preferred alternative of the five pathways are discussed separately, before discussing the combined Pareto fronts of the pathways. The TOPSIS method ranks alternatives by their relative proximity to an ideal solution and distance from a worst-case (nadir) solution. These ideal points are derived from the minimum and maximum observed values across all objectives: The lowest TCO, LCA, and LP form the ideal solution, while the highest values define the nadir solution. A TOPSIS score between 0 and 1 is then calculated using the weighted Euclidean distances between each alternative and both the ideal and the nadir solutions. A TOPSIS score of 1 corresponds to the ideal solution and 0 to the nadir solution. The higher the score that an alternative receives, the higher the overall performance of that alternative under the given weights.

8.1.1. Pathway 1 – Global BOF steel

This subsection presents the results for the first pathway, which represents a globally distributed configuration with a conventional supply chain. In this pathway, the hull materials are processed in the EU, the hull production takes place in Vietnam, and the dismantling of the hull in Turkey. Batteries are produced and dismantled in China. The hull material used is BOF steel. Electric vessels use electricity from the Dutch mixed electricity grid, whereas conventional vessels use MDO as fuel.

The Pareto front in Figure 8.1 shows all non-dominated solutions for the first pathway. The solutions range between TCO values of € 708–726 million, LCA values of 89,000–97,000 t CO_2-eq and LP values of 14,400–140,000 kg NO_x. The Pareto front exhibits a convex shape near the origin, illustrating the trade-off between the LCA and LP objectives. The convex shape shows that it not possible to minimise both objectives simultaneously. The most economical alternatives, coloured green, are located farther away from the convex front, highlighting that there is also a trade-off between minimising emissions or cost. This is also seen if the TCO is placed on the y axis of the Pareto front plot, as a convex Pareto front is displayed between both TCO–LCA and TCO–LP in Figure I.1 & Figure I.2.

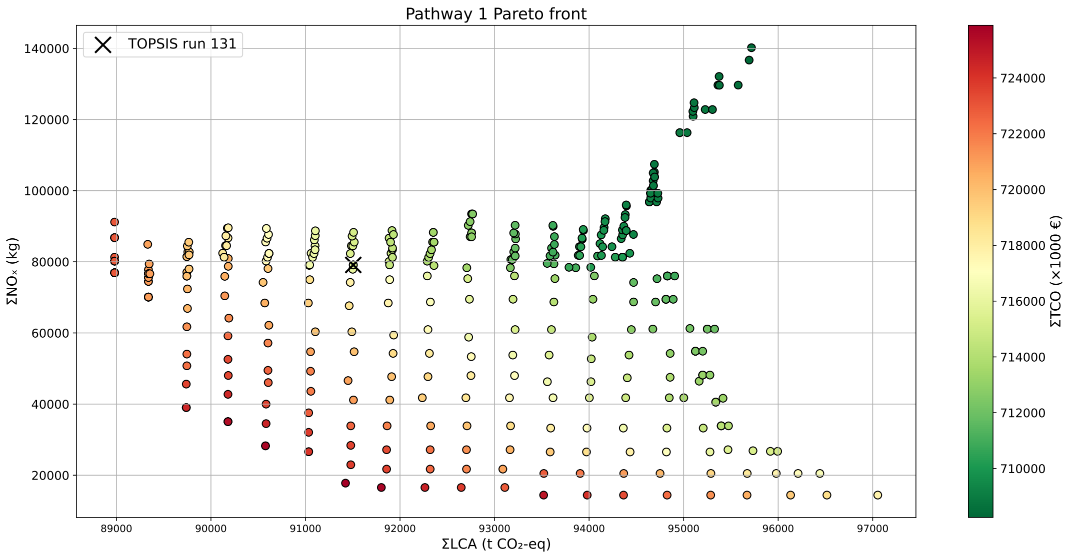


Figure 8.1: Pathway 1: Pareto front.

The TOPSIS preferred alternative is run 131 (Table J.1), with a score of 0.586. The alternative has a TCO of € 715 million, an LCA value of 91,500 t CO_2-eq and a LP value of 79,100 kg NO_x. This places the alternative at the lower end of the LCA emissions, and around mid-way in the TCO and LP ranges.

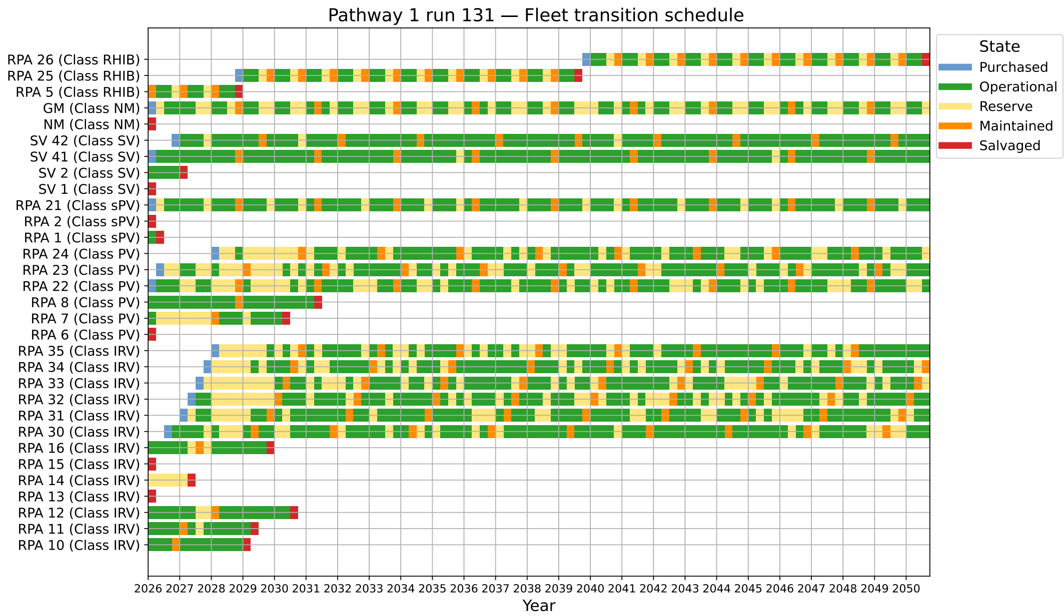


Figure 8.2: Pathway 1 run 131: Fleet schedule.

The fleet transition schedule in Figure 8.2 reveals several key operational dynamics. Conventional vessels of both the IRV and PV class remain operational for the next five years, while new vessels are acquired and placed in reserve. The vessels that are not needed to meet operational demand are salvaged immediately (*RPA 6, 13 & 15*) The conventional vessels of the sPV and SV class are retired either directly at the start (*RPA 1, 2 & SV 1*), in the second quarter (*RPA 1*) or in the first quarter of 2027 (*SV 2*). The *NM* is also retired immediately, while the *RPA 5* is replaced in Q3 of 2028.

The behaviour of the IRV and PV class vessels is mainly emission driven. For the IRV class, the *RPA 14 & 15* have the highest emissions. The CO_2 emissions of the other vessels are the same, but the LP of the *RPA 10 & 11* is lower. For the PV class, the *RPA 8* is the least polluting vessel, while the other two vessels are equally polluting. The decision to retire the *RPA 6* instead of the *RPA 7* is due to the higher maintenance cost for the former. For IRVs, the decision to retire the *RPA 13* instead of the *RPA 12* or *RPA 16* is due to the maintenance timing. The *RPA 13* has the lowest OPEX of the three vessels, but at the time maintenance is scheduled (Q3 2028) there are no battery packs available (Figure K.1), therefore the benefit of the lower OPEX is offset by the impact of additional battery packs that would be necessary. The new vessels are purchased relatively early, before they are all needed in operation. This is done due to the inflated acquisition prices, which outweigh the extra OPEX from early vessel operation. The acquisition is spread out, to ensure that the requirement maintenance is also spread out, and demand can always be met.

For the IRV, PV and NM class, there is a trade-off for the lifecycle CO_2-eq emissions on switching to the battery electric vessels. Switching to battery electric vessels reduces WTW emissions (Figure K.2), however, the battery packs have a high impact on the impact of CTG and GTC emissions (Figure K.4), this leads to conventional vessels still being used in the first five years, after which only two full cycles of batteries are needed, instead of three. The sPV and SV class have internal batteries that do not need to be replaced over the lifetime of the vessel. Therefore, rapid electrification provides huge benefits for emission reduction, both for LP and lifecycle CO_2-eq emissions. The conventional vessels of these two classes have the same emission profile, making the decision which vessel to retire first solely economical. The *SV 1* and *sPV 2* have higher OPEX than their class counterparts, therefore they are retired first. For the NM class, the decision to retire the *NM* is mainly driven due to the high OPEX, as well as the LP reduction. Early salvage leads to the requirement of additional battery packs, which has a negative impact on the LCA and also carries an additional cost. However, these costs are lower than the OPEX benefits. The decision to acquire new RHIBs in 2028 and 2039 is driven by economics and depends on the age and technical EOL, since the new RHIBs will still be conventional and only provide minor benefits in the reduction of LP.

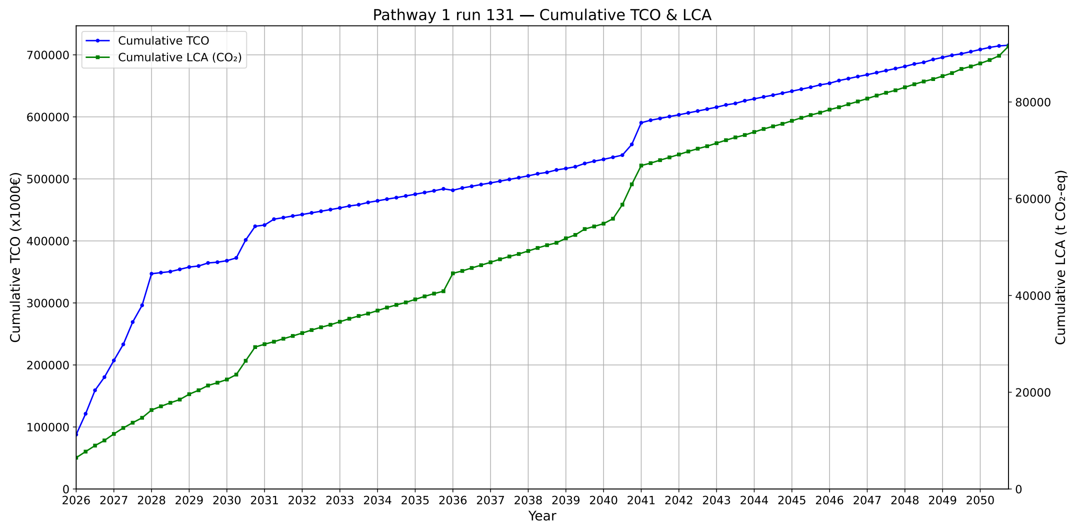


Figure 8.3: Pathway 1 run 131: Cumulative TCO and LCA.

The acquisition of new vessels, as well as battery packs and infrastructure (Figure K.1) results in almost half of the total cost accumulated in the first two years (Figure 8.3). The spikes in the TCO and LCA curves in 2030 and 2040 are due to the acquisition and CTG emissions of new battery packs, with more than 50 batteries being acquired in both of those years, resulting in a total acquisition of 168 battery packs over the planning horizon. The spikes in the LCA and the small drop in the TCO in 2035 are due to the salvage of old battery packs, which still represent a small value at their EOL, resulting in a negative battery CAPEX at that time step (Figure K.3). Other large salvage moments of battery packs between 2040 and 2050 also cause an increase in the LCA, but the total cost at those moments is still positive, which is why there is no visible dip. The constant increase in both LCA and TCO during 2031–2035, 2036–2040, and 2041–2050 is due to the absence of CAPEX and CTG or GTC emissions in these periods. Figure K.2 shows that local pollution for NO_x and PM goes down to zero in 2030, while CO_2 emissions decrease from approximately 800 to 600 t CO_2-eq, showing the benefits of switching from MDO to electric vessels using mixed electricity. At the end of the planning horizon, the book value of the company assets is almost € 89 million (Figure K.3).

8.1.2. Pathway 2 – Regional aluminium and HVO

The second pathway explores regional manufacturing and dismantling, with the hulls manufactured in Romania and dismantled in Turkey. The batteries are produced and dismantled in Norway. For the new vessels, primary aluminium is used as the hull material. The electrical mix for the use phase remains unchanged from the first pathway, while conventional vessels use HVO as fuel.

The Pareto front (Figure 8.4) spans TCO values of € 692-–723 million, LCA values of 52,700-–67,800 t CO_2-eq, and LP values of 14,400-–312,000 kg NO_x. The alternatives still display a convex Pareto front. However, the alternatives are clustered closer to this Pareto front compared to the first pathway, where the TCO dominant alternatives moved further from the Pareto front. This is also visible in Figure I.3 for the TCO–LCA trade-off and in Figure I.4 for the TCO–LP trade-off, where the Pareto front shows a convex front near the origin, but after a certain point a correlation is seen between an increase in both cost and emissions. Although the TCO range has decreased slightly compared to the first pathway, the spread between alternatives has increased. The LCA spread has also grown, but the absolute values have dropped significantly. The LP objective shows a significantly higher upper bound, while the lower bound remains consistent with Pathway 1.

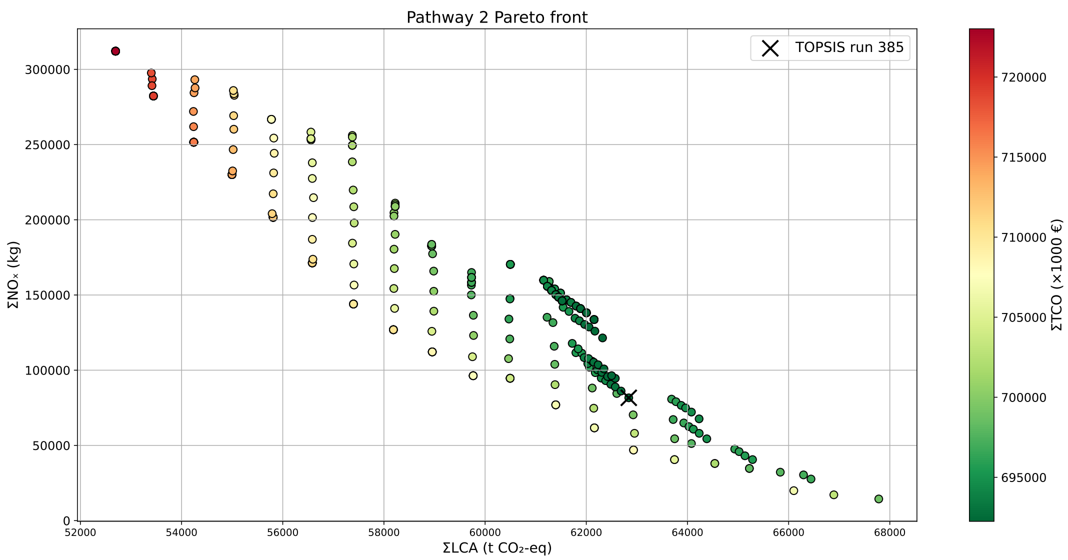


Figure 8.4: Pathway 2: Pareto Front.

Run 385, the preferred alternative (Table J.2), lies at the high end of the LCA range but at the low end of both LP and TCO. Run 385 has a TCO of € 693 million, an LCA of 62,800 t CO_2-eq and a LP of 81,600 kg NO_x, resulting in a score of 0.646. The solution is placed in the lower half of the LP range, above the midrange for the LCA objective, and the lower end of the TCO range.

The difference in the Pareto fronts becomes clear when examining the WTW emissions and the fleet transition schedule. After the fleet has switched to electric vessels Figure 8.5, the WTW CO_2 emissions increase, while the NO_x and PM emissions go to zero Figure K.6. This is due to conventional vessels using HVO rather than MDO as in Pathway 1. In Pathway 1, the trade-off between the reduction of WTW and increase of CTG and GTC led to a switch to battery electric vessels within the first five years, the LCA dominant solutions, continuing sailing with the conventional vessels until the EOL. This results in the upper bound of the LP emissions doubling and the increased spread in the TCO and LCA, as seen in the Pareto front figures.

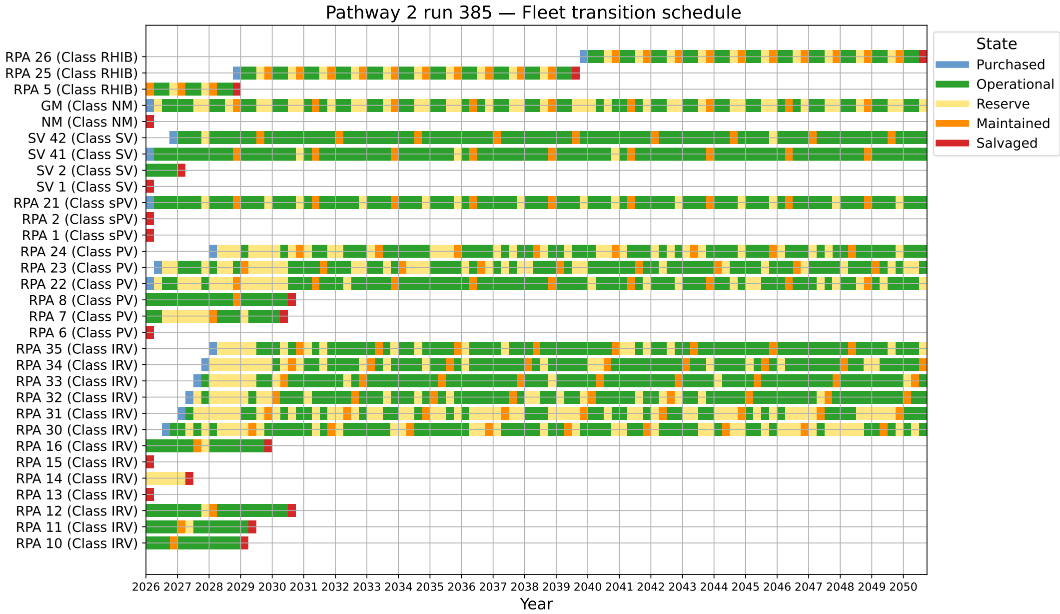


Figure 8.5: Pathway 2 run 385: Fleet schedule.

Figure 8.5 indicates a transition schedule that closely mirrors that of Pathway 1, with only minor deviations. The *RPA 8* and *RPA 1* are retired slightly earlier, which reduces LP, OPEX and provides a higher salvage value. The conventional IRVs have a higher utilisation rate, while they are retired at the same moment. Battery pack acquisition is unchanged, with 168 packs acquired (Figure K.5).

The cumulative TCO and LCA curves (Figure 8.6) are also similar to those of Pathway 1, with the same heavy CAPEX in the first two years, and the acquisition and salvage moments of the battery packs seen clearly in both curves in 2030, 2035, 2040 and 2050. At the end of the planning horizon, the assets represent a book value of €88 million (Figure K.7). The decrease in book value compared to the first pathway is mainly due to the infrastructure being purchased at earlier time steps, resulting in an additional depreciation of the book value. This results from the earlier retirement of the *RPA 1 & 8* which results in the infrastructure being required earlier. Figure K.6 shows that the WTW emissions of the electric fleet are lower compared to Pathway 1, due to the use of lighter aluminium vessels requiring less electricity. This also reduces the fuel cost (Figure K.7), and explains the decreased lower TCO bound. The CTG impact of the vessels has, however, increased because the production process of primary aluminium is more energy intensive (Figure K.8).

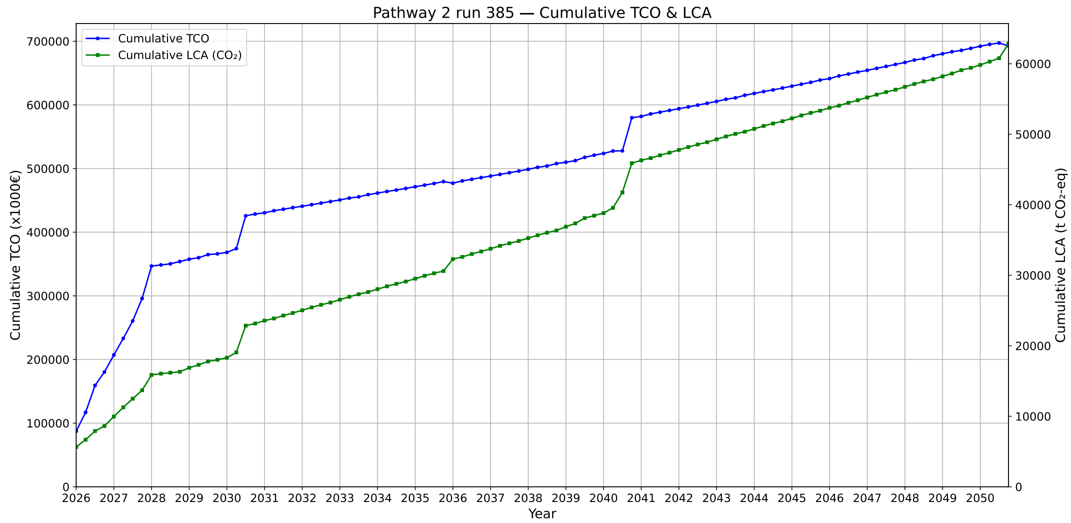


Figure 8.6: Pathway 2 run 385: Cumulative TCO and LCA.

8.1.3. Pathway 3 – Local supply chain and decarbonised electricity grid

Pathway 3 represents a high-sustainability scenario. This pathway features a local supply chain and the use of clean energy throughout the asset lifecycle (processing, production, use, and dismantling). Secondary (recycled) aluminium is used for the hull, while batteries are made from new materials. Conventional vessels use HVO as fuel.

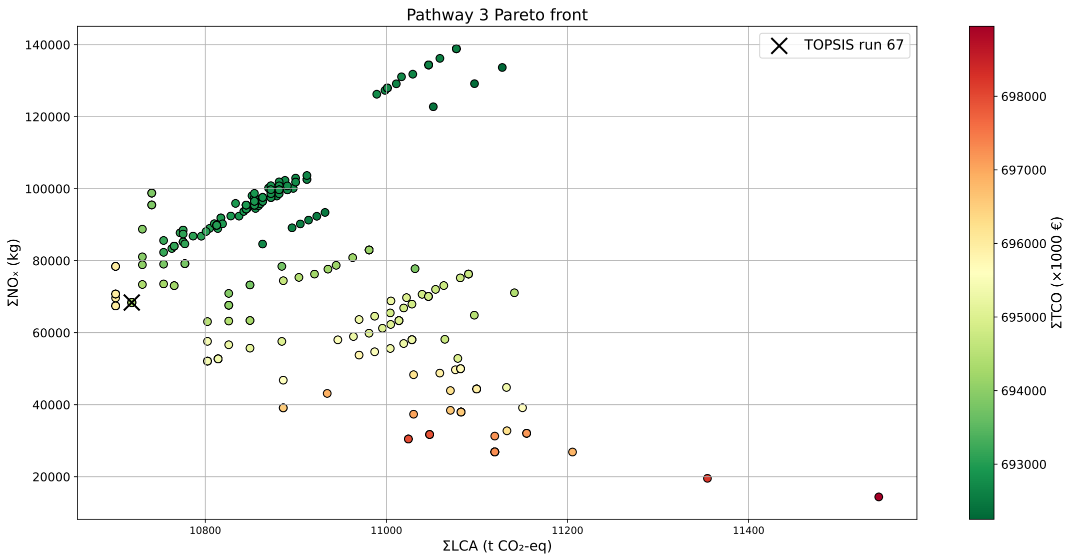


Figure 8.7: Pathway 3: Pareto Front.

The Pareto front (Figure 8.7) shows a highly compact set of solutions for the LCA objective, with values narrowly ranging between 10,700–11,500 t CO_2-eq. This tight clustering reflects the clean energy used in the production, use, and dismantling phases. In contrast, NO_x emissions still span a wide range (14,400-–139,000 kg NO_x), similar to Pathway 1, due to ongoing combustion-related emissions from conventional vessels early in the timeline. TCO values in Pathway 3 range from € 692–699 million. The lower bound aligns with Pathway 2, attributable to aluminium hulls, which reduce energy demand and fuel costs. However, unlike Pathway 2, the upper bound is significantly lower, since extended use of the conventional fleet no longer yields LCA benefits, making such alternatives non-Pareto-optimal.

The preferred alternative, run 67 (Table J.3), has a TCO of € 694 million, an LCA of 10,700 t CO_2-eq and LP of 68,400 kg NO_x, resulting in a score of 0.709. The alternative is placed at the bottom of the LCA range and slightly below the midrange for the LP and TCO objectives.

The fleet transition schedule (Figure 8.8) shows three main differences. The *RPA 10* and *RPA 11* undergo additional service to extend their service by a year, while the *RPA 12* is retired earlier. At the same time *RPA 30, 31 & 32* are acquired earlier and used more heavily in the first years. The salvage of the *NM* is delayed until 2028. However, longer use of the *RPA 10, 11 & NM* is not cost-effective in terms of vessel OPEX, due to high maintenance costs. It also causes additional WTW emissions. Extending *RPA 10*, *RPA 11*, and *NM* reduces battery pack demand, creating net cost benefits despite higher OPEX and WTW emissions. Conversely, earlier *RPA 30, 31, & 32* deployment improves LCA and LP outcomes but increases battery pack needs. In total, these changes cause a slight decrease in the number of battery packs acquired (166) (Figure K.9).

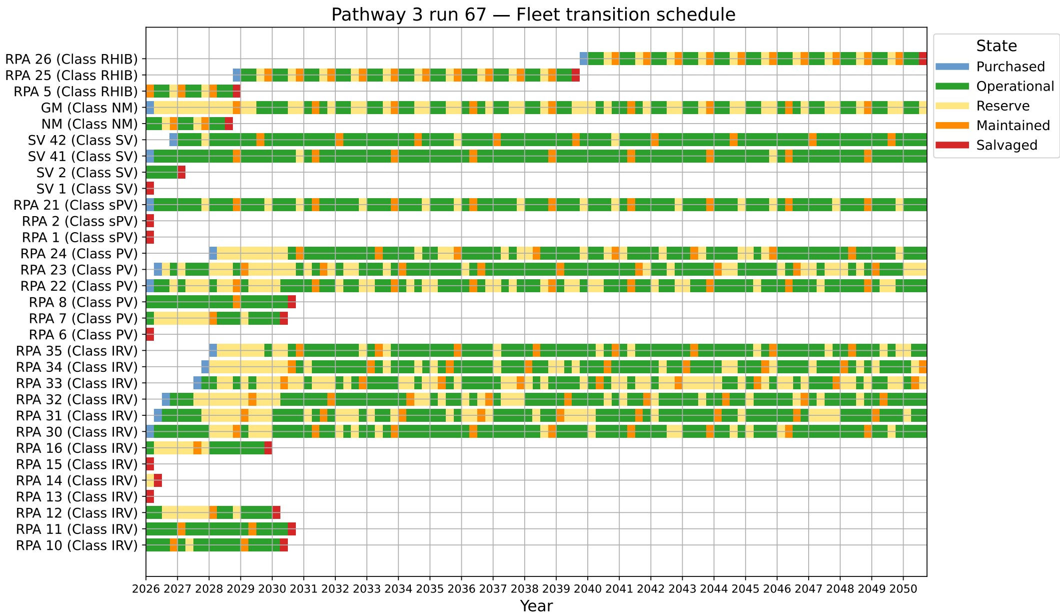


Figure 8.8: Pathway 3 run 67: Fleet schedule.

With indirect CTG and GTC emissions being zero due to the clean energy that is used, the main LCA driver is direct battery CTG emissions, as shown in Figure K.12. The three large steps in the LCA curve coincide with battery acquisition in 2026, 2030 and 2040 (Figure 8.9). The increase in LCA between 2026–2030 is attributable to WTW emissions of the conventional fleet (Figure K.10), which reach zero in 2030. After 2030, LCA emissions remain flat between 2030–2039 and 2041–2050, as there are no WTW emissions and no direct CTG or GTC emissions in that period. The TCO curves show the heavy investments in the first two years due to the CAPEX for the various assets (Figure K.11). In 2028–2030, the TCO is increasing by various amounts due to the salvage of conventional vessels in some quarters offsetting the OPEX and fuel cost. After 2030, the TCO grows steadily driven solely by OPEX and fuel cost, with the exception of the salvage of battery packs in 2036 and 2050, as well as the combination of salvage and acquisition of battery packs in 2040. At the end of the planning horizon, the assets represent a value of € 83 million (Figure K.11). The lower value of the assets is due to the earlier acquisition of the three IRVs, which increases depreciation and results in a lower book value.

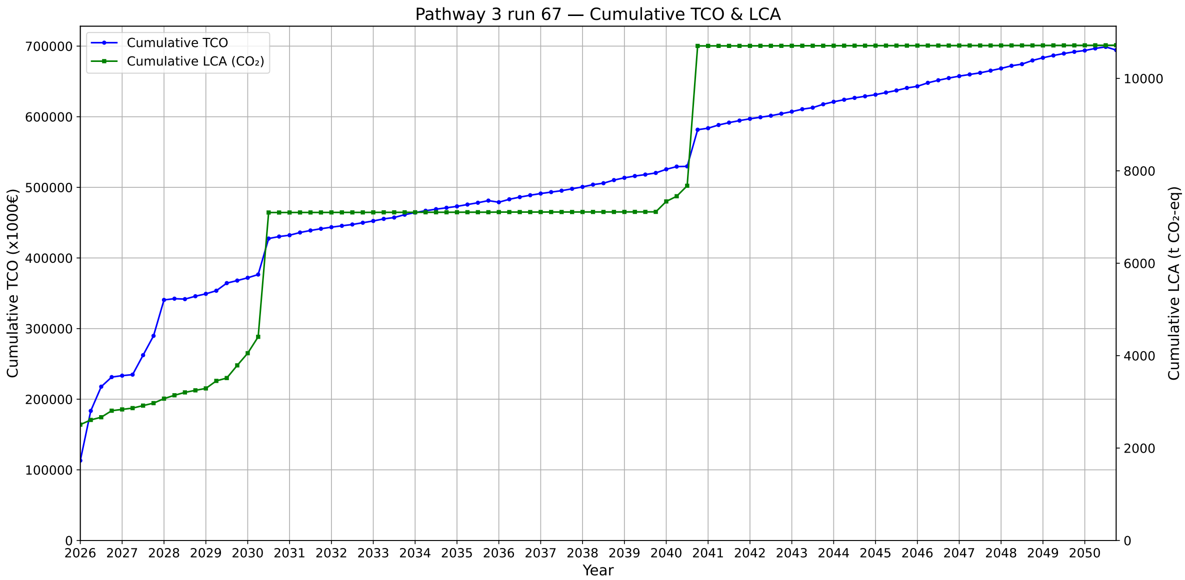


Figure 8.9: Pathway 3 run 67: Cumulative TCO and LCA.

8.1.4. Pathway 4 – Regional Recycled Aluminium

The fourth pathway represents a medium-sustainability scenario with a regional supply chain, where all processes are carried out within the EU. This pathway uses secondary aluminium for the hull and the vessels are sailing on HVO and clean energy. Batteries are produced from new materials. This pathway is also used as the standard scenario for comparison with the economic scenarios in the next section.

In Pathway 2, switching to electric vessels did not benefit the LCA objective, while in Pathway 3 it was highly beneficial due to the use of clean energy throughout the lifecycle. In Pathway 4, switching from HVO to clean electricity reduces WTW emissions but increases CTG and GTC impacts, creating a trade-off similar to that in Pathway 1. This is reflected in the similarity between the Pareto fronts of pathways 4 and 1. This also shows itself in the similarity between the Pareto fronts of Pathway 4 and Pathway 1. The Pareto front (Figure 8.10) shows alternatives ranging between a TCO value of € 692–709 million, an LCA value between 17,100–21,100 t CO_2-eq. and a LP value between 14,400–139,000 kg NO_x.

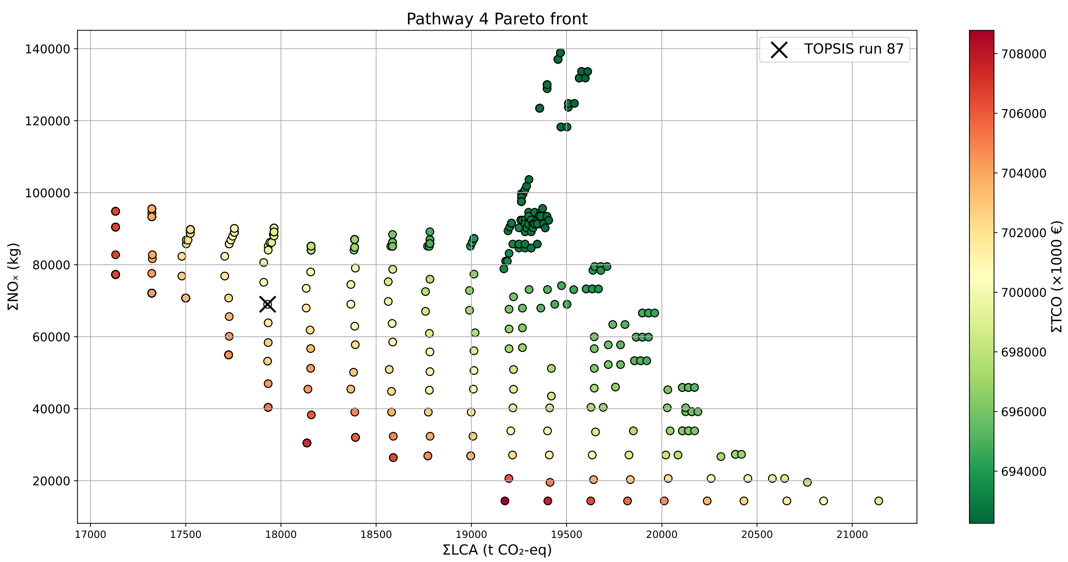


Figure 8.10: Pathway 4: Pareto Front.

Run 87 is the preferred solution (Table J.4), having a TCO of € 700 million, an LCA of 17,900 t CO_2-eq and a LP of 69,000 kg NO_x, with a score of 0.617. The solution lies at the lower end of the LCA range and about midrange for TCO and LP, similar to Pathway 1.

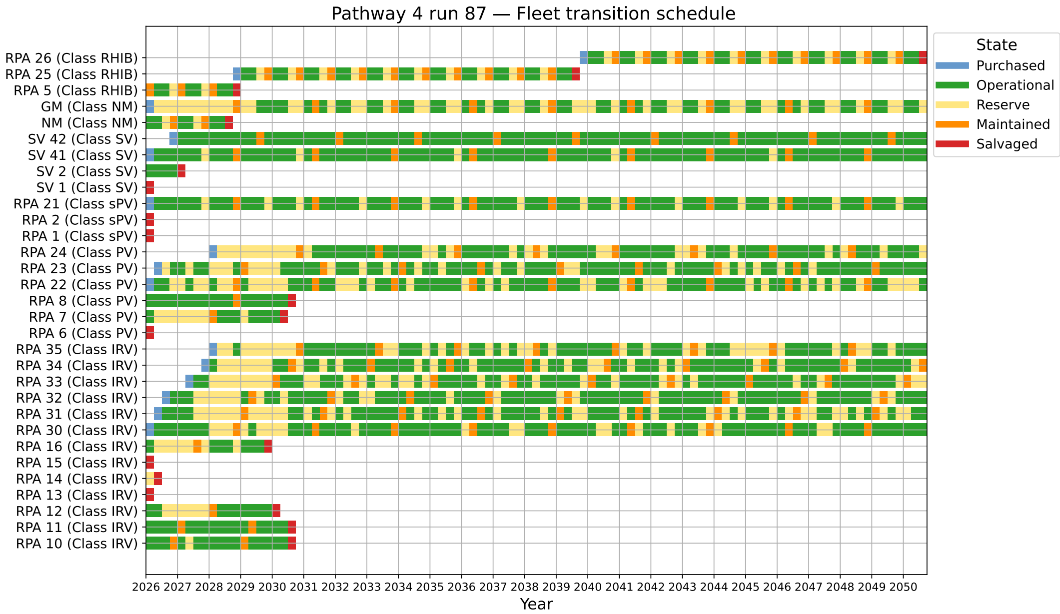


Figure 8.11: Pathway 4 run 87: Fleet schedule.

The transition schedule (Figure 8.11) is similar to the preferred alternative of Pathway 3. The *RPA 33* is acquired one quarter earlier and *RPA 10* is retired one quarter later, while all other vessel timings remain unchanged. With a similar transition schedule, the total number of battery packs is also the same (166) Figure K.13.

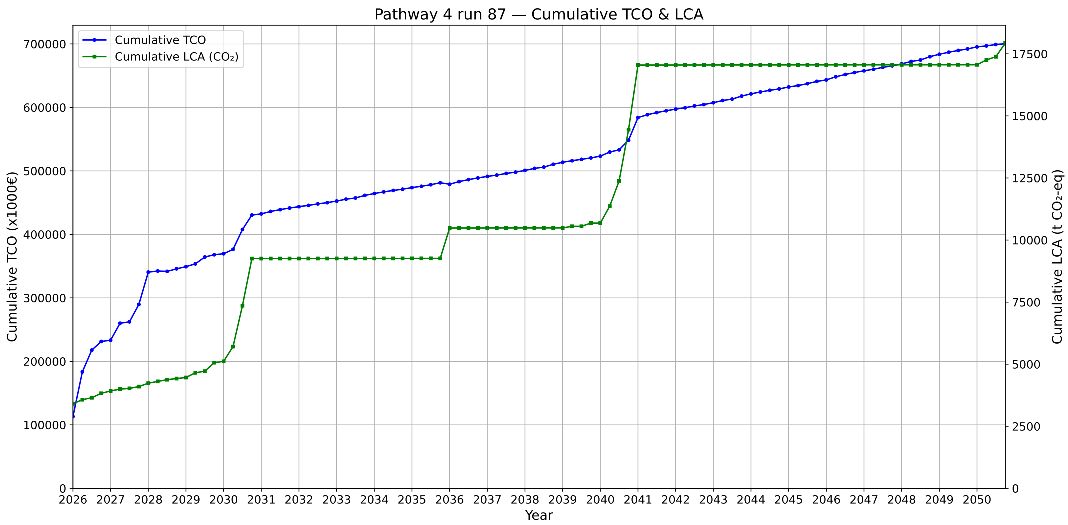


Figure 8.12: Pathway 4 run 87: Cumulative TCO and LCA.

Since the transition schedule and asset allocation are very similar to the previous pathway, cumulative TCO and LCA also behave the same Figure 8.12. Due to the higher impact of CTG and GTC emissions (Figure K.16), the LCA curve steps are larger, while the WTW driven increase in the first four years is less steep due to its lower relative contribution. Salvage events for battery packs are again visible in 2035, 2040, and 2050, now the impact of the GTC emissions has increased. At the end of the horizon, the book value is equal to €83 million (Figure K.15).

8.1.5. Pathway 5 – Regional recycled EAF steel

The fifth pathway uses recycled (scrap) steel processed by EAF for the hull. The other factors are the same as in Pathway 4, with vessels using HVO and clean energy, and batteries are produced from new materials.

This pathway shows the same trade-off between reduced WTW emissions and increased CTG/GTC emissions, producing a Pareto front similar to Pathways 1 and 4 (Figure 8.13). The TCO ranges between € 709–726 million, the LCA between 17,400–21,400 t CO_2-eq and the LP between 14,400–142,000 kg NO_x. The increased TCO range compared to Pathway 4 is mainly due to heavier steel vessels, which consume more electricity and therefore cause an increase in fuel cost. The slight increase in the LCA emissions range is due to the higher carbon footprint of the EAF steel hulls compared to the secondary aluminium hulls of the previous pathway. Since new vessels are more expensive to sail, some alternatives operate the conventional fleet slightly longer, raising the upper limit of LP.

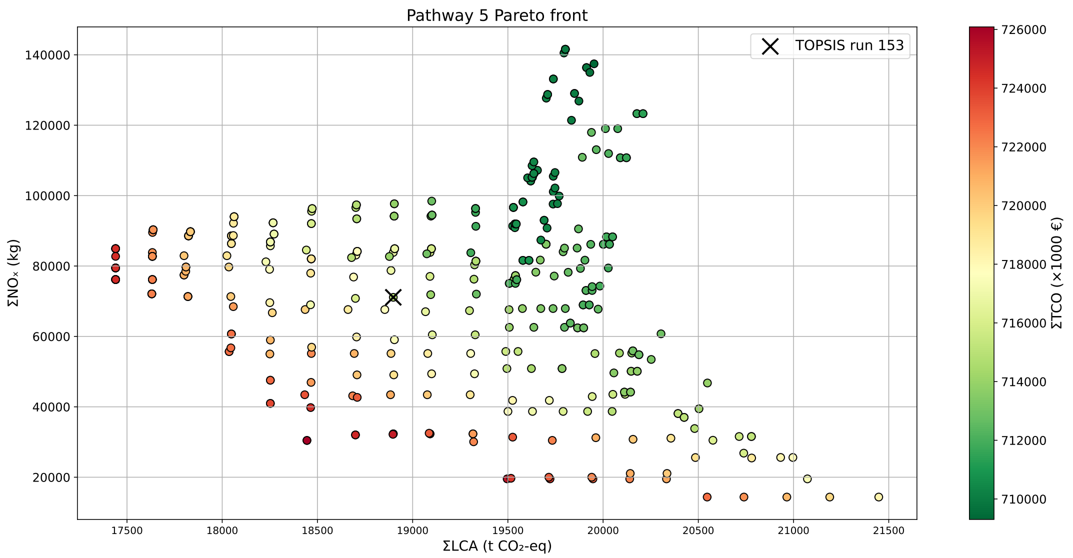


Figure 8.13: Pathway 5: Pareto front.

The preferred alternative is run 153 (Table J.5). Run 153 has a TCO of € 715 million, an LCA of 18,900 t CO_2-eq and a LP of 71,100 kg NO_x, placing the alternative slightly lower than midrange on all three objectives. The resulting TOPSIS score is 0.615.

The fleet transition schedule of run 153 (Figure 8.14) shows a combination of aspects seen in the previous four pathways. The behaviour of the IRV class is similar to the first two pathways, with the *RPA 10 & 11* only receiving one service job and retiring a year earlier compared to the third and fourth pathways. In addition, conventional IRVs are used more until their moment of salvage. This behaviour also leads to the later acquisition of the new IRVs. However, the PV class is similar to the third and fourth pathway with earlier salvage than in Pathway 1. The sPV class shows the same behaviour as in Pathway 1 with the later retirement of *RPA 1*. The SV shows the same behaviour as in all previous four pathways, just as the RHIB. The NM displays the same schedule as in the last two pathways, with salvage happening in 2028. This configuration keeps the total number of battery packs acquired unchanged from the previous two pathways at 166 (Figure K.17).

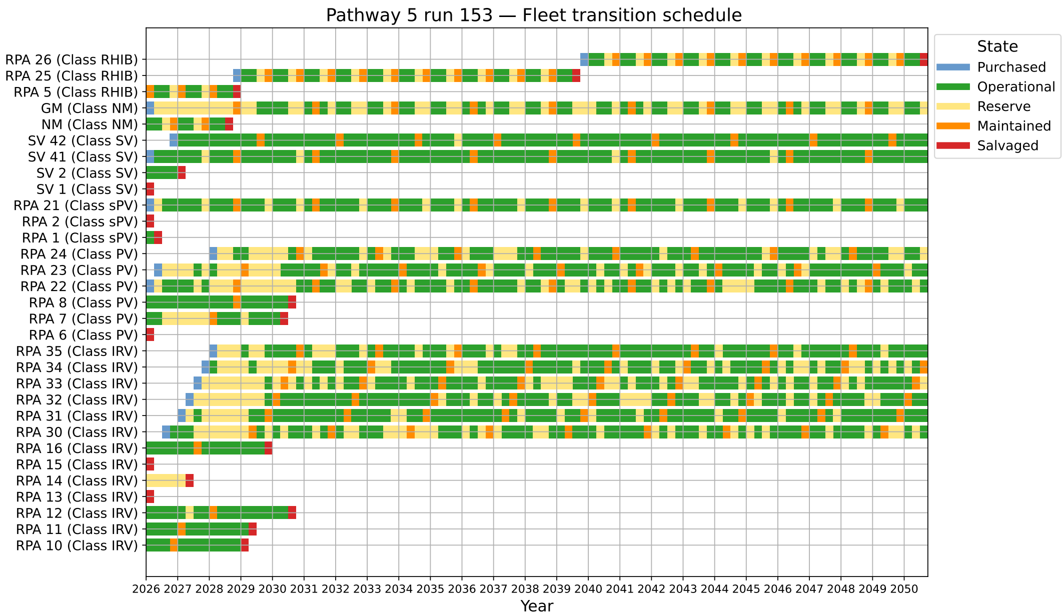


Figure 8.14: Pathway 5 run 153: Fleet schedule.

Figure 8.15 shows a TCO curve similar to the first two pathways, being dominated by IRV CAPEX in the first two years (Figure K.19) and later on by the acquisition and salvage of the battery packs in 2030, 2035, 2040 and 2050 (Figure K.17). The LCA curve mirrors Pathway 4, with minor differences due to slight shifts in the timing of the battery packs. WTW emissions are visible (Figure K.18), after which the LCA remains flat except at battery acquisition and salvage events. By the end of the planning horizon, the assets represent a value of € 84 million (Figure K.19).

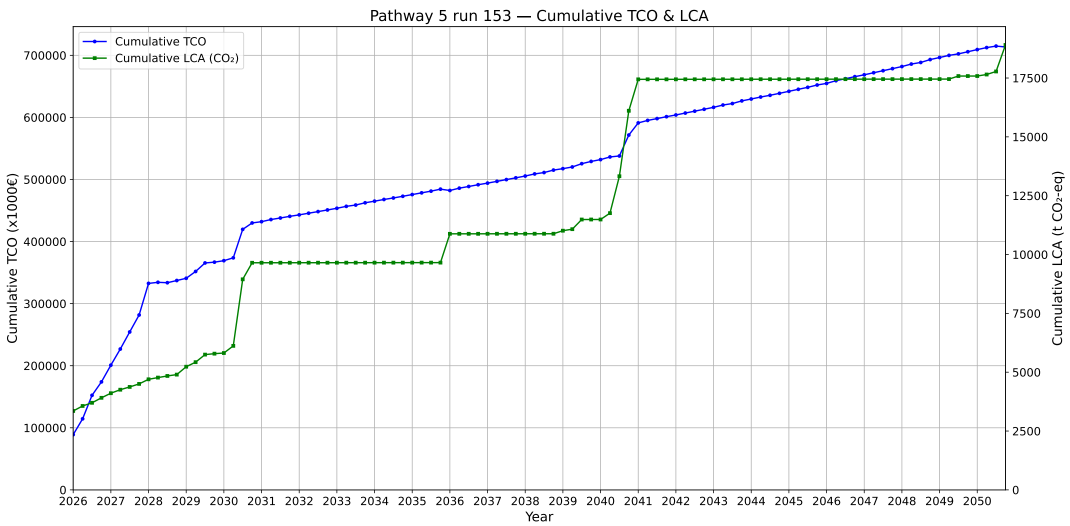


Figure 8.15: Pathway 5 run 153: Cumulative TCO and LCA.

8.1.6. Pathway comparison

The Pareto fronts of all five pathways are shown together in Figure 8.16. Based on this figure and the previous pathway-specific analyses, several key insights emerge.

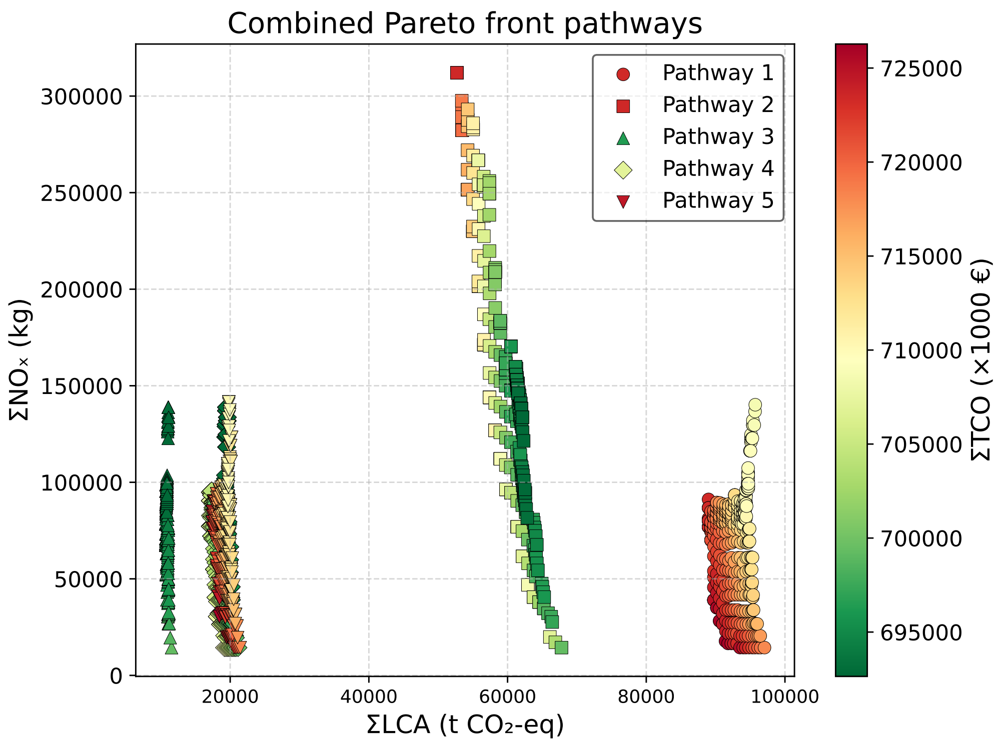


Figure 8.16: Combined Pareto front pathways.

In pathways 1, 3, 4, and 5, electrification leads to a clear net reduction in WTW emissions. Consequently, LP emissions remain within a similar lower range, and a fleet transition occurs well before conventional vessels reach their EOL. In contrast, Pathway 2 minimises LCA by delaying electrification, driven by the high environmental burden of battery production and the increase of WTW emissions after the fleet transition. This delay results in markedly higher LP emissions and a wider spread of TCO outcomes.

Comparison of pathways 4 and 5 highlights the effect of material choice. The use of heavier steel vessels in Pathway 5 increases the upper limit of TCO by approximately € 18 million. Across all pathways, LP varies widely, while LCA remains tightly clustered within each scenario. This suggests that the timing of fleet transitions primarily affects LP, whereas LCA is more strongly influenced by upstream factors.

Despite variations in the configurations, optimisation often produces transition plans that share core structural elements with the preferred alternatives from other pathways. In all scenarios, the RHIB and SV classes follow identical replacement schedules, while the main differences appear in the BSM classes, where the replacement timing shifts in response to material choice, production location, and energy sourcing. Although the transition schedules of the preferred alternatives in the five pathways share these core components, the resulting performance is strongly influenced by the individual pathway configurations.

Finally, differences in LCA ranges reinforce that the largest lifecycle emission reductions come from strategic choices in asset materials, regional production characteristics, and the use of green electricity in manufacturing and dismantling. Given the ten-year lifetime of a battery pack, there is an optimal timing for electrification. This is evident in several pathways where conventional vessels remain in operation after new vessels have been acquired, allowing battery replacements to be minimised.

8.2. Scenario results and analysis

This section evaluates the robustness of fleet renewal in three economic and CO_2 depreciation scenarios. Pathway 4 is used as the reference standard economic scenario. It uses the baseline costs described in chapter 7, with a 2% yearly inflation rate and depreciation rates of 3% for vessels, 8% for batteries, and 2.5% for infrastructure. The conservative scenario assumes higher fuel costs, increased OPEX and CAPEX for new assets, and higher depreciation and inflation rates, while the optimistic scenario assumes the opposite. The CO_2 depreciation scenario applies the same economic parameters as the standard scenario but includes a discount rate on CTG and GTC emissions.

8.2.1. Conservative scenario

The conservative economic scenario assumes worsening economic conditions, with the CAPEX and OPEX of the new assets, as well as fuel prices increasing by 40%. Inflation is set at a 3% yearly rate and the depreciation rates increase by 50%. This subsection presents the results under these economic conditions, offering insights into the financial resilience of the fleet and the impact of higher costs and inflation on sustainability outcomes.

The Pareto front (Figure 8.17) shows more alternatives at the higher end of the LP range compared to the standard scenario (Figure 8.10). These alternatives correspond to the longer use of the conventional fleet, as conventional vessels are the drivers behind LP emissions. These alternatives are also relatively more economical. This is explained by the increasing cost of the new assets, which makes it economically more favourable to continue sailing with the conventional fleet. This is also clearly seen if the TCO is plotted separately against the individual emission objectives. Figure I.12 shows a convex Pareto front between LP and TCO, while Figure I.11 only shows a minor relationship between TCO and LCA. The increased cost is also shown in the TCO range, which has increased by over € 360 million and now ranges between €1,066–1,082 million. The LCA range (17,300-–21,200 t CO_2-eq) and LP range (14,400-–138,000 kg NO_x) remain similar to the standard scenario.

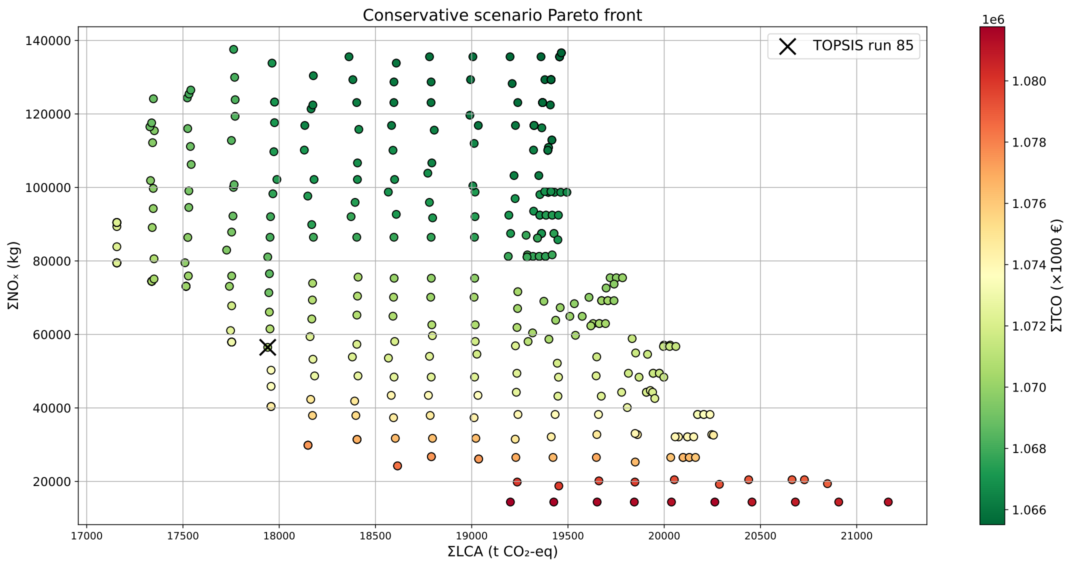


Figure 8.17: Conservative scenario: Pareto front.

The preferred alternative is run 85 (Table J.6). The alternative has a TCO of € 1,071 million, an LCA of 17,900 t CO_2-eq and a LP of 56,500 kg NO_x, resulting in a TOPSIS score of 0.706. The alternative is located at the lower end of the three objectives.

Figure 8.18 shows for the RHIB, NM, SV, sPV, and PV classes mainly the same schedule as in the standard scenario Figure 8.11. For the IRV class, there are two main changes. The *RPA 16* is retired immediately, instead of in 2029, and the purchase of the *RPA 33 & 34* has been moved forward by three quarters. The earlier retirement of the *RPA 16* is most beneficial to reducing LP emissions, which is also noted in the differences between the two preferred alternatives of both scenarios.

The current configuration leads to 56,500 kg of NO_x emissions and the preferred alternative of the standard scenario of 69,000 kg of NO_x. However, due to the earlier retirement of *RPA 16* four additional battery packs are needed. Therefore, the total number of battery pack acquisitions has increased to 170 over the 25-year period (Figure K.21).

In the standard scenario, extending the *NM* reduced LCA impacts by avoiding additional battery packs. Under current economic conditions, where both battery packs and *GM* are more expensive, this extension remains favourable. The same holds true for the earlier acquisition of the IRVs, which would be more expensive due to the increased inflation rate to delay.

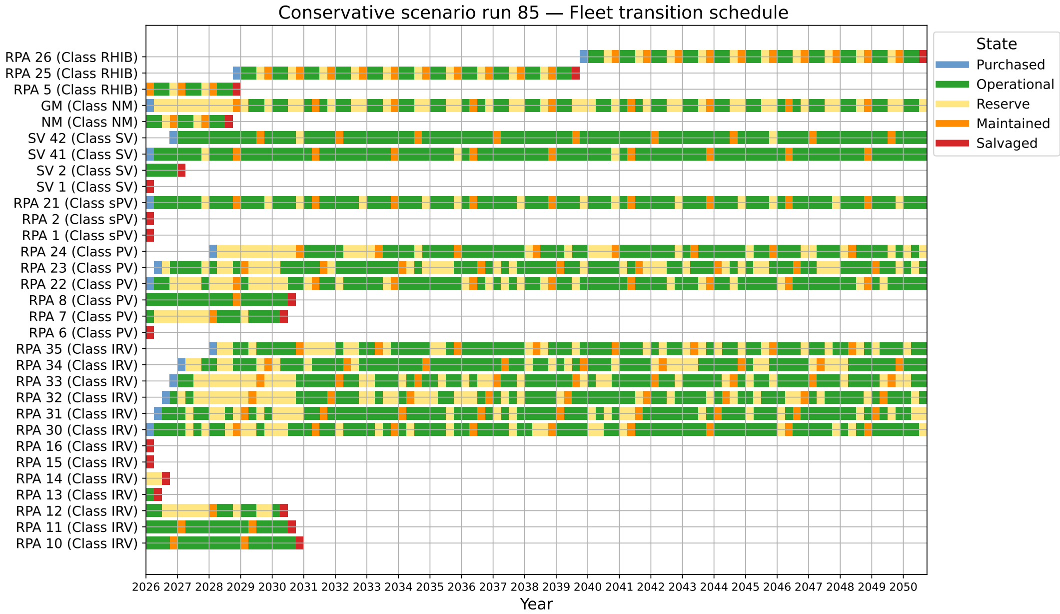


Figure 8.18: Conservative scenario run 85: Fleet schedule.

The cumulative TCO and LCA curves (Figure 8.19) largely mirror those of the standard scenario (Figure 8.12). The earlier IRV purchases shift more CAPEX in the first year, increasing the initial cost share (Figure K.23). Higher depreciation rates lower the residual value of battery packs at EOL, eliminating the TCO drop seen in 2035 under the standard scenario. As a result, the book value of the end of the horizon falls to € 21 million (Figure K.23).

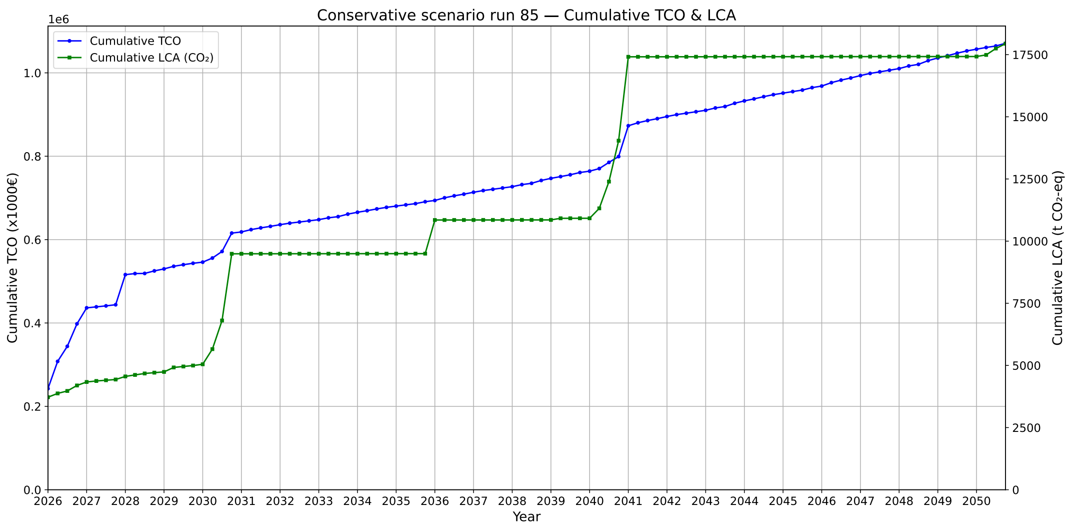


Figure 8.19: Conservative scenario run 85: Cumulative TCO and LCA.

8.2.2. Optimistic scenario

In contrast to the conservative scenario, the optimistic economic scenario assumes favourable cost dynamics. All costs related to new assets and energy prices are reduced by 40%. Inflation is set at 1% per year and the depreciation rates are reduced by 50%. The results presented in this subsection highlight the financial benefits and sustainability gains achievable under optimistic economic conditions, providing a best-case scenario for fleet performance.

Here, Figure 8.20 shows the opposite pattern to the conservative scenario. With more alternatives located at the upper end of the LCA range, and a reduction of the upper bound of the LP range, dropping by 40,000 kg NO_x to 95,000 kg NO_x. This indicates a reduction in alternatives heavily favouring the conventional fleet, and a shift toward earlier electrification, options that occupy the upper end of the LCA range. These alternatives are now more attractive economically, since the reduction in new-asset costs disproportionately benefits strategies that favour a rapid transition. This is also seen in the individual plots between the TCO and emission objectives, with a clear relationship between the TCO and LCA shown in Figure I.13, and almost no correlation between LP and TCO (Figure I.14).

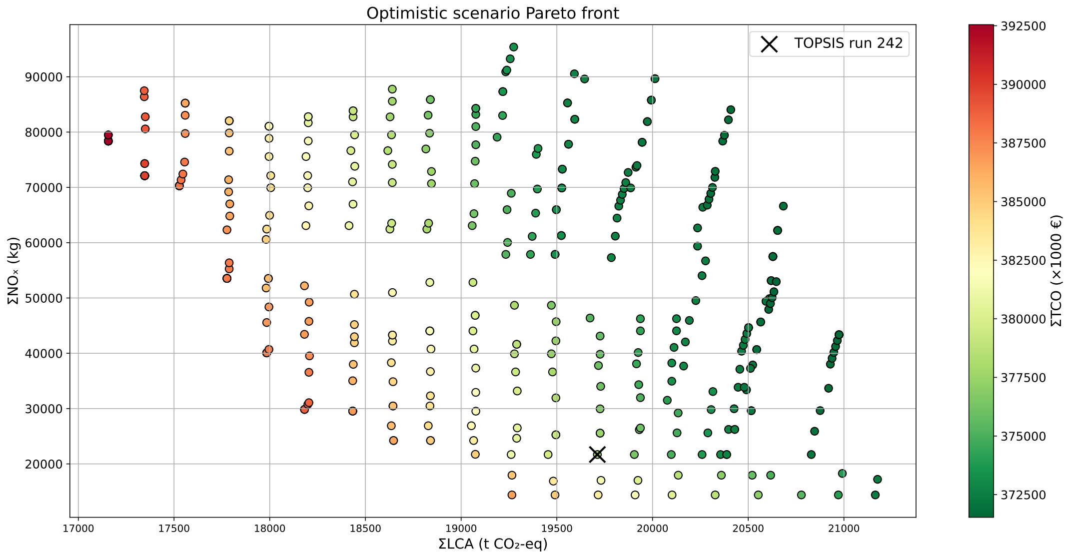


Figure 8.20: Optimistic scenario: Pareto front.

Due to the reduction in costs of new assets, the TCO range has dropped by € 320 million to € 372–393 million. The lower bound of the LP range has remained the same, as in the conservative scenario, at 14,300 kg NO_x. Also the LCA range has remained similar, ranging between 17,200–21,200 t CO_2-eq respectively. With equal TOPSIS weights, run 242 is the preferred alternative (Table J.7), with a TCO of € 378 million, an LCA of 19,700 t CO_2-eq, an LP of 21,700 kg NO_x, and a TOPSIS score of 0.629. This alternative sits above the midrange for LCA while remaining low in both the LP and the TCO ranges.

The fleet transition schedule (Figure 8.21) shows the fastest adoption of new electric vessels among all preferred alternatives. For the NM class, the *NM* is salvaged in the first quarter instead of in 2028. In the PV class, the *RPA 7* is also salvaged in the first year, with only the *RPA 8* retained in service until 2030, although it is increasingly placed in reserve. For the IRV class, all vessels except the *RPA 12* are salvaged at the earliest possible time. With reduced asset costs, the early transition becomes economically more attractive. This approach delivers substantial reductions in LP but requires more battery packs, increasing CTG and GTC emissions. In total, 184 battery packs are acquired (Figure K.25), raising the LCA from 17,900 to 19,700 t CO_2-eq compared to the preferred alternative of the standard scenario. In contrast, LP drops sharply from 69,000 to 21,700 kg of NO_x, illustrating the potential for rapid electrification to minimise local pollution.

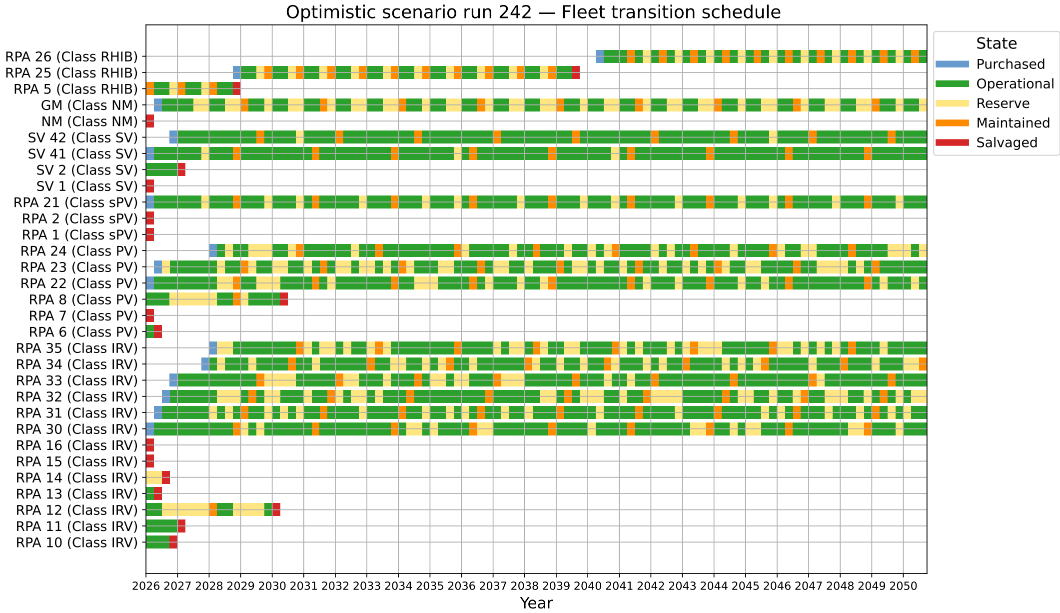


Figure 8.21: Optimistic scenario run 242: Fleet schedule.

Figure 8.22 shows a higher relative share of investment in the first year, driven by the larger number of battery packs acquired (Figure K.25). Combined with the higher residual value of these packs, due to the lower depreciation rate, this amplifies the TCO reductions observed during battery pack salvage events in 2035, 2040, and 2050. Rapid electrification reduces WTW emissions to zero from 2027, except for the three quarters in which the *RPA 12* remains operational in 2028–2030 (Figure K.26). The lower depreciation rate also increases the book value of the assets at the end of the horizon, which reaches € 100 million.

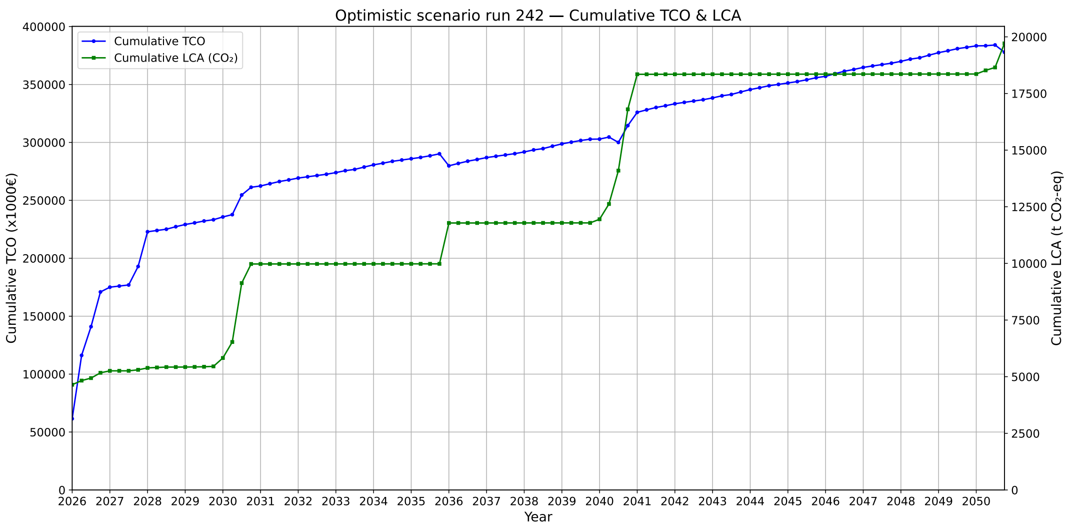


Figure 8.22: Optimistic scenario run 242: Cumulative TCO and LCA.

8.2.3. CO2 depreciation scenario

This subsection introduces a third scenario that accounts for environmental discounting, reflecting the anticipated effects of grid decarbonisation and technological advancements in reducing CTG and GTC emissions. The CO_2 depreciation scenario retains the economic parameters of the standard case but applies a 4% annual depreciation to the CTG and GTC emissions. This mirrors the emerging practice of assigning less weight to future emissions than to present ones.

The Pareto front (Figure 8.23) takes on a distinctly different shape from the other scenarios. Although most similar to the standard scenario, it is shifted toward the origin along the LCA axis, with LCA values reduced to 9,600–13,600 t CO_2-eq due to the applied depreciation. LP ranges from 14,400–134,000 kg NO_x, and TCO ranges from € 692-–749 million, giving a higher upper limit than in the standard scenario. With equal TOPSIS weights, the preferred alternative is run 122 (Table J.8), which has a TCO of € 711 million, an LCA of 10,900 t CO_2-eq, and an LP of 27,100 kg NO_x, resulting in a TOPSIS score of 0.739. This alternative is at the lower end of all three objectives.

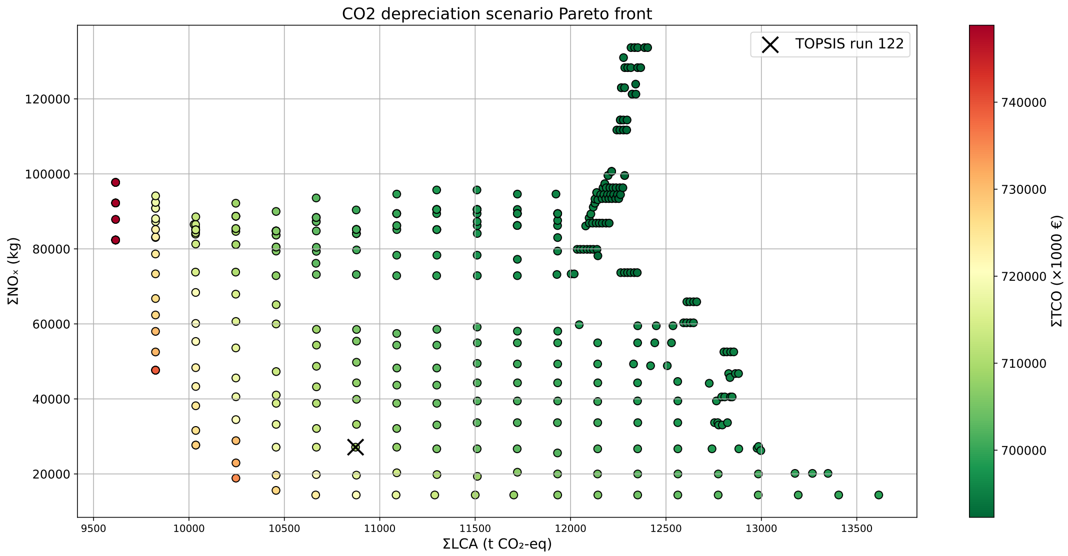


Figure 8.23: CO_2 depreciation scenario: Pareto front.

The fleet schedule (Figure 8.24) shows markedly different behaviour from the standard scenario for most vessel classes, with only the RHIB following the same pattern. The NM and PV classes behave similarly to the standard case, with early retirement of the *NM*, *RPA 6*, and *RPA 7*. The sPV and SV classes also favour early retirement, but the salvage of conventional vessels is delayed. This delay reduces the GTC emissions, but incurs significant additional costs.

For the IRV class, the two least polluting vessels (*RPA 10* and *RPA 11*) remain in service until 2030, while the others are salvaged at the beginning of the process. In general, the schedule favours the early adoption of electric vessels, resulting in the acquisition of 180 battery packs (Figure K.29). Because the CTG and GTC impacts of the battery packs acquired later in the horizon are reduced by the depreciation factor, these early transition schedules now represent the LCA-optimal alternatives. However, since economic parameters remain unchanged, the extra battery packs and the added OPEX from delaying the salvage of the sPV and SV classes make the ideal solution € 10 million more expensive than in the standard scenario. In exchange, LCA is reduced from 17,900 to 10,900 t CO_2-eq, and LP from 69,000 to 27,100 kg NO_x.

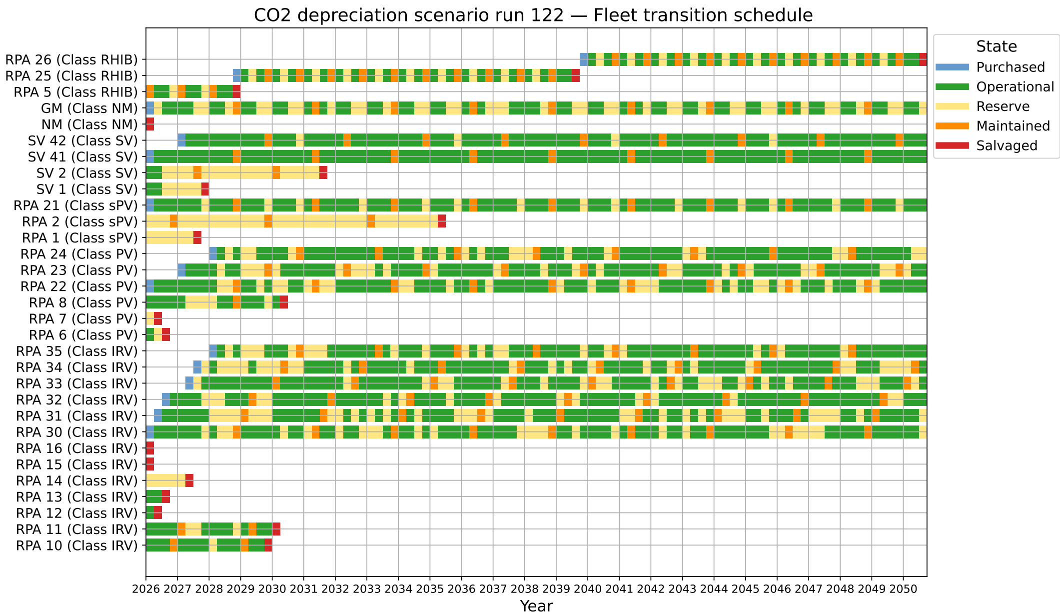


Figure 8.24: CO_2 depreciation scenario run 122: Fleet schedule.

Although in the standard scenario, a large number of battery packs were acquired within the same quarter (Figure K.13), the depreciation of CTG emissions leads to a more staggered acquisition pattern and, consequently, to more staggered salvage events (Figure K.29). This results in the smoother TCO and LCA curves shown in Figure 8.25. The effect of depreciation is also visible in the emission breakdown plot (Figure K.32). WTW emissions drop dramatically in the first year and fall to zero after the salvage of *RPA 11* at the beginning of 2030 (Figure K.30). Because battery packs are acquired later in the planning horizon, they retain a higher book value, with the total book value at the end of the horizon reaching € 95 million (Figure K.31).

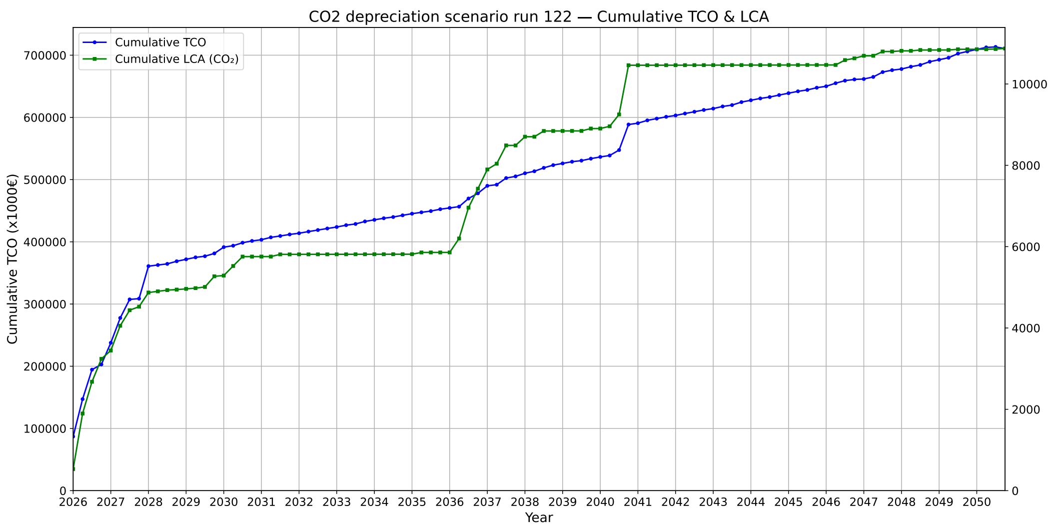


Figure 8.25: CO_2 depreciation scenario run 122: Cumulative TCO and LCA.

8.2.4. Scenario comparison

Figure 8.26 overlays the Pareto fronts of the scenarios discussed above alongside that of Pathway 4. All scenarios contain alternatives within a similar LP range, although in the optimistic scenario they are concentrated slightly lower. This reflects a trend toward faster fleet transitions driven by reduced new-asset costs. The opposite occurs in the conservative scenario, where higher costs make later transitions more attractive, resulting in slightly higher NO_x emissions.

Optimistic alternatives also show slightly higher LCA emissions, a consequence of faster transitions that require more battery acquisitions and, therefore, more CTG and GTC emissions. For optimistic and conservative scenarios, the variation between the alternatives primarily affects LP emissions, with only a reduced effect on LCA. In contrast, the CO_2 depreciation scenario demonstrates that alternative choices can also substantially influence total LCA emissions.

The relative impact of the alternative choice on TCO within a given scenario is small compared to the effect of the broader economic outlook, as evidenced by the limited colour variation within the points of each scenario. This highlights the dominant role of future economic conditions in shaping cost outcomes.

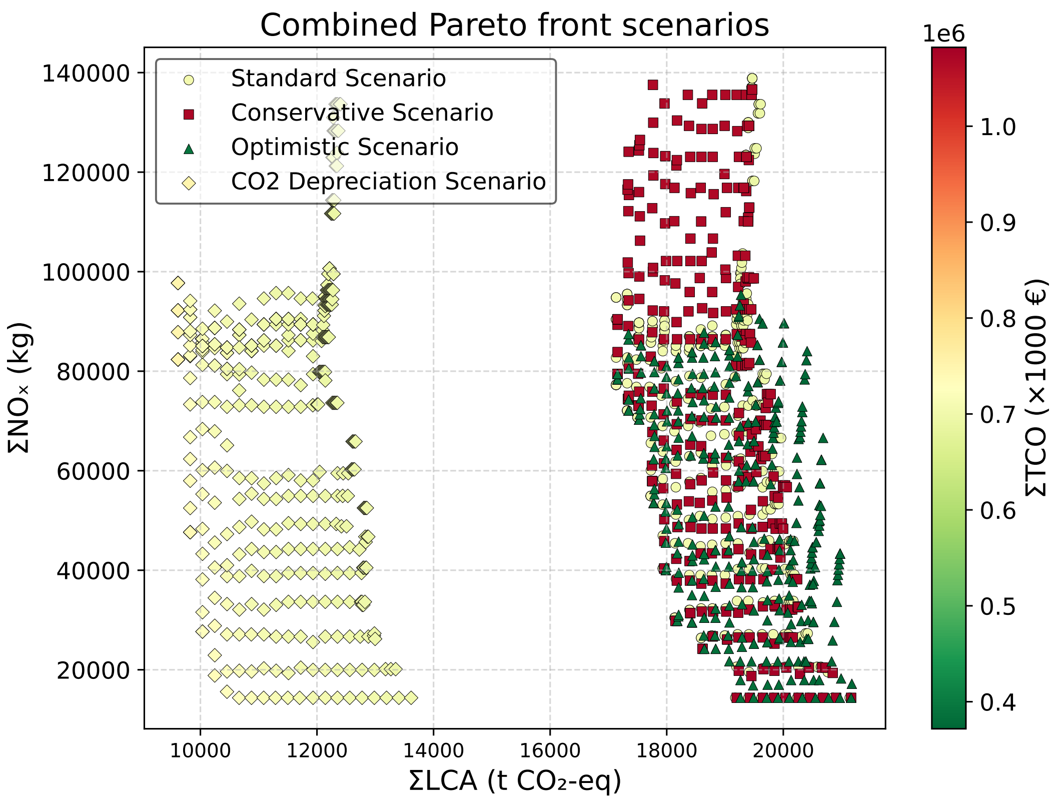


Figure 8.26: Combined Pareto fronts across scenarios.

8.3. Conclusion

This chapter has evaluated a variety of fleet renewal strategies in five transition pathways and four economic scenarios, each representing different assumptions on supply chains, energy mixes, and market conditions. The results reveal a fundamental asymmetry in the way renewal strategies affect the three main objectives.

Across all configurations, LP emissions are primarily driven by the timing of fleet transitions. Earlier adoption of electric vessels consistently reduces NO_x emissions, while delays increase them. In contrast, LCA outcomes are dominated by upstream factors, material choice, energy source, and manufacturing location, and show much narrower ranges within each pathway. This asymmetry means that operational measures can achieve substantial reductions in local pollutants, but significant reductions in lifecycle CO_2 require the decarbonisation of materials and supply chains upstream. With the exception of Pathway 2, the WTW emission reductions from electrification must be balanced against the CTG and GTC impact of additional battery packs, which vary with fuel type and production/dismantling processes.

The CO_2 depreciation scenario introduced a different dynamic. If future emissions are depreciated, earlier electrification is favoured. Since the environmental impact of the many battery packs that are required is reduced. This benefits both the LP and lifecycle CO_2 emissions. The salvage was also delayed, since future GTC emissions would be lower. This can result in inaction and underscores the importance of scrutinising time-related assumptions in policy and modelling, as they can shift what is deemed optimal.

Across pathways and under equal objective weights, the preferred alternatives consistently show similar components, even when market conditions and upstream parameters vary. Replacement schedules for certain classes, such as the RHIB and SV, remain identical across all scenarios, while variation is mainly concentrated in the BSM classes, where replacement timing shifts in response to material choice, production location, and energy sourcing. This robustness in the overall transition structure contrasts with the variation in performance scores, which remain highly sensitive to upstream configurations. Regardless of the pathway, all strategies require substantial early investment, underscoring the need for financing mechanisms that support rapid capital deployment to achieve long-term sustainability gains.

Although preferred alternatives often share similar transition structures, their TOPSIS scores, and thus their relative desirability, vary considerably between pathways and scenarios. These differences are driven less by operational timing than by upstream factors such as material type, production geography, and energy mix, which influence the LCA and TCO outcomes. This highlights that even when operational strategies converge, upstream configuration choices can substantially alter how desirable a given alternative is in a multi-objective decision context.

The three performance criteria are driven by distinct factors. TCO is primarily determined by acquisition prices, inflation, and vessel maintenance costs. LP results are shaped by vessel-specific emission profiles and fuel type, influencing whether replacements are delayed or prioritised for early electrification. LCA results are governed by both direct and indirect sources. Namely, the number of battery packs, with their carbon-intensive production, and the WTW emissions from fuel use, which depend on the carbon intensity of the electricity mix or marine fuels. These interdependencies make the identification of Pareto optimal strategies inherently complex, highlighting the need to balance costs, air quality, and climate goals.

This modelling framework makes explicit the cost and emission impacts of different technical and economic configurations, providing a transparent decision support tool. It enables stakeholders to explore trade-offs and design resilient fleet renewal strategies that are evidence-based and adaptable to future uncertainties.

9 Discussion

This chapter reflects on the results of the fleet renewal framework developed in this research, placing the findings in the context of the study objectives. It begins by interpreting the main results of the optimisation and decision support process for the PoR case study. The discussion then relates these outcomes to the research sub-questions, as these form the building blocks to answer the main research question and achieve the overall objective of the study. This is followed by a critical review of the limitations and their implications for interpreting the results. Finally, the chapter offers recommendations for methodological refinement, future research, and practical application.

This thesis examined how the PoR could strategically renew its fleet while balancing the conflicting objectives of minimising environmental impact, the TCO, and maintaining operational readiness. Although maritime fleet renewal has been studied extensively, earlier approaches often treat environmental and economic dimensions in isolation and rarely integrate decision-making under uncertainty in a stakeholder-sensitive manner. By developing a two-layer MOO and MCDA framework, combining the ε-constraint method with TOPSIS, this research offers a structured way to evaluate trade-offs and align technical feasibility with stakeholder priorities in complex fleet transitions.

The results reveal a distinct asymmetry in the sensitivities of the three modelled objectives. LP emissions are highly sensitive to transition timing. Earlier replacement of conventional vessels significantly reduces LP. In contrast, LCA emissions are more strongly influenced by upstream and embedded factors, such as the location of production, the hull material, the type of fuel, and the battery manufacturing processes. These emissions respond less to operational schedules and more to technological choices, challenging assumptions that operational electrification alone guarantees sustainability. Meanwhile, TCO outcomes are closely related to early capital investments. The more aggressive transition alternatives show a higher sensitivity to changing economic scenarios but do not significantly impact the relative cost difference between the economic and expensive alternatives.

Some results diverge from initial expectations and merit closer scrutiny. In the CO_2 depreciation scenario, under the assumption that due to technological advancement and electricity grid decarbonisation the CTG and GTC decrease over time, the fleet transition schedules diverged from standard expectations. For some classes, it delayed replacement to avoid present emissions. This raises ethical concerns about potentially rewarding inaction at the expense of long-term climate ambition.

These insights have important implications for fleet management. The framework highlights the value of structuring investments early in the transition to achieve long-term environmental gains, even at the expense of a higher short-term capital outlay. An operationally important finding is the differentiation between vessel classes with robust replacement schedules (RHIB, SV, sPV) and those with high context sensitivity (NM, PV, IRV). The ability to identify low-regret actions versus those that require careful trade-off evaluation is a central contribution of the framework.

Building on these findings, the following sections directly address each research sub-question. This step is essential to connect the detailed results of the scenario to the broader objective of understanding how the PoR can optimise its fleet renewal to minimise emissions, control costs, and maintain operational capability.

What are the key decision factors and fleet operator's interests that influence the timing of vessel replacement? The results of the different configurations show that the timing of replacement is shaped by a combination of operational, economic, and environmental factors. Operationally, the framework results demonstrate that the various classes have different sensitivities. The RHIB, SV, and sPV classes follow near-identical schedules in all scenarios, confirming their robustness as low-regret options. In contrast, the NM, PV and IRV classes display highly variable schedules depending on the upstream configurations and cost assumptions, which requires careful case-by-case evaluation. Economically, the framework results demonstrate that differences in acquisition cost, inflation, and maintenance expenditure directly influenced the timing of replacement. Environmentally, the framework results demonstrate that the different energy carriers, the hull materials, and the vessel-specific emission profile strongly affected the relative benefits of replacement. Together, these results indicate that timing decisions emerge from the interplay of these factors and that the framework can clearly separate strategies that are robust under uncertainty from those that are sensitive to specific assumptions.

What is the total cost of ownership of the vessels in the fleet? Across all economic and strategic scenarios, the modelling results confirm that the largest share of TCO occurs in the early years due to front-loaded CAPEX for vessels, infrastructure, and battery packs. The absolute TCO range shifts heavily between the conservative and optimistic scenarios, with the median TCO for the optimistic scenario being around € 380 million, raising to over € 1060 million for the conservative scenario. For the standard scenario the median value is € 700 million, within the individual scenarios differences around 1-2% are seen between the highest and lowest TCO alternatives. This narrow spread suggests that TCO alone is insufficient to discriminate between competing strategies, especially when environmental and operational outcomes diverge significantly. The TCO component of the framework therefore functions best when used alongside the LCA and LP objectives, enabling a balanced assessment that reflects both cost control and sustainability performance.

What is the life cycle impact of the vessels on the environment? The LCA results show that lifecycle CO_2-equivalent emissions are dominated by the upstream choices, such as the hull material, production location, and the carbon intensity of the manufacturing energy, rather than the operational timing. This explains why the LCA ranges are narrow within each scenario, but differ substantially between the different configurations, ranging between 10,000 - 100,000 t CO_2-eq. The LP emissions, in contrast, are strongly driven by transition timing, with early replacement of conventional high-emission vessels consistently producing large NO_x reductions. For the scenarios where the WTW emissions go down after the transition, it ranges between 14,000-140,000 kg NO_x. If the WTW emissions do increase and the current fleet would remain operational until EOL, the upper bound increases to 310,000 kg NO_x

How can the economic, operational and sustainability factors be combined in a decision support framework? The integrated MOO–MCDA structure proved effective in combining cost, environmental, and operational objectives within a single decision support tool. The Pareto front analysis revealed the trade-off space between the objectives, while TOPSIS translated that space into ranked alternatives based on user-defined weights. This approach allows for the exploration of how changes in upstream configurations or cost conditions shift the set of preferred schedules and to identify replacement actions that are universally robust or scenario-dependent. The results of the case study confirm that the framework supports transparent evaluation under uncertainty, providing decision makers with both flexibility and clarity in prioritising objectives.

At the same time, several limitations should be considered when interpreting these findings. The framework does not cost-differentiate between strategic choices for production location or material, even though these factors influence actual CAPEX. This simplification could lead to an under- or overestimation of the economic performance of certain pathways, particularly when upstream choices differ significantly in cost. Technological improvements in battery production, vessel design, or charging systems, are likely to reduce both costs and emissions in the future. With the exception of the CO_2 depreciation scenario, this was not included. As a result, the framework may underestimate the long-term performance of strategies that delay electrification in anticipation of cleaner, cheaper technology. Similarly, the static assumption for electricity grid composition ignores projected decarbonisation trends; this could bias results against later transitions that would benefit from a lower carbon grid.

For environmental modelling, only the hull was included in CTG and GTC emissions. Excluding other vessel components and maintenance-related emissions likely underestimates the lifecycle footprint, potentially skewing comparisons between early and delayed replacement strategies. The environmental scope was also limited to global warming potential, NO_x and PM emissions, omitting categories such as toxicity, acidification, and noise. Furthermore, only the local impact during the use phase was included. The local impact, such as air pollution and pour labour conditions, during the other stages of the product lifecycle was not included. These exclusions reduce the comprehensiveness of the environmental assessment and may influence the environmental performance of different alternatives.

Finally, while the TCO calculation incorporates the residual value cash flow of the salvaged assets, it does not account for the value of the active company assets at the end of the horizon. This omission may disadvantage late acquisitions in the optimisation, as their remaining value is not reflected in the objective function of the optimisation model. Including the market or book value at the end of the horizon in future studies would offset the disadvantage of late acquisitions and improve the comparability of strategies with different timing profiles.

These limitations point to clear directions for future research. Incorporating a dynamic cost function would improve the realism of the model and enhance the relevance of the decision support it provides. Expanding the range of environmental categories would also increase the depth of the sustainability assessment. Explicit modelling of technological advancements, such as improvements in battery production, technical lifetime, and expected grid decarbonisation trajectories, could substantially alter the trade-offs between LCA, LP, and TCO over time. By capturing how cleaner electricity grids and more efficient manufacturing processes reduce the impacts of CTG and GTC in the future, the framework could provide more forward-looking guidance and help identify transition strategies that remain optimal under evolving conditions of technological and energy systems. The decision support structure developed here, linking MOO with MCDA, also has potential applications beyond maritime contexts, including heavy-duty vehicle transitions and road fleet renewal.

In summary, this thesis provides a methodological and practical framework for planning sustainable fleet renewal, directly addressing the research objective of helping the Port of Rotterdam strategically schedule vessel replacements to reduce greenhouse gas and local pollutant emissions, minimise transition costs, and maintain operational capability. By combining lifecycle emissions analysis, detailed cost modelling, and stakeholder-driven preferences within an integrated optimisation-decision analysis structure, the framework makes the trade-offs that shape renewal decisions explicit. It identifies low-regret actions that are robust across scenarios, as well as context-sensitive strategies whose performance depends on upstream assumptions and economic conditions. This dual capability provides both the strategic clarity and the flexibility needed for decision making under uncertainty. Beyond the case study, the framework offers a transparent, transferable decision support tool that can be adapted to other fleets facing similar sustainability, cost, and operational challenges.

10 Conclusion

This thesis addressed the question: *How can the Port of Rotterdam optimise the fleet renewal process to minimise total polluting emissions and transition costs, while ensuring that operational capacity is maintained?* The results show that this can be achieved through a scenario-based, multi-objective decision framework, combining the ε-constraint method with TOPSIS. With lifecycle CO_2 emissions, local pollution, total cost of ownership, operational constraints, and stakeholder preferences included within the framework.

This approach enables the development and ranking of replacement schedules that minimise both local and global emissions, control the total cost of ownership, and maintain service capacity while remaining adaptable to future conditions.

The results reveal different sensitivities of the three objectives to the fleet transition schedule. Local pollutant emissions are strongly driven by the timing of conventional vessel replacement, which makes scheduling a powerful lever for improving air quality. Lifecycle greenhouse gas emissions are less influenced by operational timing and instead depend heavily on upstream choices such as hull material, production location, and energy source. The total cost of ownership is more affected by the broader economic outlook than by scheduling decisions.

In all scenarios, low-regret actions, such as the early replacement of RHIB, surveyor vessels, and small patrol vessels, remain optimal, while incident response and larger patrol vessels show greater sensitivity to upstream and market conditions. The performance of each strategy is shaped by interlinked drivers: Acquisition prices, inflation, and maintenance costs determine the economic feasibility. The vessel-specific emission profiles and the fuel type dictate the local pollution outcome. The number and production footprint of battery packs, along with the carbon intensity of fuels or electricity, drive life-cycle greenhouse gas emissions.

By combining multi-objective optimisation with multi-criteria decision analysis, the framework moves beyond single-solution approaches. It allows decision-makers to explore how different priorities and external conditions affect optimal schedules and to identify strategies that remain robust under uncertainty.

In conclusion, the Port of Rotterdam can optimise its fleet renewal by applying this flexible, scenario-based framework to evaluate trade-offs between cost, emissions, and operational capacity. Its strategic value lies not in prescribing one universal plan, but in identifying the conditions under which certain actions are consistently preferable. The methodology is transferable to other fleets facing similar sustainability, economic, and operational challenges, offering a structured, evidence-based foundation for the sector's transition.

References

Aardenne, J. v., Colette, A., Degraeuwe, B., Hammingh, P., Viana, M., & Vlieger, I. d. (2013, March). *The impact of international shipping on European air quality and climate forcing* (tech. rep.). European Environment Agency. Publications Office. https://www.eea.europa.eu/en/analysis/publications/the-impact-of-international-shipping

Aiello, G., Quaranta, S., Inguanta, R., Certa, A., & Venticinque, M. (2024). A Multi-Criteria Decision-Making Framework for Zero Emission Vehicle Fleet Renewal Considering Lifecycle and Scenario Uncertainty. *Energies*, *17*(6). https://doi.org/10.3390/en17061371

Ali, I. M., Turan, H. H., & Elsawah, S. (2023). A military fleet mix problem for high-valued defense assets: A simulation-based optimization approach. *Expert Systems with Applications*, *213*. https://doi.org/10.1016/j.eswa.2022.118964

Alp, O., Tan, T., & Udenio, M. (2022). Transitioning to sustainable freight transportation by integrating fleet replacement and charging infrastructure decisions. *Omega (United Kingdom)*, *109*. https://doi.org/10.1016/j.omega.2022.102595

Alvarez, J. F., Tsilingiris, P., Engebrethsen, E. S., & Kakalis, N. M. P. (2011). Robust Fleet Sizing and Deployment for Industrial and Independent Bulk Ocean Shipping Companies. *INFOR: Information Systems and Operational Research*, *49*(2), 93–107. https://doi.org/10.3138/infor.49.2.093

Ampherr. (n.d.). NMC-800-127 | AMPHERR Battery Technologies Inc. https://ampherr.com/en/products/nmc-800-127

Antunes, C. H., Alves, M. J., & Clímaco, J. (2016). *Multiobjective Linear and Integer Programming*. http://www.springer.com/series/13840

Arslan, A. N., & Papageorgiou, D. J. (2017). Bulk ship fleet renewal and deployment under uncertainty: A multi-stage stochastic programming approach. *Transportation Research Part E: Logistics and Transportation Review*, *97*, 69–96. https://doi.org/10.1016/j.tre.2016.10.009

Bakkehaug, R., Eidem, E. S., Fagerholt, K., & Hvattum, L. M. (2014). A stochastic programming formulation for strategic fleet renewal in shipping. *Transportation Research Part E: Logistics and Transportation Review*, *72*, 60–76. https://doi.org/10.1016/J.TRE.2014.09.010

Bellman, R. (1955). Equipment Replacement Policy. *Journal of the Society for Industrial and Applied Mathematics*, *3*(3), 133–136. http://www.jstor.org.tudelft.idm.oclc.org/stable/2098780

Brandstoffen voertuigen (tech. rep.). (2024, September). CO2 emissiefactoren. https://co2emissiefactoren.nl/factoren/2024/9/brandstoffen-voertuigen/

Brans, J. P., Vincke, P., & Mareschal, B. (1986). How to select and how to rank projects: The Promethee method. *European Journal of Operational Research*, *24*(2), 228–238. https://doi.org/10.1016/0377-2217(86)90044-5

Bre, F., & Fachinotti, V. D. (2017). A computational multi-objective optimization method to improve energy efficiency and thermal comfort in dwellings. *Energy and Buildings*, *154*, 283–294. https://doi.org/10.1016/J.ENBUILD.2017.08.002

Carbon intensity of electricity generation, 2000 to 2024 (tech. rep.). (n.d.). Ourworldindata. https://ourworldindata.org/grapher/carbon-intensity-electricity?tab=chart&country=EU-27~EU~NLD~OWID_WRL~OWID_EU27~CHN~VNM~TUR

Castillo, O., & Álvarez, R. (2023). Electrification of Last-Mile Delivery: A Fleet Management Approach with a Sustainability Perspective. *Sustainability (Switzerland)*, *15*(24). https://doi.org/10.3390/su152416909

Castillo Campo, O., & Álvarez Fernández, R. (2023). Economic optimization analysis of different electric powertrain technologies for vans applied to last mile delivery fleets. *Journal of Cleaner Production*, *385*. https://doi.org/10.1016/j.jclepro.2022.135677

CE Delft. (2019). Handbook on the external costs of transport.

Chakraborty, S. (2022). TOPSIS and Modified TOPSIS: A comparative analysis. *Decision Analytics Journal*, *2*, 100021. https://doi.org/10.1016/J.DAJOUR.2021.100021

Charnes, A., Cooper, W. W., & Rhodes, E. (1978). Measuring the efficiency of decision making units. *European Journal of Operational Research*, *2*(6), 429–444. https://doi.org/10.1016/0377-2217(78)90138-8

Cho, S. C., & Perakis, A. N. (1996). Optimal liner fleet routeing strategies. *Maritime Policy and Management*, *23*(3), 249–259. https://doi.org/10.1080/03088839600000087

Clarksons. (2024, January). 2023 Shipping Review, another resilient year | Clarksons | Clarksons. https://www.clarksons.com/home/news-and-insights/2024/2023-shipping-market-review/

Clerc, M. (2006). *Particle Swarm Optimization*.

Coppola, P., Bocciolone, M., Colombo, E., De Fabiis, F., & Sanvito, F. D. (2023). Multi-Criteria Life-Cycle Assessment of bus fleet renewal: A methodology with a case study from Italy. *Case Studies on Transport Policy*, *13*. https://doi.org/10.1016/j.cstp.2023.101044

Deb, K., Agrawal, S., Pratap, A., & Meyarivan, T. (2000). A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II. http://www.iitk.ac.in/kangal

Desantes, J. M., Molina, S., Novella, R., & Lopez-Juarez, M. (2020). Comparative global warming impact and NOX emissions of conventional and hydrogen automotive propulsion systems. *Energy Conversion and Management*, *221*. https://doi.org/10.1016/j.enconman.2020.113137

Dreyfus, S. E. (1960). A Generalized Equipment Replacement Study. *Journal of the Society for Industrial and Applied Mathematics*, *8*(3), 425–435. https://doi.org/10.1137/0108029

Du, H., & Kommalapati, R. R. (2021). Environmental sustainability of public transportation fleet replacement with electric buses in Houston, a megacity in the USA. *International Journal of Sustainable Engineering*, *14*(6), 1858–1870. https://doi.org/10.1080/19397038.2021.1972491

Ecochain. (2025, March). Cradle-to-Grave in LCA – What is it & How does it work? https://ecochain.com/blog/cradle-to-grave-in-lca/

Eirik Ovrum, Tore Longva, Marius Leisner, Eirill Bachmann Mehammer, Ola Gundersen Skåre, Henrik Helgesen, & Øyvind Endresen. (2024). *MARITIME FORECAST TO 2050* (tech. rep.). DNV.

Emissies naar lucht op Nederlands grondgebied (tech. rep.). (2024). Centraal bureau voor statistiek. https://opendata.cbs.nl/statline/#/CBS/nl/dataset/85668NED/table?ts=1747826735955

European Commision. (2016). Well-to-Wheels Analyses. https://joint-research-centre.ec.europa.eu/welcome-jec-website/jec-activities/well-wheels-analyses_en

European Commision. (2019). The European Green Deal. https://commission.europa.eu/strategy-and-policy/priorities-2019-2024/european-green-deal_en

European Commision. (2023, October). Completion of key ‘Fit for 55’ legislation. https://ec.europa.eu/commission/presscorner/detail/en/ip_23_4754

European Commission. (n.d.). Life Cycle Assessment & the EF methods. https://green-business.ec.europa.eu/environmental-footprint-methods/life-cycle-assessment-ef-methods_en

Eurostat. (2019). Greenhouse gas emissions by IPCC source sector, EU-27, 2019.png - Statistics Explained. https://ec.europa.eu/eurostat/statistics-explained/index.php?_EU-27,_2019.png&oldid=539435#filelinks

Eurostat. (2025, April). Electricity prices for non-household consumers. https://ec.europa.eu/eurostat/databrowser/view/nrg_pc_205/default/table?lang=en

Fact Sheet: The facts about steelmaking (tech. rep.). (2022, June). Institute for Energy Economics and Financial Analysis. https://ieefa.org/sites/default/files/2022-06/steel-fact-sheet.pdf

Fagorederbatt. (n.d.). EDERBATT NMC: Compact, High-Performance Battery System| Fagor Ederbatt. https://fagorederbatt.com/product/nmc-series/

Fan, W., Machemehl, R., Gemar, M., & Brown, L. (2014). A stochastic dynamic programming approach for the equipment replacement optimization under uncertainty. *Journal of Transportation Systems Engineering and Information Technology*, *14*(3), 76–84. https://doi.org/10.1016/S1570-6672(13)60137-3

Fee, M., van Oers, B., Logtmeijer, R., Caron, J. D., & Fong, V. (2019). Genetic Algorithm for Optimization of the Replacement Schedules for Major Surface Combatants. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *11934 LNCS*, 161–172. https://doi.org/10.1007/978-3-030-34500-6{_}11

Filák, M., Famfulík, J., & Richtár, M. (2021). Ways of comparing the advantages of operating different types of vehicles in the fleet. *Transportation Research Procedia*, *55*, 133–140. https://doi.org/10.1016/j.trpro.2021.07.089

GHG Protocol. (2013). *Corporate Value Chain (Scope 3) Accounting and Reporting Standard Supplement to the GHG Protocol Corporate Accounting and Reporting Standard GHG Protocol* (tech. rep.).

Giordano, A., Fischbeck, P., & Matthews, H. S. (2018). Environmental and economic comparison of diesel and battery electric delivery vans to inform city logistics fleet replacement strategies. *Transportation Research Part D: Transport and Environment*, *64*, 216–229. https://doi.org/10.1016/j.trd.2017.10.003

GREET. (2024). GREET | Department of Energy. https://www.energy.gov/eere/greet

Hadžić, N., Koričan, M., Ložar, V., & Opetuk, T. (2025). Air emissions from the shipbuilding process. *Marine Pollution Bulletin*, *212*, 117576. https://doi.org/10.1016/J.MARPOLBUL.2025.117576

Hartman, J. C., & Tan, C. H. (2014). Equipment Replacement Analysis: A Literature Review and Directions for Future Research. *The Engineering Economist*, *59*(2), 136–153. https://doi.org/10.1080/0013791X.2013.862891

Hernieuwbare energie in Nederland 2023 (tech. rep.). (2024, September). Centraal bureau voor statistiek. https://www.cbs.nl/nl-nl/longread/rapportages/2024/hernieuwbare-energie-in-nederland-2023?onepage=true

Hillier, F. S., & Lieberman, G. J. (2021). *Introduction to Operations Research* (11th ed.). McGraw-Hill Education.

Hotelling, H. (1925). A General Mathematical Theory of Depreciation. *Journal of the American Statistical Association*, *20*, 340–353. https://doi.org/10.1080/01621459.1925.10503499

Hsu, C. I., Li, H. C., Liu, S. M., & Chao, C. C. (2011). Aircraft replacement scheduling: A dynamic programming approach. *Transportation Research Part E: Logistics and Transportation Review*, *47*(1), 41–60. https://doi.org/10.1016/j.tre.2010.07.006

Hulskotte, J. (2018). *EMS-protocol Emissies door Binnenvaart: Verbrandingsmotoren* (tech. rep.). TNO.

Intergovernmental Panel on Climate Change. (2022). Summary for Policymakers. *Global Warming of 1.5°C*, 1–24. https://doi.org/10.1017/9781009157940.001

International Maritime Organization. (1974, November). International Convention for the Safety of Life at Sea (SOLAS), 1974. https://www.imo.org/en/About/Conventions/Pages/International-Convention-for-the-Safety-of-Life-at-Sea-(SOLAS),-1974.aspx

International Maritime Organization. (2023, July). *2023 IMO STRATEGY ON REDUCTION OF GHG EMISSIONS FROM SHIPS* (tech. rep.). International Maritime Organization. https://wwwcdn.imo.org/localresources/en/OurWork/Environment/Documents/annex/MEPC%2080/Annex%2015.pdf

International Maritime Organization. (2025, April). IMO approves net-zero regulations for global shipping. https://www.imo.org/en/MediaCentre/PressBriefings/pages/IMO-approves-netzero-regulations.aspx

Islam, A., & Lownes, N. (2019). When to go electric? A parallel bus fleet replacement study. *Transportation Research Part D: Transport and Environment*, *72*, 299–311. https://doi.org/10.1016/j.trd.2019.05.007

Kallitsis, E., Lindsay, J. J., Chordia, M., Wu, B., Offer, G. J., & Edge, J. S. (2024). Think global act local: The dependency of global lithium-ion battery emissions on production location and material sources. *Journal of Cleaner Production*, *449*, 141725. https://doi.org/10.1016/J.JCLEPRO.2024.141725

Kana, A. A. (2024). Tools_AHP - MT44035 Design of Complex Specials (2023/24 Q3). https://brightspace.tudelft.nl/d2l/le/content/597030/viewContent/3622352/View

Kvande, H., Saevarsdottir, G., & Welch, B. J. (2022). Direct and Indirect CO2 Equivalent Emissions from Primary Aluminium Production. *Minerals, Metals and Materials Series*, 998–1003. https://doi.org/10.1007/978-3-030-92529-1{_}130/TABLES/2

Li, J., Li, L., Yang, R., & Jiao, J. (2023). Assessment of the lifecycle carbon emission and energy consumption of lithium-ion power batteries recycling: A systematic review and meta-analysis. *Journal of Energy Storage*, *65*, 107306. https://doi.org/10.1016/J.EST.2023.107306

Liu, M., Zhu, G., & Tian, Y. (2024). The historical evolution and research trends of life cycle assessment. *Green Carbon*, *2*(4), 425–437. https://doi.org/10.1016/J.GREENCA.2024.08.003

Loennechen, O., Fagerholt, K., Lagemann, B., & Stålhane, M. (2024). Maritime fleet composition under future greenhouse gas emission restrictions and uncertain fuel prices. *Maritime Transport Research*, *6*. https://doi.org/10.1016/j.martra.2024.100103

Madanchian, M., & Taherdoost, H. (2023). A comprehensive guide to the TOPSIS method for multi-criteria decision making. *Sustainable Social Development*, *1*. https://doi.org/10.54517/ssd.v1i1.2220

Martin, J., Golab, A., Durakovic, G., Zwickl-Bernhard, S., Auer, H., & Neumann, A. (2024). Modeling cost-optimal fuel choices for truck, ship, and airplane fleets: The impact of sustainability commitments. *Energy*, *308*. https://doi.org/10.1016/j.energy.2024.132882

Meng, Q., & Wang, T. (2010). A chance constrained programming model for short-term liner ship fleet planning problems. *Maritime Policy and Management*, *37*(4), 329–346. https://doi.org/10.1080/03088839.2010.486635

Meng, Q., & Wang, T. (2011). A scenario-based dynamic programming model for multi-period liner ship fleet planning. *Transportation Research Part E: Logistics and Transportation Review*, *47*(4), 401–413. https://doi.org/10.1016/j.tre.2010.12.005

Meng, Q., Wang, T., & Wang, S. (2015). Multi-period liner ship fleet planning with dependent uncertain container shipment demand. *Maritime Policy and Management*, *42*(1), 43–67. https://doi.org/10.1080/03088839.2013.865848

Moreno Sader, K., Biswas, S., Jones, R., Mennig, M., Rezaei, R., & Green, W. H. (2025). Battery electric long-haul trucking with overnight charging in the United States: A comprehensive costing and emissions analysis. *Applied Energy*, *384*, 125443. https://doi.org/10.1016/j.apenergy.2025.125443

National Renewable Energy Laboratory. (2025, February). Utility-Scale Battery Storage. https://atb.nrel.gov/electricity/2024/utility-scale_battery_storage

Nature. (2021). Lithium-ion batteries need to be greener and more ethical. *Nature*, *595*(7865), 7. https://doi.org/10.1038/D41586-021-01735-Z

Nicholson, T. A. J., & Pullen, R. D. (1971). Dynamic Programming Applied to Ship Fleet Management. *Journal of the Operational Research Society 1971 22:3*, *22*(3), 211–220. https://doi.org/10.1057/JORS.1971.55

Nordelöf, A., Romare, M., & Tivander, J. (2019). Life cycle assessment of city buses powered by electricity, hydrogenated vegetable oil or diesel. *Transportation Research Part D: Transport and Environment*, *75*, 211–222. https://doi.org/10.1016/J.TRD.2019.08.019

Pantuso, G., Fagerholt, K., & Hvattum, L. M. (2014). A survey on maritime fleet size and mix problems.

Pantuso, G., Fagerholt, K., & Wallace, S. W. (2015). Uncertainty in fleet renewal: a case from maritime transportation. *Transportation Science*.

Parthanadee, P., Buddhakulsomsiri, J., & Charnsethikul, P. (2012). A study of replacement rules for a parallel fleet replacement problem based on user preference utilization pattern and alternative fuel considerations. *Computers and Industrial Engineering*, *63*(1), 46–57. https://doi.org/10.1016/j.cie.2012.01.011

Patricksson, Ø. S., Fagerholt, K., & Rakke, J. G. (2015). The fleet renewal problem with regional emission limitations: Case study from Roll-on/Roll-off shipping. *Transportation Research Part C: Emerging Technologies*, *56*, 346–358. https://doi.org/10.1016/j.trc.2015.04.019

Pelletier, S., Jabali, O., Mendoza, J. E., & Laporte, G. (2019). The electric bus fleet transition problem. *Transportation Research Part C: Emerging Technologies*, *109*, 174–193. https://doi.org/10.1016/j.trc.2019.10.012

Port of Rotterdam. (2023, September). HAVENBEDRIJF ROTTERDAM KLIMAATDOELEN 2030. https://www.portofrotterdam.com/sites/default/files/2023-09/klimaatdoelen-havenbedrijf-rotterdam.pdf

Port of Rotterdam. (2024a). Brandstofverbruik schepen 2024.

Port of Rotterdam. (2024b). Kosten vaartuigen.

Port of Rotterdam. (2024c). Kosteninventarisatie Vloot.

Port of Rotterdam. (2025). Vlootvervangingsplanning 2023-2037 Variant 2C.

Sadeghpour, H., Tavakoli, A., Kazemi, M., & Pooya, A. (2019). A novel approximate dynamic programming approach for constrained equipment replacement problems: A case study. *Advances in Production Engineering And Management*, *14*(3), 355–366. https://doi.org/10.14743/apem2019.3.333

Scholten, P., Otten, M., & van der Veen, R. (2022, September). *Maatregelen verschoning binnenvaart Rotterdam Quickscan voor gemeente Rotterdam* (tech. rep.). CE Delft. Delft. www.ce.nl

Selçuklu, S. B. (2023). Multi-objective Genetic Algorithms. In *Handbook of formal optimization* (pp. 1–37). Springer Nature Singapore. https://doi.org/10.1007/978-981-19-8851-6{_}31-1

Sharma, S., & Kumar, V. (2022). A Comprehensive Review on Multi-objective Optimization Techniques: Past, Present and Future. *Archives of Computational Methods in Engineering*, *29*(7), 5605–5633. https://doi.org/10.1007/S11831-022-09778-9/TABLES/3

Shen, A., & Zhang, J. (2024). Technologies for CO2 emission reduction and low-carbon development in primary aluminum industry in China: A review. *Renewable and Sustainable Energy Reviews*, *189*, 113965. https://doi.org/10.1016/J.RSER.2023.113965

SHIFTR. (n.d.). https://shiftr.no/

Sønnervik, H. H., Msakni, M. K., & Schütz, P. (2024). Decarbonizing the Norwegian fishery fleet – strategic fleet renewal with environmental considerations. *Maritime Transport Research*, *7*. https://doi.org/10.1016/j.martra.2024.100118

Stocker, T. F., Qin, D., Plattner, G. K., Tignor, M. M., Allen, S. K., Boschung, J., Nauels, A., Xia, Y., Bex, V., & Midgley, P. M. (2014). Climate Change 2013 – The Physical Science Basis: Working Group I Contribution to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change. *Climate Change 2013 the Physical Science Basis: Working Group I Contribution to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change*, *9781107057999*, 1–1535. https://doi.org/10.1017/CBO9781107415324

Sustainable Ships. (2025). What is the carbon footprint of steel? https://www.sustainable-ships.org/stories/2022/carbon-footprint-steel

Taherdoost, H., & Madanchian, M. (2023a). Using PROMETHEE Method for Multi-Criteria Decision Making: Applications and Procedures. https://doi.org/10.33552/IJEBM.2023.01.000502

Taherdoost, H., & Madanchian, M. (2023b). Multi-Criteria Decision Making (MCDM) Methods and Concepts. *Encyclopedia*, *3*(1), 77–87. https://doi.org/10.3390/ENCYCLOPEDIA3010006

Taherdoost, H., & Madanchian, M. (2023c). Analytic Network Process (ANP) Method: A Comprehensive Review of Applications, Advantages, and Limitations. *Journal of Data Science and Intelligent Systems*, *1*(1), 12–18. https://doi.org/10.47852/BONVIEWJDSIS3202885

Taherdoost, H., & Madanchian, M. (2023d). A Comprehensive Overview of the ELECTRE Method in Multi Criteria Decision-Making. *Journal of Management Science & Engineering Research*, *6*(2), 5–16. https://doi.org/10.30564/JMSER.V6I2.5637

Taylor, J. S. (1923). A Statistical Theory of Depreciation. *Journal of the American Statistical Association*, *18*, 1010–1023. https://doi.org/10.1080/01621459.1923.10502133

TNO. (2020). Opties voor verduurzaming vloot.

Trading Economics. (2025, April). EU Carbon Permits - Price - Chart - Historical Data - News. https://tradingeconomics.com/commodity/carbon

TU Delft. (n.d.). Resources. https://www.tudelft.nl/tulib/searching-resources/resources#c1036492

Turan, H. H., Elsawah, S., & Ryan, M. J. (2020). A long-term fleet renewal problem under uncertainty: A simulation-based optimization approach. *Expert Systems with Applications*, *145*. https://doi.org/10.1016/j.eswa.2019.113158

Turan, H. H., Jalalvand, F., Elsawah, S., & Ryan, M. J. (2022). A joint problem of strategic workforce planning and fleet renewal: With an application in defense. *European Journal of Operational Research*, *296*(2), 615–634. https://doi.org/10.1016/j.ejor.2021.04.010

Turan, H. H., Jalalvand, F., Kahagalage, S., & El Sawah, S. (2021). Integrating decision maker preferences to a risk-averse multi-objective simulation-based optimization for a military workforce planning, asset management and fleet management problem. *Computers and Industrial Engineering*, *162*. https://doi.org/10.1016/j.cie.2021.107752

United Nations. (2015). THE 17 GOALS | Sustainable Development. https://sdgs.un.org/goals

United Nations. (2024, June). Net Zero Coalition | United Nations. https://www.un.org/en/climatechange/net-zero-coalition

VesselFinder. (2024). Scheepsdatabank - VesselFinder. https://www.vesselfinder.com/nl/vessels?name=RPA&type=0&flag=NL

Watróbski, J., Ziemba, P., Jankowski, J., & Ziolo, M. (2016). Green energy for a green city-A multi-perspective model approach. *Sustainability (Switzerland)*, *8*(8). https://doi.org/10.3390/SU8080702

Wijsmuller, M. A., & Beumee, J. G. (1979). INVESTMENT AND REPLACEMENT ANALYSIS IN SHIPPING. *International Shipbuilding Progress*, *26*(294), 32–43. https://doi.org/10.3233/ISP-1979-2629402

Winkelmann, J., Spinler, S., & Neukirchen, T. (2024). Green transport fleet renewal using approximate dynamic programming: A case study in German heavy-duty road transportation. *Transportation Research Part E: Logistics and Transportation Review*, *186*. https://doi.org/10.1016/j.tre.2024.103547

World Health Organization. (2016). *Ambient air pollution: a global assessment of exposure and burden of disease* (tech. rep.). World Health Organization. https://iris.who.int/handle/10665/250141

WWF. (2017). TRANSPORT SCIENCE-BASED TARGET SETTING GUIDANCE. *WWF*.

Xie, X., Wang, T., & Chen, D. (2000). A dynamic model and algorithm for fleet planning. *Maritime Policy and Management*, *27*(1), 53–63. https://doi.org/10.1080/030888300286680

Young, J., McQueen, N., Charalambous, C., Foteinis, S., Hawrot, O., Ojeda, M., Pilorgé, H., Andresen, J., Psarras, P., Renforth, P., Garcia, S., & van der Spek, M. (2023). The cost of direct air capture and storage can be reduced via strategic deployment but is unlikely to fall below stated cost targets. *One Earth*, *6*(7), 899–917. https://doi.org/10.1016/J.ONEEAR.2023.06.004

Zhao, Y., Ma, Y., Peng, Z., & Zhou, J. (2024). A risk-based decision-making scheme for short-sea liner fleet renewal to achieve carbon reduction targets. *Research in Transportation Business and Management*, *54*. https://doi.org/10.1016/j.rtbm.2024.101112

Zhao, Y., Ye, J., & Zhou, J. (2021). Container fleet renewal considering multiple sulfur reduction technologies and uncertain markets amidst COVID-19. *Journal of Cleaner Production*, *317*. https://doi.org/10.1016/j.jclepro.2021.128361

Zhou, Y., Ong, G. P., & Meng, Q. (2023). The road to electrification: Bus fleet replacement strategies. *Applied Energy*, *337*. https://doi.org/10.1016/j.apenergy.2023.120903

Zitzler, E., Laumanns, M., & Thiele, L. (2001). SPEA2: Improving the Strength Pareto Evolutionary Algorithm.

A

Overview researched literature

Table A.1: Reviewed articles on fleet renewal.

Title	Year	Author	Label	Origin
Green transport fleet renewal using approximate dynamic programming: A case study in German heavy-duty road transportation	2024	Winkelmann	A01	Query A
A risk-based decision-making scheme for short-sea liner fleet renewal to achieve carbon reduction targets	2024	Zhao	A02	Query A
A Multi-Criteria Decision-Making Framework for Zero Emission Vehicle Fleet Renewal Considering Lifecycle and Scenario Uncertainty	2024	Aiello	A04	Query A
Electrification of Last-Mile Delivery: A Fleet Management Approach with a Sustainability Perspective	2023	Castillo	A05	Query A
Multi-Criteria Life-Cycle Assessment of bus fleet renewal: A methodology with a case study from Italy	2023	Copolla	A06	Query A
Integrating decision maker preferences to a risk-averse multi-objective simulation-based optimization for a military workforce planning, asset management and fleet management problem	2021	Turan	A07	Query A
A long-term fleet renewal problem under uncertainty: A simulation-based optimization approach	2020	Turan	A08	Query A
The electric bus fleet transition problem	2019	Pelletier	A09	Query A
A scenario-based dynamic programming model for multi-period liner ship fleet planning	2011	Meng	AA01	AA10
Optimal liner fleet routeing strategies	1996	Cho and Perakis	AA03	AA10
Investment And Replacement Analysis in Shipping	1979	Wijsmuller and Beumee	AA04	AA10
Dynamic Programming Applied to Ship Fleet management	1971	Nicholson and Pullen	AA05	AA10
Robust Fleet Sizing and Deployment for Industrial and Independent Bulk Ocean Shipping Companies	2011	Alvarez	AA06	AA10
A general Mathematical Theory of Depreciation	1925	Hotelling	AA08	AA17
A statistical Theory of Depreciation	1923	Taylor	AA09	AA17
A survey on maritime fleet and size and mix problems	2014	Pantuso	AA10	A11
Uncertainty in fleet renewal: a case from maritime transportation	2015	Pantuso	AA11	A10

Continued on next page

Table A.1: Reviewed Literature on Fleet Renewal and Strategic Replacement (continued)

Title	Year	Author	Label	Origin
Multi-period liner ship fleet planning with dependent uncertain container shipment demand	2015	Meng	AA12	A10
A dynamic model and algorithm for fleet planning	2000	Xie et al.	AA13	AA10
A chance constrained programming model for short-term liner ship fleet planning problems	2010	Meng	AA14	A10
A stochastic programming formulation for strategic fleet renewal in shipping	2014	Bakkehaug	AA16	B21
Equipment Replacement Analysis: A literature Review	2014	Hartman	AA17	Search
Transitioning to sustainable freight transportation by integrating fleet replacement and charging infrastructure decisions	2022	Alp	AA20	A01
A Stochastic Dynamic Programming Approach for the Equipment Replacement Optimization under Uncertainty	2014	Fan	AA23	A01
Aircraft replacement scheduling: A dynamic programming approach	2011	Hsu	AA24	A01
A military fleet mix problem for high-valued defense assets: A simulation-based optimization approach	2023	Ali	AA25	A01
A novel approximate dynamic programming approach for constrained equipment replacement problems: A case study	2019	Sadeghpour	AA26	A01
Battery electric long-haul trucking with overnight charging in the United States: A comprehensive costing and emissions analysis	2025	Sader	B01	Query B
Maritime fleet composition under future greenhouse gas emission restrictions and uncertain fuel prices	2024	Loennechen	B03	Query B
Decarbonizing the Norwegian fishery fleet – strategic fleet renewal with environmental considerations	2024	Sonnervik	B04	Query B
Modeling cost-optimal fuel choices for truck, ship, and airplane fleets: The impact of sustainability commitments	2024	Martin	B05	Query B
Electric mobility toward sustainable cities and road-freight logistics: A systematic review and future research directions	2023	Alarcón	B06	Query B
Economic optimization analysis of different electric powertrain technologies for vans applied to last mile delivery fleets	2023	Castillo Campo	B07	Query B
The road to electrification: Bus fleet replacement strategies	2023	Zhou	B09	Query B
A joint problem of strategic workforce planning and fleet renewal: With an application in defense	2022	Turan	B10	Query B
Environmental sustainability of public transportation fleet replacement with electric buses in Houston, a megacity in the USA	2021	DU	B11	Query B
Container fleet renewal considering multiple sulfur reduction technologies and uncertain markets amidst COVID-19	2021	Zhao	B12	Query B
The short-term cost of greening the global fleet	2021	Schinas	B13	Query B
When to go electric? A parallel bus fleet replacement study	2019	Islam	B17	Query B
Environmental and economic comparison of diesel and battery electric delivery vans to inform city logistics fleet replacement strategies	2018	Giordano	B18	Query B

Continued on next page

Table A.1: Reviewed Literature on Fleet Renewal and Strategic Replacement (continued)

Title	**Year**	**Author**	**Label**	**Origin**
Bulk ship fleet renewal and deployment under uncertainty: A multi-stage stochastic programming approach	2017	Arslan	B21	Query B
Simulation of the impacts on carbon dioxide emissions from replacement of a conventional Brazilian taxi fleet by electric vehicles	2016	Teixeira	B24	Query B
A study of replacement rules for a parallel fleet replacement problem based on user preference utilization pattern and alternative fuel considerations	2012	Parthanadee	B26	Query B
Fleet replacement under technological shocks	2012	Res	B27	Query B
Environmental sustainability of the vehicle fleet change in public city transport of selected city in central Europe	2020	Konecny	C03	Query C
Genetic Algorithm for Optimization of the Replacement Schedules for Major Surface Combatants	2019	Fee	C04	Query C
Electric and plug-in hybrid vehicles influence on CO2 and water vapour emissions	2011	Silva	C05	Query C
Comparative global warming impact and NOX emissions of conventional and hydrogen automotive propulsion systems	2020	Desantes	G02	Query G
The fleet renewal problem with regional emission limitations: Case study from Roll-on/Roll-off shipping	2015	Patricksson	G03	Query G

B

Model Architecture

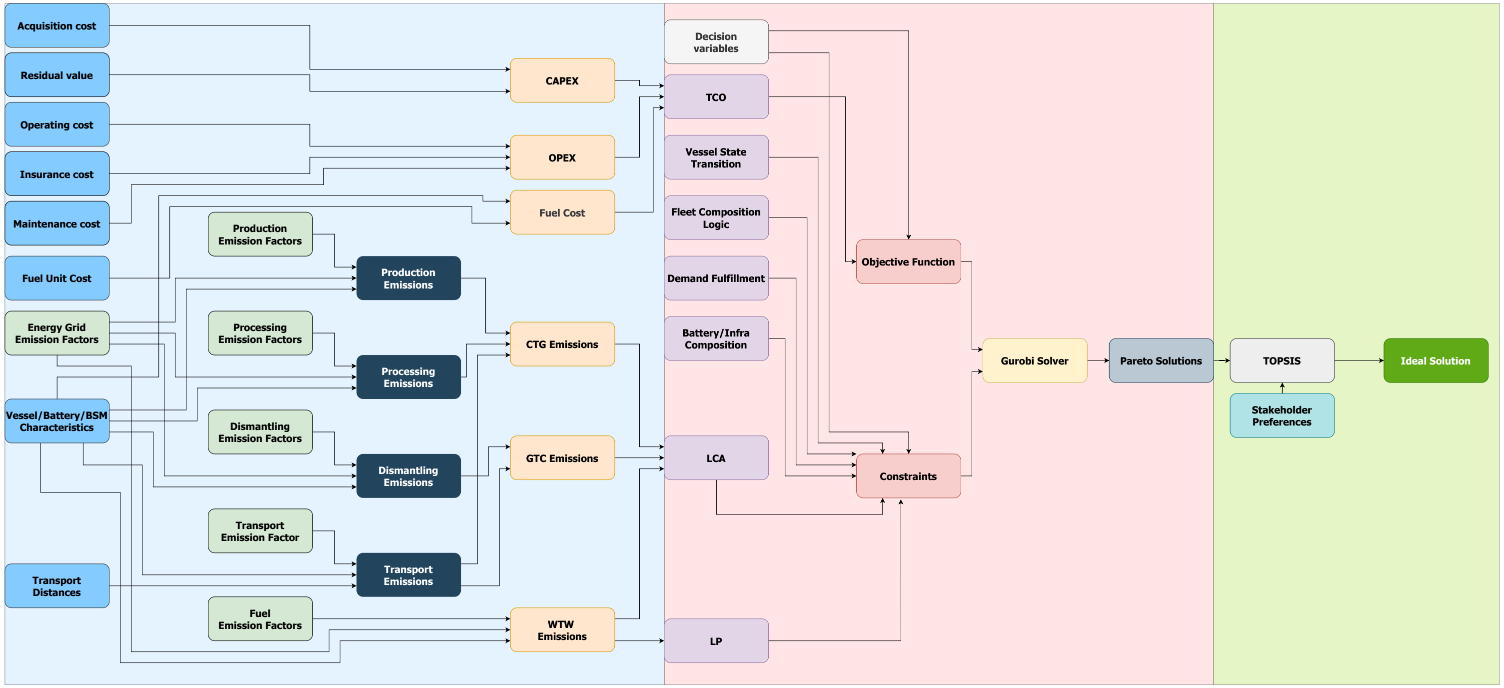


Figure B.1: Overview of the decision support model architecture.

C
Preprocessing layer code

Preprocessing_input.py

```
"""
Input parameters for fleet renewal model - Port of Rotterdam
Structured for the preprocessing layer of the decision support tool.
Author: Jelmer Pentinga
"""
# === Vessel Characteristics ===
def vessel_characteristics():
    """
  Return vessel-specific characteristics including age, weight, emissions, and material type.

  Returns:
      dict: Vessel parameters by ID.
  """

    return {
        # IRV vessels
        0: {"name": "RPA␣10",  "class": "IRV", "initial_age": 91,   "eol": 132, "materials":
            {"steel_tons": 104, "aluminium_tons": 0, "steel_type": "EAF", "al_type": "
            secondary"},    "specific_emissions": {"fuel_consumption": 28.6, "NOX": 4.57, "PM
            ": 0.186, "fuel_type": "HVO"}},
        1: {"name": "RPA␣11",  "class": "IRV", "initial_age": 91,   "eol": 136, "materials":
            {"steel_tons": 104, "aluminium_tons": 0, "steel_type": "EAF", "al_type": "
            secondary"},    "specific_emissions": {"fuel_consumption": 28.6, "NOX": 4.57, "PM
            ": 0.186, "fuel_type": "HVO"}},
        2: {"name": "RPA␣12",  "class": "IRV", "initial_age": 95,   "eol": 148, "materials":
            {"steel_tons": 104, "aluminium_tons": 0, "steel_type": "EAF", "al_type": "
            secondary"},    "specific_emissions": {"fuel_consumption": 28.6, "NOX": 12.6, "PM
            ": 0.6, "fuel_type": "HVO"}},
        3: {"name": "RPA␣13",  "class": "IRV", "initial_age": 95,   "eol": 128, "materials":
            {"steel_tons": 104, "aluminium_tons": 0, "steel_type": "EAF", "al_type": "
            secondary"},    "specific_emissions": {"fuel_consumption": 28.6, "NOX": 12.6, "PM
            ": 0.6, "fuel_type": "HVO"}},
        4: {"name": "RPA␣14",  "class": "IRV", "initial_age": 159,  "eol": 188, "materials":
            {"steel_tons": 170, "aluminium_tons": 0, "steel_type": "EAF", "al_type": "
            secondary"},    "specific_emissions": {"fuel_consumption": 44.4, "NOX": 12.6, "PM
            ": 0.6, "fuel_type": "HVO"}},
        5: {"name": "RPA␣15",  "class": "IRV", "initial_age": 159,  "eol": 192, "materials":
            {"steel_tons": 170, "aluminium_tons": 0, "steel_type": "EAF", "al_type": "
            secondary"},    "specific_emissions": {"fuel_consumption": 44.4, "NOX": 12.6, "PM
            ": 0.6, "fuel_type": "HVO"}},
        6: {"name": "RPA␣16",  "class": "IRV", "initial_age": 91,   "eol": 140, "materials":
            {"steel_tons": 110, "aluminium_tons": 0, "steel_type": "EAF", "al_type": "
            secondary"},    "specific_emissions": {"fuel_consumption": 28.6, "NOX": 12.6, "PM
            ": 0.6, "fuel_type": "HVO"}},
        7: {"name": "RPA␣30",  "class": "IRV", "initial_age": None, "eol": 100, "materials":
            {"steel_tons": 0,   "aluminium_tons": 105, "steel_type": "EAF", "al_type": "
            secondary"},  "specific_emissions": {"fuel_consumption": 152, "NOX": 0.0, "PM":
```

```
    0.0, "fuel_type": "Electricity"}},
    8: {"name": "RPA␣31",  "class": "IRV", "initial_age": None, "eol": 100, "materials": {"steel_tons": 0,   "aluminium_tons": 105, "steel_type": "EAF", "al_type": "secondary"},  "specific_emissions": {"fuel_consumption": 152, "NOX": 0.0, "PM": 0.0, "fuel_type": "Electricity"}},
    9: {"name": "RPA␣32",  "class": "IRV", "initial_age": None, "eol": 100, "materials": {"steel_tons": 0,   "aluminium_tons": 105, "steel_type": "EAF", "al_type": "secondary"},  "specific_emissions": {"fuel_consumption": 152, "NOX": 0.0, "PM": 0.0, "fuel_type": "Electricity"}},
    10: {"name": "RPA␣33", "class": "IRV", "initial_age": None, "eol": 100, "materials": {"steel_tons": 0,   "aluminium_tons": 105, "steel_type": "EAF", "al_type": "secondary"},  "specific_emissions": {"fuel_consumption": 152, "NOX": 0.0, "PM": 0.0, "fuel_type": "Electricity"}},
    11: {"name": "RPA␣34", "class": "IRV", "initial_age": None, "eol": 100, "materials": {"steel_tons": 0,   "aluminium_tons": 105, "steel_type": "EAF", "al_type": "secondary"},  "specific_emissions": {"fuel_consumption": 152, "NOX": 0.0, "PM": 0.0, "fuel_type": "Electricity"}},
    12: {"name": "RPA␣35", "class": "IRV", "initial_age": None, "eol": 100, "materials": {"steel_tons": 0,   "aluminium_tons": 105, "steel_type": "EAF", "al_type": "secondary"},  "specific_emissions": {"fuel_consumption": 152, "NOX": 0.0, "PM": 0.0, "fuel_type": "Electricity"}},
    # PV vessels
    13: {"name": "RPA␣6",  "class": "PV", "initial_age": 75,   "eol": 108, "materials": {"steel_tons": 34,   "aluminium_tons": 4, "steel_type": "EAF", "al_type": "secondary"},    "specific_emissions": {"fuel_consumption": 22.2, "NOX": 10.05, "PM": 0.54, "fuel_type": "HVO"}},
    14: {"name": "RPA␣7",  "class": "PV", "initial_age": 79,   "eol": 120, "materials": {"steel_tons": 34,   "aluminium_tons": 4, "steel_type": "EAF", "al_type": "secondary"},    "specific_emissions": {"fuel_consumption": 22.2, "NOX": 10.05, "PM": 0.54, "fuel_type": "HVO"}},
    15: {"name": "RPA␣8",  "class": "PV", "initial_age": 31,   "eol": 84,  "materials": {"steel_tons": 0,    "aluminium_tons": 20, "steel_type": "EAF", "al_type": "secondary"},   "specific_emissions": {"fuel_consumption": 14, "NOX": 4.57, "PM": 0.186, "fuel_type": "HVO"}},
    16: {"name": "RPA␣22", "class": "PV", "initial_age": None, "eol": 100, "materials": {"steel_tons": 0,    "aluminium_tons": 95, "steel_type": "EAF", "al_type": "secondary"},   "specific_emissions": {"fuel_consumption": 143, "NOX": 0, "PM": 0, "fuel_type": "Electricity"}},
    17: {"name": "RPA␣23", "class": "PV", "initial_age": None, "eol": 100, "materials": {"steel_tons": 0,    "aluminium_tons": 95, "steel_type": "EAF", "al_type": "secondary"},   "specific_emissions": {"fuel_consumption": 143, "NOX": 0, "PM": 0, "fuel_type": "Electricity"}},
    18: {"name": "RPA␣24", "class": "PV", "initial_age": None, "eol": 100, "materials": {"steel_tons": 0,    "aluminium_tons": 95, "steel_type": "EAF", "al_type": "secondary"},   "specific_emissions": {"fuel_consumption": 143, "NOX": 0, "PM": 0, "fuel_type": "Electricity"}},
    # sPV vessels
    19: {"name": "RPA␣1",  "class": "sPV", "initial_age": 91,  "eol": 120, "materials": {"steel_tons": 20,   "aluminium_tons": 2.5, "steel_type": "EAF", "al_type": "secondary"},  "specific_emissions": {"fuel_consumption": 17.5, "NOX": 12.6, "PM": 0.6, "fuel_type": "HVO"}},
    20: {"name": "RPA␣2",  "class": "sPV", "initial_age": 91,  "eol": 136, "materials": {"steel_tons": 20,   "aluminium_tons": 2.5, "steel_type": "EAF", "al_type": "secondary"},  "specific_emissions": {"fuel_consumption": 17.5, "NOX": 12.6, "PM": 0.6, "fuel_type": "HVO"}},
    21: {"name": "RPA␣21", "class": "sPV", "initial_age": None,"eol": 100, "materials": {"steel_tons": 0,    "aluminium_tons": 35, "steel_type": "EAF", "al_type": "secondary"},   "specific_emissions": {"fuel_consumption": 110, "NOX": 0, "PM": 0, "fuel_type": "Electricity"}},

    # SV vessels
    22: {"name": "SV␣1",   "class": "SV", "initial_age": 75,    "eol": 100, "materials": {"steel_tons": 30,  "aluminium_tons": 3.8, "steel_type": "EAF", "al_type": "secondary"}, "specific_emissions": {"fuel_consumption": 9.4, "NOX": 7.26, "PM": 0.23, "fuel_type": "HVO"}},
    23: {"name": "SV␣2",   "class": "SV", "initial_age": 75,    "eol": 104, "materials": {"steel_tons": 30,  "aluminium_tons": 3.8, "steel_type": "EAF", "al_type": "secondary"}, "specific_emissions": {"fuel_consumption": 9.4, "NOX": 7.26, "PM": 0.23, "fuel_type": "HVO"}},
    24: {"name": "SV␣41",  "class": "SV", "initial_age": None,  "eol": 100, "materials": {"steel_tons": 0,   "aluminium_tons": 35, "steel_type": "EAF", "al_type": "
```

```
            secondary"},  "specific_emissions": {"fuel_consumption": 51, "NOX": 0, "PM": 0, "fuel_type": "Electricity"}},
        25: {"name": "SV␣42",  "class": "SV", "initial_age": None,  "eol": 100, "materials": {"steel_tons": 0,  "aluminium_tons": 35, "steel_type": "EAF", "al_type": "secondary"},  "specific_emissions": {"fuel_consumption": 51, "NOX": 0, "PM": 0, "fuel_type": "Electricity"}},
        # NM vessels
        26: {"name": "NM",    "class": "NM", "initial_age": 123,   "eol": 148, "materials": {"steel_tons": 0,  "aluminium_tons": 60, "steel_type": "EAF", "al_type": "secondary"}, "specific_emissions": {"fuel_consumption": 25, "NOX": 7.26, "PM": 0.23, "fuel_type": "HVO"}},
        27: {"name": "GM",    "class": "NM", "initial_age": None,  "eol": 100, "materials": {"steel_tons": 0,  "aluminium_tons": 60, "steel_type": "EAF", "al_type": "secondary"}, "specific_emissions": {"fuel_consumption": 135, "NOX": 0, "PM": 0, "fuel_type": "Electricity"}},
        # RHIB vessels
        28: {"name": "RPA␣5",  "class": "RHIB", "initial_age": 23,   "eol": 44, "materials": {"steel_tons": 0,  "aluminium_tons": 1.6, "steel_type": "EAF", "al_type": "secondary"}, "specific_emissions": {"fuel_consumption": 1.6, "NOX": 7.26, "PM": 0.23, "fuel_type": "HVO"}},
        29: {"name": "RPA␣25", "class": "RHIB", "initial_age": None, "eol": 44, "materials": {"steel_tons": 0,  "aluminium_tons": 1.6, "steel_type": "EAF", "al_type": "secondary"}, "specific_emissions": {"fuel_consumption": 1.6, "NOX": 4.57, "PM": 0.186, "fuel_type": "HVO"}},
        30: {"name": "RPA␣26", "class": "RHIB", "initial_age": None, "eol": 44, "materials": {"steel_tons": 0,  "aluminium_tons": 1.6, "steel_type": "EAF", "al_type": "secondary"}, "specific_emissions": {"fuel_consumption": 1.6, "NOX": 4.57, "PM": 0.186, "fuel_type": "HVO"}}
    }
def vessel_costs():
    """
    Return vessel cost estimates (CAPEX/OPEX/residual) in €1000.

    Returns:
        dict: Cost components per vessel ID.
    """
    return {
        0: {"acquisition": 0,      "residual": 1500,  "operating": 20, "insurance": 11}, # RPA 10
        1: {"acquisition": 0,      "residual": 1500,  "operating": 18, "insurance": 11}, # RPA 11
        2: {"acquisition": 0,      "residual": 850,   "operating": 18, "insurance": 8},  # RPA 12
        3: {"acquisition": 0,      "residual": 950,   "operating": 15, "insurance": 8},  # RPA 13
        4: {"acquisition": 0,      "residual": 650,   "operating": 15, "insurance": 7},  # RPA 14
        5: {"acquisition": 0,      "residual": 850,   "operating": 20, "insurance": 8},  # RPA 15
        6: {"acquisition": 0,      "residual": 1250,  "operating": 17, "insurance": 10}, # RPA 16
        7: {"acquisition": 24600,  "residual": 18400, "operating": 15, "insurance": 61}, # RPA 30
        8: {"acquisition": 24600,  "residual": 18400, "operating": 15, "insurance": 61}, # RPA 31
        9: {"acquisition": 24600,  "residual": 18400, "operating": 15, "insurance": 61}, # RPA 32
        10: {"acquisition": 24600, "residual": 18400, "operating": 15, "insurance": 61}, # RPA 33
        11: {"acquisition": 24600, "residual": 18400, "operating": 15, "insurance": 61}, # RPA 34
        12: {"acquisition": 24600, "residual": 18400, "operating": 15, "insurance": 61}, # RPA 35
        13: {"acquisition": 0,     "residual": 900,   "operating": 14, "insurance": 3},  # RPA 6
        14: {"acquisition": 0,     "residual": 875,   "operating": 16, "insurance": 3},  # RPA 7
        15: {"acquisition": 0,     "residual": 3500,  "operating": 16, "insurance": 16}, # RPA 8
        16: {"acquisition": 21200, "residual": 15900, "operating": 14, "insurance": 53}, # RPA 22
```

```python
        17: {"acquisition": 21200, "residual": 15900, "operating": 14, "insurance": 53}, # RPA 23
        18: {"acquisition": 21200, "residual": 15900, "operating": 14, "insurance": 53}, # RPA 24
        19: {"acquisition": 0,     "residual": 450,   "operating": 12, "insurance": 4},  # RPA 1
        20: {"acquisition": 0,     "residual": 500,   "operating": 13, "insurance": 4},  # RPA 2
        21: {"acquisition": 11200, "residual": 8430,  "operating": 10, "insurance": 28}, # RPA 21
        22: {"acquisition": 0,     "residual": 350,   "operating": 11, "insurance": 3},  # SV 1
        23: {"acquisition": 0,     "residual": 350,   "operating": 10, "insurance": 3},  # SV 2
        24: {"acquisition": 10700, "residual": 8050,  "operating": 10, "insurance": 27}, # SV 41
        25: {"acquisition": 10700, "residual": 8050,  "operating": 10, "insurance": 27}, # SV 42
        26: {"acquisition": 0,     "residual": 2500,  "operating": 33, "insurance": 16}, # NM
        27: {"acquisition": 23400, "residual": 17500, "operating": 25, "insurance": 59}, # GM
        28: {"acquisition": 0,     "residual": 250,   "operating": 6,  "insurance": 1.5},  # RPA 5
        29: {"acquisition": 600,   "residual": 450.0, "operating": 4,  "insurance": 1.5}, # RPA 25
        30: {"acquisition": 600,   "residual": 450.0, "operating": 4,  "insurance": 1.5} # RPA 26
    }

def maintenance(): #(age,cost)
    """
    Return quarterly maintenance schedule and associated cost per vessel.

    Returns:
        dict: Keys are vessel IDs, values are lists of (quarter, €1000) tuples.
    """
    maintenance = {
        0: [(99, 865), (109, 700), (120, 740)],  # IRV 10
        1: [(101, 865), (111, 700), (121, 740), (131, 525)],  # IRV 11
        2: [(98, 465), (108, 825), (118, 540), (128, 650), (138, 615)],  # IRV 12
        3: [(111, 675), (121, 590)],  # IRV 13
        4: [(168, 1000), (178, 650)],  # IRV 14
        5: [(172, 525), (182, 1000)],  # IRV 15
        6: [(102, 765), (112, 900), (122, 615), (132, 650)],  # IRV 16
        7: [(10, 500), (20, 500), (30, 500), (40, 500), (50, 500), (60, 500), (70, 500), (80, 500), (90, 500), (100, 500), (110, 500), (120, 250)],  # IRV 30
        8: [(10, 500), (20, 500), (30, 500), (40, 500), (50, 500), (60, 500), (70, 500), (80, 500), (90, 500), (100, 500), (110, 500), (120, 500)],  # IRV 31
        9: [(10, 500), (20, 500), (30, 500), (40, 500), (50, 500), (60, 500), (70, 500), (80, 500), (90, 500), (100, 500), (110, 500), (120, 500)],  # IRV 32
        10: [(10, 500), (20, 500), (30, 500), (40, 500), (50, 500), (60, 500), (70, 500), (80, 500), (90, 500), (100, 500), (110, 500), (120, 500)],  # IRV 33
        11: [(10, 500), (20, 500), (30, 500), (40, 500), (50, 500), (60, 500), (70, 500), (80, 500), (90, 500), (100, 500), (110, 500), (120, 500)],  # IRV 34
        12: [(10, 500), (20, 500), (30, 500), (40, 500), (50, 500), (60, 500), (70, 500), (80, 500), (90, 500), (100, 500), (110, 500), (120, 500)],  # IRV 35
        13: [(81, 465), (90, 650), (100, 465)],  # PV 6
        14: [(80, 465), (90, 650), (99, 465), (108, 650)],  # PV 7
        15: [(44, 75), (56, 200), (64, 75), (72, 200)],  # PV 8
        16: [(10, 400), (20, 400), (30, 400), (40, 400), (50, 400), (60, 400), (70, 400), (80, 400), (90, 400), (100, 400), (110, 400), (120, 400)],  # PV 22
        17: [(10, 400), (20, 400), (30, 400), (40, 400), (50, 400), (60, 400), (70, 400), (80, 400), (90, 400), (100, 400), (110, 400), (120, 400)],  # PV 23
        18: [(10, 400), (20, 400), (30, 400), (40, 400), (50, 400), (60, 400), (70, 400), (80, 400), (90, 400), (100, 400), (110, 400), (120, 400)],  # PV 24
        19: [(100, 750), (110, 465)],  # PV 1
        20: [(99, 750), (111, 350), (124, 350)],  # PV 2
        21: [(10, 275), (20, 275), (30, 275), (40, 275), (50, 275), (60, 275), (70, 275), (80, 275), (90, 275), (100, 275), (110, 275), (120, 275)],  # sPV 21
        22: [(75, 275), (85, 450)],  # SV 1
        23: [(84, 525), (94, 275)],  # SV 2
        24: [(10, 275), (20, 275), (30, 275), (40, 275), (50, 275), (60, 275), (70, 275),
```

```
            (80, 275), (90, 275), (100, 275), (110, 275), (120, 275)],  # SV 41
        25: [(10, 275), (20, 275), (30, 275), (40, 275), (50, 275), (60, 275), (70, 275),
            (80, 275), (90, 275), (100, 275), (110, 275), (120, 275)],  # SV 42
        26: [(123, 450), (127, 450), (131, 450), (135, 450), (139, 450), (143, 450), (147,
            450)],  # NM
        27: [(10, 400), (20, 400), (30, 400), (40, 400), (50, 400), (60, 400), (70, 400),
            (80, 400), (90, 400), (100, 400), (110, 400), (120, 400)],  # GM
        28: [(23, 50), (27, 50), (31, 50), (35, 50), (39, 50), (43, 50), (47, 50)],  # RHIB 5
        29: [(3, 50), (7, 50), (11, 50), (15, 50), (19, 50), (23, 50), (27, 50), (31, 50),
            (35, 50), (39, 50), (43, 50), (47, 50), (51, 50), (55, 50), (59, 50)],  # RHIB 25
        30: [(3, 50), (7, 50), (11, 50), (15, 50), (19, 50), (23, 50), (27, 50), (31, 50),
            (35, 50), (39, 50), (43, 50), (47, 50), (51, 50), (55, 50), (59, 50)]  # RHIB 55
    }
    return maintenance

# === Battery Specifications ===
def battery_characteristics():
    """
    Return capacity and weight of batteries used by vessel class.

    Returns:
        dict: Battery capacity (MWh) and weight (tons/MWh).
    """
    capacity = {
             "IRV": 1,
             "PV": 1,
             "sPV": 2,
             "SV": 2,
             "RHIB": 0,
             "NM": 1}
    weight = 5.9                  # in ton per MWh
    return{
        "capacity": capacity,
        "weight": weight
        }

def battery_costs():
    """
    Return economic parameters for battery acquisition, residual value, and OPEX in €1000.

    Returns:
        dict
    """
    return {
        "acquisition": 670,
        "residual": 570,
        "operating": 4
    }

# === Infrastructure Parameters ===
def infrastructure_characteristics():
    """
    Return infrastructure asset material compositions and types.

    Returns:
        dict
    """
    return {"BSM":
            {"steel_tons": 4, "steel_type": "SCRAP", "aluminium_tons": 0,"al_type": "
                secondary"}
            }

def infrastructure_costs():
    """
    Return CAPEX, OPEX, and residual value of infrastructure assets in €1000.

    Returns:
        dict
    """
    acquisition = {
        "BSM": 8110,
```

```
        "Shorepower": 3800
    }
    residual = {
        "BSM": 6082,
        "Shorepower": 2850
    }
    operating = {
        "BSM": 51,
        "Shorepower": 24
    }
    return {
        "acquisition": acquisition,
        "residual": residual,
        "operating": operating
    }

# === Fuel and Transport ===
def fuel_type_cost():
    """
    Return fuel prices per energy unit €(1000 per ton or MWh).

    Returns:
        dict
    """
    return {
        "MDO": 0.92,
        "HVO": 1.27,
        "Electricity": 0.24
    }

def transport_distances():
    """
    Return relevant transport distances in km used in emission calculation.

    Returns:
        dict:
    """
    return {
        "romania_vietnam_rotterdam": 31800,
        "china_rotterdam": 19500,
        "romania_rotterdam": 6300,
        "norway_rotterdam": 1000,
        "rotterdam_antalya": 6000,
        "local": 0
    }
# === Economic Adjustment Factors ===
def depreciation_factors():
    """
    Return quarterly depreciation factors for economic components.

    Returns:
        dict
    """
    return {
        "vessels": 0.0075,
        "batteries": 0.02,
        "infrastructure": 0.00625,
        "CO2": 0.00
    }

def inflation_factor():
    """
    Return quarterly inflation factor for price adjustment.

    Returns:
        float
    """
    return 0.005
```

Emission_factors.py Code

```python
"""
Emission factors for fleet renewal model - Port of Rotterdam
Structured for the preprocessing layer of the decision support tool.
Author: Jelmer Pentinga
"""

# === Material Processing Emissions (CTG: Cradle-to-Gate) ===

def processing_emission_factors():
    """
    Emissions from raw material processing for steel and aluminium.

    Returns:
        dict: Emission factors per material type (in tCO2-eq or MWh/ton).
    """
    return {
        "steel": {
            "BOF": {
                "direct": {"CO2": 1.2},
                "indirect": {"MWh": 2.22}
            },
            "EAF": {
                "direct": {"CO2": 1.0},
                "indirect": {"MWh": 0.89}
            },
            "SCRAP": {
                "direct": {"CO2": 0.04},
                "indirect": {"MWh": 0.58}
            }
        },
        "aluminium": {
            "primary": {
                "direct": {"CO2": 6.0},
                "indirect": {"MWh": 14.27}
            },
            "secondary": {
                "direct": {"CO2": 0.23},
                "indirect": {"MWh": 0.03}
            }
        }
    }

# === Manufacturing Emissions ===

def production_emission_factors():
    """
    Emissions from production of components (batteries and metals).

    Returns:
        dict: Direct and indirect emissions per ton produced.
    """
    return {
        "metals": {
            "direct_emissions": {"CO2": 0.000771},
            "indirect_emissions": {"MWh": 0.566}
        },
        "battery": {
            "direct_emissions": {"CO2": 56}, #per MWH battery
            "indirect_emissions": {"MWh": 60.0} # per MWH battery
        }
    }

# === Dismantling Emissions (GTC: Grave-to-Cradle) ===

def dismantling_emission_factors():
    """
    Emissions from dismantling and recycling of vessel and battery materials.

    Returns:
```

```
        dict: Indirect emissions per ton.
    """
    return {
        "metals": {
            "indirect_emissions": {"MWh": 0.518}
        },
        "battery": {
            "indirect_emissions": {"MWh": 22.98}
        }
    }

# === Transport Emissions (in gCO2-eq / ton·km) ===

def transport_emission_factors():
    """
    Emission factor for maritime transport of materials.

    Returns:
        float: gCO2-eq per ton·km (converted in model to tCO2).
    """
    return 7.9


# === Fuel Emissions (WTW: Well-to-Wheel) ===

def fuel_emission_factors():
    """
    Emissions per ton of fuel combusted.

    Returns:
        dict: Fuel type mapped to tCO2-eq/ton.
    """
    return {
        "MDO": {"CO2": 4.04},
        "HVO": {"CO2": 0.41},
        "Electricity": None  # Handled separately via grid emissions
    }

# === Electricity Grid Emissions ===

def energy_grid_emission_factors():
    """
    Emission factors per MWh based on source of electricity.

    Returns:
        dict: Grid composition mapped to tCO2-eq/MWh and local pollutant emissions.
    """
    return {
        "grid_composition": {
            "clean_energy": {"CO2": 0.0, "NOX": 0.0, "PM": 0.0},
            "netherlands_grey": {"CO2": 0.536, "NOX": 0.00025, "PM": 0.0000056},
            "netherlands_mix": {"CO2": 0.328, "NOX": 0.00015, "PM": 0.0000034},
            "worldwide": {"CO2": 0.473, "NOX": 0.0, "PM": 0.0},
            "EU": {"CO2": 0.237, "NOX": 0.0, "PM": 0.0},
            "china": {"CO2": 0.56, "NOX": 0.0, "PM": 0.0},
            "turkey": {"CO2": 0.47, "NOX": 0.0, "PM": 0.0},
            "vietnam": {"CO2": 0.472, "NOX": 0.0, "PM": 0.0}
        }
    }
```

Intermediate_calculations.py Code

```
"""
Intermediate calculations for fleet renewal model - Port of Rotterdam
Structured for the preprocessing layer of the decision support tool.
Author: Jelmer Pentinga
"""

# === MATERIAL STAGE EMISSIONS ===

```

```python
def processing_emissions(materials, processing_factors, grid_emission):
    """
    Computes cradle-to-gate (CTG) emissions from steel and aluminium processing.

    Parameters:
        materials (dict): {steel_tons, aluminium_tons, steel_type, al_type}
        processing_factors (dict): Output of processing_emission_factors()
        grid_emission (dict): CO2 per MWh from selected electricity mix

    Returns:
        float: Total processing emissions in tons  CO-equivalent
    """
    steel = materials["steel_tons"]
    aluminium = materials["aluminium_tons"]
    steel_type = materials["steel_type"]
    al_type = materials["al_type"]

    steel_direct = steel * processing_factors["steel"][steel_type]["direct"]["CO2"]
    steel_indirect = steel * processing_factors["steel"][steel_type]["indirect"]["MWh"] * grid_emission["CO2"]
    al_direct = aluminium * processing_factors["aluminium"][al_type]["direct"]["CO2"]
    al_indirect = aluminium * processing_factors["aluminium"][al_type]["indirect"]["MWh"] * grid_emission["CO2"]

    return steel_direct + steel_indirect + al_direct + al_indirect


def metal_production_emissions(materials, production_factors, grid_emission):
    """
    Computes emissions from vessel metal part production.

    Parameters:
        materials (dict)
        production_factors (dict): Output of production_emission_factors()
        grid_emission (dict)

    Returns:
        float: Total emissions in tons  CO-equivalent
    """
    indirect_energy = production_factors["metals"]["indirect_emissions"]["MWh"]
    direct_emissions = production_factors["metals"]["direct_emissions"]["CO2"]
    total_tons = materials["steel_tons"] + materials["aluminium_tons"]

    indirect = total_tons * indirect_energy * grid_emission["CO2"]
    direct = total_tons * direct_emissions
    return indirect + direct


def metal_dismantling_emissions(materials, dismantling_factors, grid_emission):
    """
    Computes dismantling emissions for vessel metals.

    Returns:
        float:  CO-equivalent emissions from dismantling (tons)
    """
    steel = materials["steel_tons"]
    aluminium = materials["aluminium_tons"]
    total_energy = (steel + aluminium) * dismantling_factors["metals"]["indirect_emissions"]["MWh"]
    return total_energy * grid_emission["CO2"]



# === BATTERY EMISSIONS ===

def battery_production_emissions(vessel_class, production_factors, grid_emission, battery_characteristics):
    """
    Computes  CO emissions from battery production by vessel class.

    Returns:
```

```
        float: Total CO-equivalent emissions (tons)
    """
    capacity_dict = battery_characteristics["capacity"]
    capacity_mwh = capacity_dict.get(vessel_class, 0)

    indirect = capacity_mwh * production_factors["battery"]["indirect_emissions"]["MWh"] * grid_emission["CO2"]
    direct = capacity_mwh * production_factors["battery"]["direct_emissions"]["CO2"]
    return indirect + direct


def battery_dismantling_emissions(vessel_class, dismantling_factors, grid_emission, battery_characteristics):
    """
    Computes CO emissions from battery dismantling by vessel class.

    Returns:
        float: Total CO-equivalent emissions (tons)
    """
    capacity_dict = battery_characteristics["capacity"]
    weight_per_mwh = battery_characteristics["weight"]
    capacity_mwh = capacity_dict.get(vessel_class, 0)
    battery_weight = capacity_mwh * weight_per_mwh

    total_energy = battery_weight * dismantling_factors["battery"]["indirect_emissions"]["MWh"]
    return total_energy * grid_emission["CO2"]


# === TRANSPORT EMISSIONS ===

def transport_emissions(materials, distance_km, transport_factor):
    """
    Computes emissions from transporting materials by sea/land.

    Returns:
        float: Emissions in tons CO-equivalent
    """
    total_tons = materials["steel_tons"] + materials["aluminium_tons"]
    emissions_grams = total_tons * distance_km * transport_factor
    return emissions_grams / 1e6  # grams to tons


# === OPERATIONAL PHASE ===

def fuel_cost(fuel_type, fuel_consumption, fuel_type_costs):
    """
    Compute operating fuel cost per quarter.

    Returns:
        float: Cost in €1000
    """
    return fuel_type_costs.get(fuel_type, 0) * fuel_consumption


def wtw_emissions(fuel_type, fuel_consumption, fuel_emission_factors, nox_factor, pm_factor, energy_grid_mix):
    """
    Computes well-to-wake (WTW) emissions from vessel operation.

    Returns:
        dict: {CO2: ton, NOX: ton, PM: ton}
    """
    emissions = {"CO2": 0, "NOX": 0, "PM": 0}
    grid_emission = energy_grid_mix()["grid_composition"]["clean_energy"]
    fuel_factors = fuel_emission_factors

    # Lower Heating Value for liquid fuels
    LHV_MWh = 43 / 3.6  # MWh/ton
    efficiency = 0.4

```

```
    if fuel_type == "Electricity":
        emissions["CO2"] = fuel_consumption * grid_emission["CO2"]
        emissions["NOX"] = fuel_consumption * grid_emission["NOX"]
        emissions["PM"] = fuel_consumption * grid_emission["PM"]
    else:
        co2_factor = fuel_factors[fuel_type]["CO2"]
        emissions["CO2"] = fuel_consumption * co2_factor
        emissions["NOX"] = fuel_consumption * LHV_MWh * nox_factor * efficiency
        emissions["PM"] = fuel_consumption * LHV_MWh * pm_factor * efficiency

    return emissions


# === ECONOMIC MODELING ===

def inflation(value, periods, inflation_rate):
    """
    Calculate inflation-adjusted value across time.

    Returns:
        dict: {quarter: value}
    """
    return {q: value * ((1 + inflation_rate) ** q) for q in range(periods)}


def depreciation(value, periods, depreciation_rate):
    """
    Calculate linear depreciation across quarters.

    Returns:
        dict: {quarter: value}
    """
    return {
        q: max(0, value * (1 - q * depreciation_rate))
        for q in range(periods)
    }
```

Final_calculations.py Code

```
"""
Final calculations for fleet renewal model - Port of Rotterdam
Structured for the preprocessing layer of the decision support tool.
Author: Jelmer Pentinga
"""
# === IMPORTS ===

from Preprocessing_input import (
    vessel_characteristics, vessel_costs, maintenance,
    depreciation_factors, inflation_factor, battery_costs,
    infrastructure_costs, battery_characteristics, infrastructure_characteristics,
    fuel_type_cost, transport_distances
)
from Emission_factors import (
    processing_emission_factors,
    production_emission_factors,
    dismantling_emission_factors,
    energy_grid_emission_factors,
    transport_emission_factors,
    fuel_emission_factors
)
from Intermediate_calculations import (
    processing_emissions,
    metal_production_emissions,
    battery_production_emissions,
    metal_dismantling_emissions,
    battery_dismantling_emissions,
    transport_emissions,
    wtw_emissions,
    inflation,
    depreciation,
```

```python
    fuel_cost
)

# === PARAMETERS ===
T = 100  # quarters (15 years)
J = 200  # asset planning horizon (residual)


# === LIFECYCLE COSTS ===

def calculate_lifecycle_costs(vessel_class, vessel_costs, battery_costs, infrastructure_costs,
                              depreciation_factors, inflation_factor,
                              maintenance, fuel_type, fuel_consumption, fuel_type_costs, J, T, initial_age=None):
    """
    Computes CAPEX and OPEX profiles for vessels, batteries, infrastructure and fuel.

    Returns:
        dict: Cost dictionaries per time period and asset class
    """

    def get_infra_type(vclass):
        return "BSM" if vclass in ["IRV", "PV", "NM"] else "Shorepower" if vclass in ["sPV", "SV"] else None

    def uses_battery(vclass):
        return vclass in ["IRV", "PV", "NM"]

    results = {}

    # Vessel costs
    results['v_acq'] = inflation(vessel_costs["acquisition"], T, inflation_factor)
    base_res = vessel_costs["residual"]
    raw_dep = depreciation(base_res, J, depreciation_factors["vessels"])
    if initial_age is not None:
        shifted_dep = {j: raw_dep[j - initial_age] for j in range(initial_age, J)}
    else:
        shifted_dep = raw_dep
    results['v_res'] = {
        t: {j: inflation(shifted_dep[j], t + 1, inflation_factor)[t] for j in shifted_dep}
        for t in range(T)}
    results['v_ope'] = inflation(vessel_costs["operating"], T, inflation_factor)
    results['v_ins'] = inflation(vessel_costs["insurance"], T, inflation_factor)
    results['v_maint'] = {
        t: {age: inflation(cost, t+1, inflation_factor)[t] for age, cost in maintenance}
        for t in range(T)
    }

    # Battery costs
    if uses_battery(vessel_class):
        results['b_acq'] = inflation(battery_costs["acquisition"], T, inflation_factor)
        results['b_ope'] = inflation(battery_costs["operating"], T, inflation_factor)
        b_res_dep = depreciation(battery_costs["residual"], J, depreciation_factors["batteries"])
        results['b_res'] = {
            t: {j: inflation(b_res_dep[j], t + 1, inflation_factor)[t] for j in b_res_dep}
            for t in range(T)
        }
    else:
        results['b_res'] = 0

    # Infrastructure costs
    infra_type = get_infra_type(vessel_class)
    if infra_type:
        results['i_acq'] = inflation(infrastructure_costs["acquisition"][infra_type], T, inflation_factor)
        results['i_ope'] = inflation(infrastructure_costs["operating"][infra_type], T, inflation_factor)
        i_res_dep = depreciation(infrastructure_costs["residual"][infra_type], J, depreciation_factors["infrastructure"])
```

```
        results['i_res'] = {
            t: {j: inflation(i_res_dep[j], t + 1, inflation_factor)[t] for j in i_res_dep}
            for t in range(T)
        }
    else:
        results['i_acq'] = results['i_ope'] = results['i_res'] = 0

    # Fuel costs
    base_fuel_cost = fuel_cost(fuel_type, fuel_consumption, fuel_type_costs)
    results['f_cost'] = inflation(base_fuel_cost, T, inflation_factor)

    return results


def run_cost_analysis():
    """
    Wrapper to compute lifecycle costs for all vessels.

    Returns:
        dict: Nested costs dictionary per vessel ID
    """
    costs_all = {}
    vc_data = vessel_characteristics()

    for v_id, vc in vc_data.items():
        costs_all[v_id] = calculate_lifecycle_costs(
            vc["class"],
            vessel_costs()[v_id],
            battery_costs(),
            infrastructure_costs(),
            depreciation_factors(),
            inflation_factor(),
            maintenance()[v_id],
            vc["specific_emissions"]["fuel_type"],
            vc["specific_emissions"]["fuel_consumption"],
            fuel_type_cost(),
            J,
            T,
            vc["initial_age"]
        )
    return costs_all


# === LIFECYCLE EMISSIONS ===
def run_emission_analysis(materials, vessel_class, fuel_type, fuel_consumption,
    fuel_emission_factors, nox_factor, pm_factor):
    """
    Computes total emissions: CTG, GTC, WTW across vessel, battery, and infrastructure.

    Returns:
        dict: {CTG, GTC, WTW emissions by component}
    """
    grid_mix = energy_grid_emission_factors()["grid_composition"]
    dists = transport_distances()
    transport_factor = transport_emission_factors()
    battery_chars = battery_characteristics()
    depr_factors = depreciation_factors()

    transport_dists = {
        "v_proc": dists["romania_rotterdam"],
        "v_prod": dists["local"],
        "v_dism": dists["local"],
        "b_prod": dists["norway_rotterdam"],
        "b_dism": dists["norway_rotterdam"],
        "i_proc": dists["norway_rotterdam"],
        "i_prod": dists["local"],
    }

    country_mix = {
        "proc_v": "EU",
        "prod_v": "EU",
```

```
        "dism_v": "EU",
        "prod_b": "EU",
        "dism_b": "EU",
        "i_proc": "EU",
        "i_prod": "EU"
    }
    # === Vessel emissions ===
    proc_v = processing_emissions(materials, processing_emission_factors(), grid_mix[
        country_mix["proc_v"]])
    prod_v = metal_production_emissions(materials, production_emission_factors(), grid_mix[
        country_mix["prod_v"]])
    dism_v = metal_dismantling_emissions(materials, dismantling_emission_factors(), grid_mix[
        country_mix["dism_v"]])

    v_ctg_transport = (
        transport_emissions(materials, transport_dists["v_proc"], transport_factor))
        #transport_emissions(materials, transport_dists["v_prod"], transport_factor)

    v_gtc_transport = transport_emissions(materials, transport_dists["v_dism"],
        transport_factor)

    v_ctg = proc_v + prod_v + v_ctg_transport
    v_gtc = dism_v + v_gtc_transport

    # === Battery emissions ===
    if vessel_class in ["IRV", "PV", "sPV", "SV", "NM"]:
        capacity_mwh = battery_chars["capacity"].get(vessel_class, 0)
        battery_weight = capacity_mwh * battery_chars["weight"]
        battery_materials = {"steel_tons": battery_weight, "aluminium_tons": 0}

        b_ctg = battery_production_emissions(vessel_class, production_emission_factors(),
            grid_mix[country_mix["prod_b"]], battery_chars)
        b_ctg += transport_emissions(battery_materials, transport_dists["b_prod"],
            transport_factor)

        b_gtc = battery_dismantling_emissions(vessel_class, dismantling_emission_factors(),
            grid_mix[country_mix["dism_b"]], battery_chars)
        b_gtc += transport_emissions(battery_materials, transport_dists["b_dism"],
            transport_factor)
    else:
        b_ctg = b_gtc = 0

    if vessel_class in ["sPV", "SV"]:
        v_ctg += b_ctg
        v_gtc += b_gtc
        b_ctg = b_gtc = 0

    # === Infrastructure emissions ===
    if vessel_class in ["IRV", "PV", "NM"]:
        infra_materials = infrastructure_characteristics()["BSM"]

        i_ctg = processing_emissions(infra_materials, processing_emission_factors(), grid_mix
            [country_mix["i_proc"]])
        i_ctg += metal_production_emissions(infra_materials, production_emission_factors(),
            grid_mix[country_mix["i_prod"]])
        i_ctg += transport_emissions(infra_materials, transport_dists["i_proc"],
            transport_factor)
        i_ctg += transport_emissions(infra_materials, transport_dists["i_prod"],
            transport_factor)
    else:
        i_ctg = 0

    # === Operational (WTW) emissions ===
    fuel_emissions = wtw_emissions(
        fuel_type, fuel_consumption, fuel_emission_factors,
        nox_factor, pm_factor, energy_grid_emission_factors
    )

    return {
        'v_ctg': depreciation(v_ctg, T, depr_factors["CO2"]),
        'v_gtc': depreciation(v_gtc, T, depr_factors["CO2"]),
```

```
        'b_ctg': depreciation(b_ctg, T, depr_factors["CO2"]),
        'b_gtc': depreciation(b_gtc, T, depr_factors["CO2"]),
        'i_ctg': depreciation(i_ctg, T, depr_factors["CO2"]),
        'f_e': {
            "CO2": depreciation(fuel_emissions["CO2"], T, depr_factors["CO2"]),
            "NOX": fuel_emissions["NOX"],
            "PM": fuel_emissions["PM"]
        }
    }

# === MAIN EXECUTION ===

if __name__ == "__main__":
    cost_results = run_cost_analysis()

    vc_data = vessel_characteristics()
    emission_results = {}

    for v_id, vessel in vc_data.items():
        emission_results[v_id] = run_emission_analysis(
            materials=vessel["materials"],
            vessel_class=vessel["class"],
            fuel_type=vessel["specific_emissions"]["fuel_type"],
            fuel_consumption=vessel["specific_emissions"]["fuel_consumption"],
            fuel_emission_factors=fuel_emission_factors(),
            nox_factor=vessel["specific_emissions"]["NOX"],
            pm_factor=vessel["specific_emissions"]["PM"]
        )
```

D

Multi-objective optimisation layer code

model_config.py

```
    # model_config.py
import numpy as np
def get_license_options():
    return {
        "WLSACCESSID": "f8df8428-a2a5-4d9f-a9a3-732d8c6502e6",
        "WLSSECRET": "4ae4a663-d32b-4259-b2d2-b95e691e0cac",
        "LICENSEID": 2647002
    }
MODEL_CONFIG = {
    "IRV": {
        "H":             "BSM",
        "battery":       True,
        "infra":         True,
        "name":          "modelIRV",
        "infra_factor": 0.6,
        "req_period":    "quarter",
        "res_period":    "quarter",
        "has_d_res":     True,
        "epsilon": {
            "_LCA": np.linspace(10049, 11629, 20).tolist(),
            "_LP":  np.linspace(9537, 101229, 20).tolist()
        }
    },
    "PV": {
        "H":             "BSM",
        "battery":       True,
        "infra":         True,
        "name":          "modelPV",
        "infra_factor": 0.6,
        "req_period":    "quarter",
        "res_period":    "year",
        "has_d_res":     True,
        "epsilon": {
            "_LCA": np.linspace(4983, 5659, 20).tolist(),
            "_LP":  np.linspace(1678, 25451, 20).tolist()
        }
    },
    "sPV": {
        "H":             "Shorepower",
        "battery":       False,
        "infra":         True,
        "name":          "modelsPV",
        "infra_factor": 0.75,
        "req_period":    "year",
```

```
        "has_d_res":    False,
        "epsilon": {
            "_LCA": np.linspace(290, 290, 20).tolist(),
            "_LP":  np.linspace(0, 0, 20).tolist()
        }
    },
    "SV": {
        "H":            "Shorepower",
        "battery":      False,
        "infra":        True,
        "name":         "modelSV",
        "infra_factor": 0.5,
        "req_period":   "year",
        "has_d_res":    False,
        "epsilon": {
            "_LCA": np.linspace(464, 464, 20).tolist(),
            "_LP":  np.linspace(1305, 1305, 20).tolist()
        }
    },
    "NM": {
        "H":            "BSM",
        "battery":      True,
        "infra":        True,
        "name":         "modelNM",
        "infra_factor": 0.5,
        "req_period":   "year",
        "epsilon": {
            "_LCA": np.linspace(992, 1199, 20).tolist(),
            "_LP":  np.linspace(0, 5204, 20).tolist()
        }
    },
    "RHIB": {
        "H":            False,
        "battery":      False,
        "infra":        False,
        "name":         "modelRHIB",
        "req_period":   "year",
        "has_d_res":    False,
        "epsilon": {
            "_LCA": np.linspace(35, 35, 20).tolist(),
            "_LP":  np.linspace(1809, 1809, 20).tolist()
        }
    },
}
```

RHS_input.py

```
"""
This module defines input index sets and right-hand-side (RHS) parameters
used in the fleet optimisation model, specific to each vessel class.
Author: Jelmer Pentinga
"""

def define_index_sets(vessel_class):
    """
    Defines index sets E (infrastructure units), I (vessels), J (age bins), T (time).

    Parameters:
    - vessel_class: str, one of {"IRV", "PV", "sPV", "SV", "NM", "RHIB"}

    Returns:
    - Tuple: (E, I, J, T)
        - E: range, infrastructure unit indices (always range(3))
        - I: range, vessel identifiers (class-specific)
        - J: range, age bins (class-specific)
        - T: range, time periods (always 10 quarters)
    """
    E = range(3)    # 3 infrastructure locations (e.g., base, forward, outer)
    T = range(100)   # 10 quarters (2.5 years horizon)

```

```
    index_map = {
        "IRV":   (range(13), range(200)),  # 13 vessels, 199 age bins
        "PV":    (range(6),  range(125)),
        "sPV":   (range(3),  range(140)),
        "SV":    (range(4),  range(110)),
        "NM":    (range(2),  range(150)),
        "RHIB":  (range(3),  range(110))
    }

    if vessel_class not in index_map:
        raise ValueError(f"Unknown vessel class: {vessel_class}")

    I, J = index_map[vessel_class]
    return E, I, J, T


def define_rhs_parameters(vessel_class):
    """
    Defines RHS parameters for operational and reserve vessel demands,
    as well as battery parameters.

    Parameters:
    - vessel_class: str, one of {"IRV", "PV", "sPV", "SV", "NM", "RHIB"}

    Returns:
    - Tuple: (d_req, d_res, b_eol, b_d)
        - d_req: int, required number of operational vessels per period
        - d_res: int or None, number of required reserve vessel periods per year
        - b_eol: int, battery end-of-life age
        - b_d: int, battery demand per vessel unit (scaled: op=2×b_d, res=1×b_d)
    """
    rhs_map = {
        "IRV":  {"d_req": 4, "d_res": 1, "b_d": 4},   # High redundancy
        "PV":   {"d_req": 2, "d_res": 2, "b_d": 4},   # Balanced ops and reserve
        "sPV":  {"d_req": 3, "d_res": None, "b_d": None},
        "SV":   {"d_req": 7, "d_res": None, "b_d": None},# High operations, no reserve
        "NM":   {"d_req": 2, "d_res": None, "b_d": 2},
        "RHIB": {"d_req": 2, "d_res": None, "b_d": None}
    }

    if vessel_class not in rhs_map:
        raise ValueError(f"Unknown vessel class: {vessel_class}")

    # Fixed across vessel types
    b_eol = 40  # Battery end-of-life age

    d_req = rhs_map[vessel_class]["d_req"]
    d_res = rhs_map[vessel_class]["d_res"]
    b_d =  rhs_map[vessel_class]["b_d"]
    return d_req, d_res, b_eol, b_d
```

build_model.py

```
"""
This module defines the Gurobi Optimisation model
Author: Jelmer Pentinga
"""
import gurobipy as gp
from gurobipy import GRB, quicksum
from model_config import MODEL_CONFIG


def compute_feasible_age_map(I, v_i_a, J, T, v_eol_a):
    """
    For each vessel i in I, return list of (age, time) pairs
    that respect end-of-life and time-step logic.
    """
    feasible = {}
    for i in I:
        if v_i_a[i] is not None:
```

```
            # existing vessels age in lock-step with t
            feasible[i] = [
                (v_i_a[i] + t, t)
                for t in T
                if v_i_a[i] + t <= v_eol_a[i]
            ]
        else:
            # new/second-hand: allow any purchase age   eol at any t
            feasible[i] = [
                (j, t)
                for t in T
                for j in J
                if j <= t
                   and j <= v_eol_a[i]
            ]
    return feasible


def filter_feasible_age_for_maintenance(feasible_age_map, v_maint_c):
    """
    Only keep (age, time) pairs for which maintenance data exists.
    v_maint_c[i][t] is a dict mapping age -> cost.
    """
    filtered = {}
    for i, pairs in feasible_age_map.items():
        valid = []
        for (j, t) in pairs:
            # first make sure we have an entry at time t
            if t not in v_maint_c[i]:
                continue
            # then make sure that at time t we have a cost for age j
            if j not in v_maint_c[i][t]:
                continue
            valid.append((j, t))
        filtered[i] = valid
    return filtered

def build_model(
    vessel_class,
    E, I, J, T,
    v_i_a, d_req, d_res,
    v_eol_a, v_maint_c,
    b_d, b_eol_a,
    params,
    model=None,
    epsilon_lca=None,
    epsilon_lp=None
):
    cfg = MODEL_CONFIG[vessel_class]
    if model is None:
        model = gp.Model(cfg["name"])

    # Compute age maps
    feasible_age = compute_feasible_age_map(I, v_i_a, J, T, v_eol_a)
    maint_age = filter_feasible_age_for_maintenance(feasible_age, v_maint_c)
    new_vessels = [i for i in I if v_i_a[i] is None]
    final_quarters = {t for t in T if t//4 == (T[-1]//4)}
    salv_cand = {}
    for i in I:
        if v_i_a[i] is not None:
            t_dead = v_eol_a[i] - v_i_a[i]
            salv_cand[i] = [
                (j,t) for (j,t) in feasible_age[i]
                if t <= t_dead]
        else:
            salv_cand[i] = [
                (j,t) for (j,t) in feasible_age[i]]

    # Vessel state variables: O, R, U, S
    v = {}
    v['O'] = model.addVars([(i, j, t) for i in I for j, t in feasible_age[i]],
```

```
        vtype=GRB.BINARY, name='O')
    v['R'] = model.addVars([(i, j, t) for i in I for j, t in feasible_age[i]],
        vtype=GRB.BINARY, name='R')
    v['U'] = model.addVars([(i, j, t) for i in I for j, t in maint_age[i]],
        vtype=GRB.BINARY, name='U')
    v['S'] = model.addVars([(i,j,t) for i in I for (j,t) in salv_cand[i]],vtype=GRB.BINARY,
        name="S")

    # Purchase decisions
    v['P'] = model.addVars([(i, t) for i in new_vessels for t in T],
        vtype=GRB.BINARY, name='P')

    # Battery variables
    if cfg['battery']:
        J_batt = [j for j in J if j <= b_eol_a]
        v['Q'] = model.addVars(T,lb=0, ub=70,vtype=GRB.INTEGER, name='Q')
        v['W'] = model.addVars([(j, t) for j in J_batt for t in T if j <= t],lb=0, ub=70,
            vtype=GRB.INTEGER, name='W')
        v['V'] = model.addVars([(j, t) for j in J_batt for t in T if j <= t],lb=0, ub=70,
            vtype=GRB.INTEGER, name='V')
        v['X'] = model.addVars([(b_eol_a, t) for t in T if t >= b_eol_a],lb=0, ub=70, vtype=
            GRB.INTEGER, name='X')

    # Infrastructure variables
    if cfg['infra']:
        h = cfg['H']
        v['Y'] = model.addVars([(h, t) for t in T],lb=0, ub=6,vtype=GRB.INTEGER, name='Y')
        v['Z'] = model.addVars([(h, j, t) for j in J for t in T],lb=0, ub=3,vtype=GRB.INTEGER
            , name='Z')

    # Call shared routines
    define_objective(model, v, params, cfg, E, I, J, T, v_i_a, v_eol_a,
                     epsilon_lca, epsilon_lp)
    add_constraints(model, v,
                    v_i_a, d_req, d_res,
                    v_eol_a, v_maint_c,
                    b_d, b_eol_a,
                    cfg, E, I, J, T)
    return model, v



def define_objective(
    model, v, p, cfg, E, I, J, T, v_i_a, v_eol_a,
    epsilon_lca=None, epsilon_lp=None
):
    """
    Build LCA, TCO, TCO_res, LP expressions and attach to model.
    """
    def get_keys_for(state, vessel):
    # return all (i,j,t) tuples for v[state] where i == vessel
        return [key for key in v[state].keys() if key[0] == vessel]
    T_max = max(T)
    new_vessels = [i for i in I if v_i_a[i] is None]
    feasible_age = compute_feasible_age_map(I, v_i_a, J, T, v_eol_a)
    known_age      = [i for i in I if v_i_a[i] is not None]

        # --- Insurance rate per vessel: new vs existing ---
    # Vessel CAPEX
    salv_known = quicksum(v['S'][i,j,t] * p['v_res_c'][i][0][j]
    for i in known_age for j,t in feasible_age[i] if (i,j,t) in v['S'])

    salv_new   = quicksum(v['S'][i,j,t] * p['v_res_c'][i][t-j][j]
        for i in new_vessels for j,t in feasible_age[i] if (i,j,t) in v['S'])

    vessel_capex = (
        quicksum(v['P'][i, t] * p['v_acq_c'][i][t] for i, t in v['P'].keys())
      - (salv_known + salv_new))

    # Vessel OPEX
    vessel_opex = (quicksum(v[state][i, j, t] * (
```

```
                p['v_om_c'][i][t] + p['v_ins_c'][i][t]
              + (p['v_maint_c'][i][t][j] if state == 'U' else 0)
            ) for state in ['O','R','U'] for i in known_age
            for (i2, j, t) in get_keys_for(state, i)) +
            quicksum(v[state][i, j, t] * (p['v_om_c'][i][t] + p['v_ins_c'][i][t - j]
              + (p['v_maint_c'][i][t][j] if state == 'U' else 0)
            )
            for state in ['O','R','U']
            for i in new_vessels
            for (i2, j, t) in get_keys_for(state, i)
        )
    )
    # Vessel Residual
    kept_keys = set(v['O'].keys()) | set(v['R'].keys()) | set(v['U'].keys())
    vessel_residual = quicksum(
        (
          (v['O'][i,j,T_max] if (i,j,T_max) in v['O'] else 0)
        + (v['R'][i,j,T_max] if (i,j,T_max) in v['R'] else 0)
        + (v['U'][i,j,T_max] if (i,j,T_max) in v['U'] else 0)
        )
        # if it was an existing vessel (initial age   None), purchase time = 0
        # else purchase time = T_max - j
        * (
            p['v_res_c'][i][0][j]
            if v_i_a[i] is not None
            else p['v_res_c'][i][T_max - j][j]
          )
        for (i, j, t) in kept_keys
        if t == T_max
    )

    # Vessel emissions
    vessel_ctg = quicksum(v['P'][i, t] * p['v_ctg_e'][i][t] for i, t in v['P'].keys())
    vessel_gtc = quicksum(v['S'][i, j, t] * p['v_gtc_e'][i][t] for i, j, t in v['S'].keys())
    # Battery
    batt_capex = 0
    batt_opex = 0
    batt_residual = 0
    batt_ctg = 0
    batt_gtc = 0
    if cfg['battery']:
        # Battery CAPEX
        batt_capex = (
            quicksum(v.get('Q',{}).get(t,0) * p['b_acq_c'][t]
                     for t in v.get('Q',{}).keys())
          - quicksum(
                # salvage at its "purchase" quarter -tj
                v.get('X',{}).get((j,t),0) * p['b_res_c'][t-j][j]
                for (j, t) in v.get('X',{}).keys()
            )
        )
        # Battery OPEX
        batt_opex = quicksum(v['W'][j, t] * p['b_om_c'][t] for j, t in v['W'].keys())
        # Battery Residual
        all_batt_keys = set(v['V'].keys()) | set(v['W'].keys())
        batt_residual = quicksum(
            (
                (v['V'][j, t] if (j, t) in v['V'] else 0)
              + (v['W'][j, t] if (j, t) in v['W'] else 0)
            )
            * p['b_res_c'][t - j][j]
            for (j, t) in all_batt_keys
            if t == T_max
        )
        # Battery emissions
        batt_ctg = quicksum(v['Q'][t] * p['b_ctg_e'][t] for t in v.get('Q', {}).keys())
        batt_gtc = quicksum(v['X'][j, t] * p['b_gtc_e'][t] for (j, t) in v.get('X', {}).keys
            ())
    # Infra
    infra_capex = 0
    infra_opex = 0
```

```
    infra_residual = 0
    infra_ctg = 0
    if cfg['infra']:
        #h = cfg['H']
        # Infra CAPEX
        infra_capex = quicksum(v['Y'][h, t] * p['i_acq_c'][t] for (h, t) in v['Y'].keys())
        # Infra OPEX
        infra_opex = quicksum(quicksum(v['Y'][h, tt] for (h, tt) in v['Y'].keys() if tt <= t) * p['i_om_c'][t]
            for (h, t) in v['Y'].keys())
        # Infra Residual
        infra_residual = quicksum(
            v['Z'][h, j, T_max] * p['i_res_c'][T_max - j][j]
            for h, j, t in v['Z'].keys()
            if t == T_max and (T_max - j) >= 0  # guard out negative indices
        )
        # Infra emissions
        infra_ctg = quicksum(v['Y'][h, t] * p['i_ctg_e'][t] for (h, t) in v['Y'].keys())

    # Fuel cost & emissions
    fuel_cost = quicksum(v['O'][i, j, t] * p['f_c'][i][t] for i, j, t in v['O'].keys())
    wtw_co2 = quicksum(v['O'][i, j, t] * p['f_e'][i]['CO2'][t] for i, j, t in v['O'].keys())
    wtw_nox = quicksum(v['O'][i, j, t] * p['f_e'][i]['NOX'] for i, j, t in v['O'].keys())
    wtw_pm  = quicksum(v['O'][i, j, t] * p['f_e'][i]['PM'] for i, j, t in v['O'].keys())

    # Metrics
    LCA     = vessel_ctg + vessel_gtc + wtw_co2 + infra_ctg + batt_ctg + batt_gtc
    TCO     = vessel_capex + vessel_opex + fuel_cost + infra_capex + infra_opex + batt_capex + batt_opex
    TCO_res = TCO - (vessel_residual + batt_residual + infra_residual)
    LP_nox  = wtw_nox
    LP_pm   = wtw_pm

    # Attach to model
    model._LCA     = LCA
    model._LPNOX   = LP_nox
    model._LPPM    = LP_pm
    model._TCO     = TCO
    model._TCO_res = TCO_res

def add_constraints(
    model, v, v_i_a, d_req, d_res,
    v_eol_a, v_maint_c,
    b_d, b_eol_a,
    cfg, E, I, J, T
):
    """
    Add operational, maintenance, battery, and infrastructure constraints.
    """
    feasible_age = compute_feasible_age_map(I, v_i_a, J, T, v_eol_a)
    maint_age = filter_feasible_age_for_maintenance(feasible_age, v_maint_c)
    new_vessels = [i for i in I if v_i_a[i] is None]
    known_age = [i for i in I if v_i_a[i] is not None]
    final_quarters = {t for t in T if t//4 == (T[-1]//4)}
    salv_cand = {}
    for i in I:
        if v_i_a[i] is not None:
            t_dead = v_eol_a[i] - v_i_a[i]
            salv_cand[i] = [
                (j,t) for (j,t) in feasible_age[i]
                if t <= t_dead]
        else:
            salv_cand[i] = [
                (j,t) for (j,t) in feasible_age[i]
                if t not in final_quarters]
    # === OPERATIONAL DEMAND ===
    occ_O = {t: [] for t in T}
    occ_R = {t: [] for t in T}
    for (i, j, t) in v['O'].keys(): occ_O[t].append((i, j))
    for (i, j, t) in v['R'].keys(): occ_R[t].append((i, j))

```

```
    if cfg['req_period'] == 'quarter':
        model.addConstrs(
            (quicksum(v['O'][i, j, t] for i, j in occ_O[t]) >= d_req for t in T),
            name="demand_operational_quarterly"
        )
    else:
        for y in range(len(T) // 4):
            quarters = [4 * y + q for q in range(4)]
            model.addConstr(
                quicksum(v['O'][i, j, t] for t in quarters for i, j in occ_O[t]) >= d_req,
                name=f"demand_operational_year_{y}"
            )

    if cfg.get('has_d_res', False):
        if cfg['res_period'] == 'quarter':
            model.addConstrs(
                (quicksum(v['R'][i, j, t] for i, j in occ_R[t]) >= d_res for t in T),
                name="demand_reserve_quarterly"
            )
        else:
            for y in range(len(T) // 4):
                quarters = [4 * y + q for q in range(4)]
                model.addConstr(
                    quicksum(v['R'][i, j, t] for t in quarters for i, j in occ_R[t]) >= d_res,
                    name=f"demand_reserve_year_{y}"
                )

    # === 1) MUTUAL EXCLUSION via SOS1 (at most one of O,R,U,S per (i,j,t)) ===
    for i, pairs in feasible_age.items():
        for (j, t) in pairs:
            svars = [v[s][i,j,t] for s in ('O','R','U','S') if (i,j,t) in v[s]]
            if len(svars) > 1:
                model.addSOS(GRB.SOS_TYPE1, svars)

    model.addConstrs(
        (quicksum(v[s][i, v_i_a[i], 0] for s in ['O', 'R', 'U', 'S']
                  if (i, v_i_a[i], 0) in v[s]) == 1
         for i in I if v_i_a[i] is not None), name="initial_state"
    )

    # === AGEING & ENTRY TRANSITIONS ===
    model.addConstrs(
        (quicksum(v[s][i, j + 1, t + 1] for s in ['O', 'R', 'U', 'S']
                  if (i, j + 1, t + 1) in v[s]) ==
         quicksum(v[s][i, j, t] for s in ['O', 'R', 'U', 'S']
                  if (i, j, t) in v[s]) - v['S'].get((i, j, t), 0)
         for i in I for (j, t) in feasible_age[i]
         if j + 1 <= v_eol_a[i] and t + 1 in T), name="ageing"
    )
    for (i, i_next) in zip(new_vessels, new_vessels[1:]):
        model.addConstr(
            quicksum(t * v['P'][i, t] for t in T)
          <= quicksum(t * v['P'][i_next, t] for t in T),
            name=f"symmetry_{i}_{i_next}"
        )
    model.addConstrs(
        (quicksum(v[s][i, 0, t + 1] for s in ["O", "R", "U", "S"]
                  if (i, 0, t + 1) in v[s]) == v["P"][i, t]
         for i in new_vessels for t in T if (i, t) in v["P"] and t + 1 in T),
        name="initial_age_entry_after_purchase"
    )

    model.addConstrs(
        (quicksum(v['P'][i, t] for t in T if (i, t) in v['P']) <= 1
         for i in new_vessels), name="unique_purchase"
    )

    model.addConstrs(
        (quicksum(v[s][i, j, t] for s in ["O", "R", "U", "S"] if (i, j, t) in v[s]) <=
         quicksum(v["P"][i, tp] for tp in T if tp < t and (i, tp) in v["P"])
```

```
        for i in new_vessels for (j, t) in feasible_age[i]),
        name="use_only_after_purchase_strict"
    )

    # === 2) CUMULATIVE SALVAGE & MAINTENANCE ===
    # 2b) existing vessels must pick exactly one slot in their EOL window
    for i in known_age:
        model.addConstr(
            quicksum(v['S'][i, j, t] for (j, t) in salv_cand[i]) == 1,
            name=f"salv_by_eol_initial_{i}"
        )

    # 2c) new vessels: same, but only if you actually buy them
    for i in new_vessels:
        for t in T:
            if (i, t) not in v['P']:
                continue

            t_salv = t + v_eol_a[i] + 1
            # only if that -salvageperiod is in your model
            if t_salv not in T or (i, v_eol_a[i], t_salv) not in v['S']:
                continue
            model.addConstr(
                v['S'][i, v_eol_a[i], t_salv] == v['P'][i, t],
                name=f"salv_by_eol_new_{i}_{t}"
            )

    # 2d) for every maintenance date on an existing hull, either you dry-dock or
    # '    youve already salvaged (including doing it _right_ on that quarter).
    for i in known_age:
        for (j, t_dead) in maint_age[i]:
            # build the list of all S[i,*,*]   t_dead
            salvage_by_then = [
                v['S'][i, jp, tp]
                for (jp, tp) in salv_cand[i]
                if tp <= t_dead
            ]
            model.addConstr(
                v['U'][i, j, t_dead]
                  + quicksum(salvage_by_then)
                == 1,
                name=f"maint_or_salv_known_{i}_{j}_{t_dead}"
            )

    # 2e) simplified "--every10years "maintenance for newly purchased hulls
    for i in new_vessels:
        # if you ever buy …it
        p_i = quicksum(v['P'][i, t] for t in T)
        # …then at each --maintenanceagegroup you must perform exactly one U
        for age in sorted({j for (j, _) in maint_age[i]}):
            times_at_that_age = [t for (j2, t) in maint_age[i] if j2 == age]
            if not times_at_that_age:
                continue
            model.addConstr(
                quicksum(v['U'][i, age, t] for t in times_at_that_age)
                == p_i,
                name=f"maint_new_exact_{i}_{age}"
            )

    # === BATTERY COMPOSITION ===
    if cfg['battery']:
        model.addConstrs(
            (v['V'][0, t] + v['W'][0, t] == v['Q'][t]
             for t in T if (0, t) in v['V']), name="batt_entry")
        model.addConstrs(
            (v['V'][j, t] + v['W'][j, t] == v['V'][j - 1, t - 1] + v['W'][j - 1, t - 1]
             for j in J if 1 <= j <= b_eol_a for t in T if t >= 1
             and (j, t) in v['V'] and (j - 1, t - 1) in v['V']),
            name="batt_ageing")
        z_batt = model.addVars(T, lb=0, vtype=GRB.INTEGER, name="z_batt")
        for t in T:
```

```
            expr = (
                2 * quicksum(
                    v['O'][i, j, t]
                    for i in new_vessels
                    for (j, tt) in feasible_age[i]
                    if tt == t and (i, j, t) in v['O']
                )
                + quicksum(
                    v['R'][i, j, t]
                    for i in new_vessels
                    for (j, tt) in feasible_age[i]
                    if tt == t and (i, j, t) in v['R']
                )
            )
            model.addConstr(
                z_batt[t] == expr,
                name=f"batt_zdef_{t}"
            )
            model.addConstr(
                quicksum(v['W'][j, t]
                         for j in J if (j, t) in v['W'])
                == b_d * z_batt[t],
                name=f"batt_use2_{t}"
            )
        b = b_eol_a
        for t in T:
            if t >= b:
                # salvage in t  = purchases in t-b
                model.addConstr(v['X'][b, t] == v['Q'][t - b],
                    name=f"batt_salv_{t}"
                )



    # === INFRASTRUCTURE COMPOSITION ===
    if cfg['infra']:
        h = cfg['H']
        i_d = cfg['infra_factor']
        model.addConstrs(
            (quicksum(v['Z'][h, j, t] for j in J if (h, j, t) in v['Z']) ==
             quicksum(v['Y'][h, tt] for tt in T if tt <= t)
             for t in T), name="infra_cap")
        u_op = model.addVars(T, lb=0, vtype=GRB.INTEGER, name="u_op")
        for t in T:
            model.addConstr(
                u_op[t] == quicksum(
                    v['O'][i, j, t]
                    for i in new_vessels
                    for (j, tt) in feasible_age[i]
                    if tt == t and (i, j, t) in v['O']
                ),
                name=f"uop_def_{t}"
            )
            model.addConstr(
                quicksum(v['Z'][cfg['H'], j, t]
                         for j in J if (cfg['H'], j, t) in v['Z'])
                >= i_d * u_op[t],
                name=f"infra_util2_{t}"
            )
        model.addConstrs(
            (v['Z'][h, 0, t] == v['Y'][h, t]
             for t in T if (h, 0, t) in v['Z']), name="infra_entry")
        model.addConstrs(
            (v['Z'][h, j, t] == v['Z'][j - 1, t - 1]
             for j in J for t in T if t >= 1 and (j, t) in v['Z'] and (j - 1, t - 1) in v['Z']),
            name="infa_ageing")
```

e_constraint.py

```python
"""
Handles epsilon-constraint loop for fleet renewal models.
Integrates plotting, data extraction, and infeasibility diagnosis.
Author: Jelmer Pentinga
"""

# === Imports ===
import os
import time
import datetime
import pandas as pd
#import matplotlib.pyplot as plt
import gurobipy as gp
from gurobipy import GRB

# === Local modules ===
#from plotter import run_all_plots_for_fleet
from model_config import MODEL_CONFIG, get_license_options
from Preprocessing_input import vessel_characteristics
from data_extract import run_all_data_extract_for_fleet
from combine_fleet_data import combine_fleet_data
try:
    SCRIPT_DIR = os.path.dirname(os.path.abspath(__file__))
except NameError:
    # __file__ 'wont exist in some -REPLsfall back to cwd
    SCRIPT_DIR = os.getcwd()


# === Helper: Pull epsilon grid per fleet ===
def build_epsilon_constraints():
    return {clas: MODEL_CONFIG[clas]["epsilon"] for clas in MODEL_CONFIG}



# === Main solver loop per fleet ===
def run_epsilon_loop(fleet_name, base_case, build_model_func, eps_grid, extract_func):
    vc_data = vessel_characteristics()
    results, detailed = [], []
    grid = [(e_lca, e_lp) for e_lca in eps_grid["_LCA"] for e_lp in eps_grid["_LP"]]

    with gp.Env(params=get_license_options()) as env:
        model = gp.Model(f"FleetModel_{fleet_name}", env=env)

        # === Build model and insert dummy epsilon RHS ===
        model, vars_ = build_model_func(
            vessel_class=fleet_name,
            E=base_case["E"],
            I=base_case["I"],
            J=base_case["J"],
            T=base_case["T"],
            v_i_a=base_case["v_i_a"],
            d_req=base_case["d_req"],
            d_res=base_case.get("d_res"),
            v_eol_a=base_case["v_eol_a"],
            v_maint_c=base_case["v_maint_c"],
            b_d=base_case.get("b_d"),
            b_eol_a=base_case.get("b_eol_a"),
            params=base_case,
            model=model,
            epsilon_lca=grid[0][0],
            epsilon_lp=grid[0][1]
        )

        model.setObjective(model._TCO, GRB.MINIMIZE)
        eps_constr_lca = model.addConstr(model._LCA <= grid[0][0], name="eps_lca")
        eps_constr_lp  = model.addConstr(model._LPNOX <= grid[0][1], name="eps_lp")


        # model.setObjectiveN(model._LCA,
        #             index=0,
        #             priority=2,      # highest priority
```

```
        #             name="LCA")
        # model.setObjectiveN(model._TCO,
        #                     index=1,
        #                     priority=1,      # lower priority
        #                     name="TCO")


        # model.setObjectiveN(model._LPNOX,
        #             index=0,
        #             priority=2,
        #             name="LP_NOx")
        # model.setObjectiveN(model._TCO,
        #                     index=1,
        #                     priority=1,
        #                     name="TCO")

        model.update()
                # 3) Tune on the first  -instance
        model.setParam("PrePasses",  2)
        model.setParam("Threads",    3)
        #model.setParam("Method",     2)
        #model.setParam("BranchDir",  1)
        model.setParam("CutPasses",  3)
        model.setParam("Aggregate",  2)
        model.setParam("OBBT", 0)
        model.setParam("Heuristics", 0.001)

        # 4) Now sweep all  -pairs with the tuned settings
        prev_sol = None
        # === Solve grid of epsilon values ===
        for idx, (eps_lca, eps_lp) in enumerate(grid):
            print(f" {fleet_name} run {idx+1}/{len(grid)} -  _LCA={eps_lca:.2f},  _LP={eps_lp:.2f}")
            eps_constr_lca.RHS = eps_lca
            eps_constr_lp.RHS  = eps_lp

                # Warm-start
            if prev_sol is not None:
                for v in model.getVars():
                    if v.VarName in prev_sol:
                        v.Start = prev_sol[v.VarName]

            model.optimize()
            status = model.Status

            results.append({
                "Fleet": fleet_name,
                "Run": idx,
                "_LCA": eps_lca,
                "_LP": eps_lp,
                "Status": status,
                "TCO_res": model._TCO_res.getValue() if status == GRB.OPTIMAL else None,
                "TCO":     model._TCO.getValue()     if status == GRB.OPTIMAL else None,
                "LCA":     model._LCA.getValue()     if status == GRB.OPTIMAL else None,
                "LPnox":   model._LPNOX.getValue()   if status == GRB.OPTIMAL else None,
                "LPpm":    model._LPPM.getValue()    if status == GRB.OPTIMAL else None
            })

            if status == GRB.INFEASIBLE:
                print(f" {fleet_name} is infeasible  (_LCA={eps_lca},  _LP={eps_lp})")
                prev_sol = None

                # model.computeIIS()
                # model.write(f"IIS_{fleet_name}_run{idx}.ilp")
                # for c in model.getConstrs():
                #     if c.IISConstr:
                #         print("  IIS constraint:", c.constrName)
                # for v in model.getVars():
                #     if v.IISLB or v.IISUB:
                #         print("  IIS variable:", v.varName)
                continue
```

```
            if status == GRB.OPTIMAL:
                vessel_names = [vc_data[v]['name'] for v in base_case['vids']]
                detail = extract_func(base_case, vars_, vessel_names, fleet_name, fleet_name)
                detailed.append((fleet_name, idx, eps_lca, eps_lp, detail))
                prev_sol = { v.VarName: v.X for v in model.getVars() }

    return results, detailed


# === Main controller for all fleets ===
def run_all_epsilon_loops(fleet_map):
    eps = build_epsilon_constraints()
    timestamp = datetime.datetime.now().strftime("%Y-%m-%d_%H-%M-%S")
    outdir = os.path.join(SCRIPT_DIR, timestamp)
    os.makedirs(outdir, exist_ok=True)

    all_results, all_detailed = [], []
    fleet_tables, fleet_times = {}, {}

    for fleet, (base, build_fn) in fleet_map.items():
        start = time.time()
        r, d = run_epsilon_loop(fleet, base, build_fn, eps[fleet],
            run_all_data_extract_for_fleet)
        fleet_times[fleet] = time.time() - start

        # Save fleet results
        df_fleet = pd.DataFrame(r)
        fleet_dir = os.path.join(outdir, fleet)
        os.makedirs(fleet_dir, exist_ok=True)
        df_fleet.to_csv(os.path.join(fleet_dir, f"{fleet}_epsilon.csv"), index=False)
        fleet_tables[fleet] = df_fleet

        for (_fleet, run_idx, eps_lca, eps_lp, detail) in d:
            if _fleet != fleet: continue
            run_dir = os.path.join(fleet_dir, f"Run_{run_idx}")
            os.makedirs(run_dir, exist_ok=True)

            # Save data
            detail["df_vessels"].to_csv(os.path.join(run_dir, "df_vessels.csv"), index=False)
            detail["df_cost_flat"].to_csv(os.path.join(run_dir, "df_cost.csv"), index=False)
            detail["df_emissions_flat"].to_csv(os.path.join(run_dir, "df_emissions.csv"),
                index=False)
            detail["df_assets"].to_csv(os.path.join(run_dir, "df_assets.csv"), index=False)

            # # Generate plots
            # figs = run_all_plots_for_fleet(
            #     f"{fleet}_Run_{run_idx}",
            #     detail["df_vessels"],
            #     detail["df_cost"],
            #     detail["df_emissions"],
            #     detail["df_assets"],
            #     base
            # )
            # for name, fig in figs.items():
            #     fig.savefig(os.path.join(run_dir, f"{name}.png"), bbox_inches="tight")
            #     plt.close(fig)

        all_results.extend(r)
        all_detailed.extend(d)

    # Save timing summary
    with open(os.path.join(outdir, "fleet_timing_summary.txt"), "w") as f:
        f.write("Fleet  -grid timing (seconds)\n------------------------------\n")
        for fleet, secs in fleet_times.items():
            f.write(f"{fleet:6s} : {secs:7.1f} s\n")

    # Save all fleet summary
    df_all = pd.DataFrame(all_results)
    df_all.to_csv(os.path.join(outdir, "epsilon_all_fleets.csv"), index=False)
```

```
    # === Combined summary and plots ===
    num_runs = len(next(iter(fleet_tables.values())))
    combined = []
    for run in range(num_runs):
        row = {"Run": run}
        feas = True
        sums = {"TCO": 0, "TCO_res": 0, "LCA": 0, "LPnox": 0, "LPpm": 0}
        for fleet, df in fleet_tables.items():
            sub = df[df.Run == run]
            if sub.empty or sub.iloc[0].Status != GRB.OPTIMAL:
                feas = False
                break
            for k in sums:
                sums[k] += sub.iloc[0][k]
        row.update(sums if feas else {k: None for k in sums})
        row["Status"] = "Feasible" if feas else "Infeasible"
        combined.append(row)

    df_comb = pd.DataFrame(combined)
    df_comb.to_csv(os.path.join(outdir, "epsilon_combined_summary.csv"), index=False)

    # === Pareto Plot ===
    # feas = df_comb[df_comb.Status == "Feasible"]
    # if not feas.empty:
    #     plt.figure(figsize=(8, 6))
    #     sc = plt.scatter(
    #         feas["LCA"], feas["LPnox"],
    #         c=feas["TCO"], cmap="RdYlGn_r",
    #         s=60, edgecolor="k"
    #     )
    #     plt.colorbar(sc, label="Summed TCO (×1000 €)")
    #     plt.xlabel("Total LCA (t  CO-eq)")
    #     plt.ylabel("Total  NO (kg)")
    #     plt.title("Combined Pareto: ΣLCA vs Σ NO (colorΣ=TCO)")
    #     plt.grid(True)
    #     plt.tight_layout()
    #     plt.savefig(os.path.join(outdir, "combined_pareto_LCA_LPnox_colorTCO.png"))
    #     plt.close()

    # === Combined data and plots for each run ===
    combined_dir = os.path.join(outdir, "combined")
    os.makedirs(combined_dir, exist_ok=True)

    for run_idx in range(num_runs):
        if df_comb.loc[run_idx, "Status"] != "Feasible":
            continue
        this_run = [d for (_f, r, _, _, d) in all_detailed if r == run_idx]
        if not this_run:
            continue

        all_H = {h for f in fleet_map if (h := MODEL_CONFIG[f].get("H"))}
        base_H = sorted(all_H)

        df_vessels, cost_data, emissions_data, df_assets = combine_fleet_data(this_run, base_H)
        run_dir = os.path.join(combined_dir, f"Run_{run_idx}")
        os.makedirs(run_dir, exist_ok=True)

        df_vessels.to_csv(os.path.join(run_dir, "combined_vessels.csv"), index=False)
        pd.DataFrame.from_dict(cost_data).to_csv(os.path.join(run_dir, "combined_costs.csv"))
        pd.DataFrame.from_dict(emissions_data).to_csv(os.path.join(run_dir, "combined_emissions.csv"))
        pd.DataFrame.from_dict(df_assets).to_csv(os.path.join(run_dir, "combined_assets.csv"))

        # figs = run_all_plots_for_fleet(
        #     fleet_label=f"Combined_Run_{run_idx}",
        #     df_vessels=df_vessels,
        #     cost_data=cost_data,
        #     emissions_data=emissions_data,
        #     df_assets=df_assets,
```

```
        #     base_case=next(iter(fleet_map.values()))[0]
        # )
        # for name, fig in figs.items():
        #     fig.savefig(os.path.join(run_dir, f"{name}.png"), bbox_inches="tight")
        #     plt.close(fig)

    return df_all, all_detailed
```

MOO.py

```

"""
Main script to execute epsilon-constraint based multi-objective optimization
for fleet renewal planning using cost and emission results.
Author: Jelmer Pentinga
"""

# === Imports ===
from Final_calculations import run_cost_analysis, run_emission_analysis
from Preprocessing_input import vessel_characteristics
from RHS_input import define_index_sets, define_rhs_parameters
from Emission_factors import fuel_emission_factors
from build_model import build_model
from e_constraint import run_all_epsilon_loops

# === Global emission data (fetched once) ===
emission_data = fuel_emission_factors()


# === Base Case Generator ===
def get_base_case():
    """
    Build and return the base-case parameters grouped by vessel class.
    Each class entry contains:
        - Index sets (E, I, J, T)
        - RHS parameters (d_req, d_res, b_d, b_eol)
        - Per-vessel input parameters for cost and emissions

    Returns:
        dict: base_case[class] -> data for optimization model
    """
    vc_data = vessel_characteristics()
    cost_results = run_cost_analysis()

    # Run emission analysis for each vessel
    emission_results = {
        v_id: run_emission_analysis(
            vc["materials"],
            vc["class"],
            vc["specific_emissions"]["fuel_type"],
            vc["specific_emissions"]["fuel_consumption"],
            emission_data,
            vc["specific_emissions"]["NOX"],
            vc["specific_emissions"]["PM"]
        )
        for v_id, vc in vc_data.items()
    }

    base_case = {}
    # Group vessels by class
    class_groups = {}
    for v_id, vc in vc_data.items():
        clss = vc["class"]
        class_groups.setdefault(clss, []).append(v_id)

    # Build base-case per class
    for vessel_class, vids in class_groups.items():
        E, I, J, T = define_index_sets(vessel_class)
        d_req, d_res, b_eol, b_d = define_rhs_parameters(vessel_class)

```

```
        entry = {
            "E": E, "I": I, "J": J, "T": T,
            "d_req": d_req,
            "vids": sorted(vids),
            "v_i_a": [vc_data[v]["initial_age"] for v in vids],
            "v_eol_a": [vc_data[v]["eol"] for v in vids],
            "v_acq_c": [cost_results[v]["v_acq"] for v in vids],
            "v_ins_c": [cost_results[v]["v_ins"] for v in vids],
            "v_res_c": [cost_results[v]["v_res"] for v in vids],
            "v_om_c": [cost_results[v]["v_ope"] for v in vids],
            "v_maint_c": [cost_results[v]["v_maint"] for v in vids],
            "f_c": [cost_results[v]["f_cost"] for v in vids],
            "v_ctg_e": [emission_results[v]["v_ctg"] for v in vids],
            "v_gtc_e": [emission_results[v]["v_gtc"] for v in vids],
            "f_e": [emission_results[v]["f_e"] for v in vids],
        }

        if vessel_class in ["IRV", "PV"]:
            entry["d_res"] = d_res

        if vessel_class in ["IRV", "PV", "NM"]:
            entry["b_eol_a"] = b_eol
            entry["b_d"] = b_d
            entry["b_acq_c"] = cost_results[vids[0]]["b_acq"]
            entry["b_om_c"] = cost_results[vids[0]]["b_ope"]
            entry["b_res_c"] = cost_results[vids[0]]["b_res"]
            entry["b_ctg_e"] = emission_results[vids[0]]["b_ctg"]
            entry["b_gtc_e"] = emission_results[vids[0]]["b_gtc"]

        if vessel_class != "RHIB":
            entry["i_acq_c"] = cost_results[vids[0]]["i_acq"]
            entry["i_om_c"] = cost_results[vids[0]]["i_ope"]
            entry["i_res_c"] = cost_results[vids[0]]["i_res"]
            entry["i_ctg_e"] = emission_results[vids[0]]["i_ctg"]
            # Note: i_gtc not available in emission_results

        base_case[vessel_class] = entry

    return base_case


# === Main Execution ===
if __name__ == "__main__":
    print (" Generating base case data...")
    base_cases = get_base_case()

    fleet_map = {
        clas: (base_cases[clas], build_model)
        for clas in base_cases
    }

    print (" Running  -constraint loops for all fleets...")
    df_epsilon_results, all_detailed_data = run_all_epsilon_loops(fleet_map)
    print ("  -constraint analysis complete.")
```

E
Data extraction code

data_extract.py

```
# data_extract.py
import pandas as pd
from model_config import MODEL_CONFIG
def safe_val(d, key):
    """Extract .X from a Gurobi var if present, else return 0/float."""
    if d is None:
        return 0
    v = d.get(key, 0)
    return v.X if hasattr(v, "X") else v

def build_vessel_state_dataframe(I, J, T, P, O, R, S, U, vessel_names, class_label=None):
    rows = []
    for t in T:
        for i in I:
            vessel = vessel_names[i]
            state = None
            age  = None
            # purchase
            if safe_val(P, (i, t)) > 0.5:
                state = "Purchased"
            # then check the other four mutually-exclusive states
            for j in J:
                if safe_val(O, (i, j, t)) > 0.5:
                    state, age = "Operational", j
                    break
                if safe_val(R, (i, j, t)) > 0.5:
                    state, age = "Reserve", j
                    break
                if safe_val(U, (i, j, t)) > 0.5:
                    state, age = "Maintained", j
                    break
                if safe_val(S, (i, j, t)) > 0.5:
                    state, age = "Salvaged", j
                    break
            rows.append({
                "Time": t,
                "Vessel": vessel,
                "Class": class_label,
                "State": state,
                "Age": age
            })
    df = pd.DataFrame(rows)
    df["Label"] = df.apply(
        lambda r: f"{r['Vessel']} (Class {r['Class']})"
                  if pd.notna(r["Class"]) else f"{r['Vessel']} ( Not purchased)",
        axis=1
    )
```

```
    return df
def extract_cost_data(
    I, J, T, H,
    P, O, R, S, U,
    Q=None, V=None, W=None, X=None, Y=None, Z=None,
    v_acq_c=None, v_res_c=None, v_om_c=None,
    v_ins_c=None, v_maint_c=None, f_c=None,
    i_acq_c=None, i_om_c=None, i_res_c=None,
    b_acq_c=None, b_om_c=None, b_res_c=None,
    initial_age=None,
    b_eol_a=None
):
    """
    initial_age: list or dict mapping vessel i -> initial_age or None
    """
    T_max = max(T)

    # --- Vessel CAPEX: acquisitions minus salvage ---
    # split salvage into known vs new
    def safe(vdict, key): return vdict[key] if key in vdict else 0

    salv_known = {
        t: sum(
            safe_val(S, (i, j, t)) * v_res_c[i][0].get(j, 0)
            for i in I if initial_age[i] is not None
            for j in J
            if (i, j, t) in S
        )
        for t in T
    }
    salv_new = {
        t: sum(
            safe_val(S, (i, j, t)) * v_res_c[i].get(t - j, {}).get(j, 0)
            for i in I if initial_age[i] is None
            for j in J
            if (i, j, t) in S and (t - j) >= 0
        )
        for t in T
    }

    vessel_capex = {
        t: sum(
              safe_val(P, (i, t)) * v_acq_c[i][t]
            for i in I
        )
        - salv_known.get(t, 0)
        - salv_new.get(t, 0)
        for t in T
    }

        # --- Vessel OPEX: O&M + insurance + maintenance ---
    vessel_opex = {
        t: sum(
            safe_val(var, (i, j, t))
            * (
                # O&M
                v_om_c[i][t]
                # insurance: at t if existing, at t-j if new
                + (
                    v_ins_c[i][t]
                    if initial_age[i] is not None
                    else v_ins_c[i].get(t-j, 0)
                  )
                # maintenance only when in U
                + (
                    v_maint_c[i][t].get(j, 0)
                    if state == 'U'
                    else 0
                  )
            )
            for state, var in (('O', O), ('R', R), ('U', U))
```

```
            for i in I
            for j in J
            if (i, j, t) in var
        )
        for t in T
    }
    # --- Residual values at T_max ---
    vessel_residual = sum(
        (
            safe_val(O, (i, j, T_max))
          + safe_val(R, (i, j, T_max))
          + safe_val(U, (i, j, T_max))
        )
        * (
            # existing: purchase at t=0
            v_res_c[i][0].get(j, 0)
            if initial_age[i] is not None
            # new: purchase at T_max-j
            else v_res_c[i].get(T_max - j, {}).get(j, 0)
        )
        for i in I for j in J
        if (i, j, T_max) in O or (i, j, T_max) in R or (i, j, T_max) in U
    )

    # --- Infra CAPEX & OPEX (unchanged) ---
    infra_capex = {
        t: sum(safe_val(Y, (h, t)) * i_acq_c[t] for h in H) if i_acq_c else 0
        for t in T
    }
    infra_opex = {
        t: sum(sum(safe_val(Y, (h, tt)) for tt in T if tt <= t) * i_om_c[t] for h in H) if i_om_c else 0
        for t in T
    }

    # --- Infra Residual at T_max (new logic) ---
    infra_residual = (
        sum(
            safe_val(Z, (h, j, T_max))
            * i_res_c.get(T_max - j, {}).get(j, 0)
            for h in H
            for j in J
            if (h, j, T_max) in Z
        )
        if i_res_c
        else 0
    )
    # --- Battery CAPEX & OPEX & Residual ---
    batt_salvage = {t: sum(safe_val(X, (j,t)) * b_res_c.get(t-j, {}).get(j, 0)
        for j in J
        if (j,t) in (X or {})
    )
    for t in T
}
    batt_capex = {
        t: safe_val(Q, t) * b_acq_c[t] - batt_salvage.get(t, 0)
        for t in T
    }  if b_acq_c else {}
    batt_opex = {
        t: sum(safe_val(W, (j, t)) * b_om_c[t] for j in range(b_eol_a + 1))
        for t in T
    } if b_om_c else {}

    batt_residual = sum(
        (safe_val(V, (j, T_max)) + safe_val(W, (j, T_max)))
        * b_res_c.get(T_max - j, {}).get(j, 0)
        for j in range(b_eol_a + 1)
        if (j, T_max) in V and (j, T_max) in W
    ) if b_res_c else 0

    # --- Fuel costs ---
```

```
    fuel_cost = {
        t: sum(safe_val(O, (i, j, t)) * f_c[i][t] for i in I for j in J if (i, j, t) in O)
        for t in T
    }

    return {
        "vessel_capex":        vessel_capex,
        "vessel_opex":        vessel_opex,
        "vessel_residual":    vessel_residual,
        "infra_capex":        infra_capex,
        "infra_opex":         infra_opex,
        "infra_residual":     infra_residual,
        "batt_capex":         batt_capex,
        "batt_opex":          batt_opex,
        "batt_residual":      batt_residual,
        "fuel_cost":          fuel_cost
    }

def extract_emissions_data(
    E, I, J, T,
    P, O, R, S,
    Q=None, X=None, Y=None,
    f_e=None, v_ctg_e=None, v_gtc_e=None,
    i_ctg_e=None, i_gtc=None, b_ctg_e=None, b_gtc_e=None
):
    # --- Vessel manufacturing (CTG) ---
    vessel_ctg = {
        t: sum(safe_val(P, (i, t)) * v_ctg_e[i][t] for i in I)
        for t in T
    }

    # --- Infra CTG ---
    infra_ctg = {
        t: sum(safe_val(Y, ("BSM", t)) * i_ctg_e[t]
            for (h, tt) in (Y or {}).keys()
            if tt == t
            )
        if i_ctg_e else 0
        for t in T
    }
        # --- Battery CTG ---
    batt_ctg = {
        t: safe_val(Q, t) * b_ctg_e[t] if b_ctg_e else 0
        for t in T
    }

    # --- Vessel end-of-life (GTC) ---
    vessel_gtc = {
        t: sum(safe_val(S, (i, j, t)) * v_gtc_e[i][t] for i in I for j in J)
        for t in T
    }

    # --- Infra GTC
    infra_gtc = {
        t: sum(safe_val(Y, ("BSM", t)) * i_gtc[t] for h in Y.keys())
        if i_gtc else 0
        for t in T
    }

    # --- Battery GTC
    batt_gtc = {
        t: sum(safe_val(X, (j, t)) * b_gtc_e[t] for j in J if j <= t)
        if b_gtc_e else 0
        for t in T
    }

    # --- Well-to-wheel (fuel) ---
    wtw = {}
    # f_e is a list (indexed by vessel i) of dicts keyed by pollutant name
    for pollutant in next(iter(f_e)).keys():
        series = {}
```

```
        for t in T:
            s = 0.0
            for i in I:
                for j in J:
                    if (i, j, t) in O and safe_val(O, (i, j, t)) > 0:
                        val = f_e[i][pollutant]
                        # if it's -timeseries, look up t; otherwise it's constant
                        s += safe_val(O, (i, j, t)) * (val[t] if isinstance(val, dict) else
                            val)
            series[t] = s
        wtw[pollutant] = series

    return {
        "vessel_ctg": vessel_ctg,
        "infra_ctg":  infra_ctg,
        "batt_ctg":   batt_ctg,
        "vessel_gtc": vessel_gtc,
        "infra_gtc":  infra_gtc,
        "batt_gtc":   batt_gtc,
        "wtw":        wtw
    }

def extract_acqinfra_data(T, H, Q=None, Y=None):
    return {
        "Infrastructure": {
            t: {h: safe_val(Y, (h, t)) for (h_,tt_) in (Y or {}) if tt_ == t for h in [h_]}
            for t in T
        },
        "Batteries": {
            t: safe_val(Q, t) for t in T
        }
    }

def extract_opinfra_data(T, H, V, W, Z):
    """
    Extracts operational infrastructure & battery data, safely handling fleets
    without batteries (V or W may be None) or without decommissioning (Z may be None).
    """
    # If no battery vars exist, treat as empty
    Wkeys = W.keys() if W is not None else []
    Vkeys = V.keys() if V is not None else []
    Zkeys = Z if Z is not None else []

    # Infrastructure in operation
    infra_op = {}
    for t in T:
        infra_op[t] = {
            h: sum(
                safe_val(Z, (hh, j, tt))
                for (hh, j, tt) in Zkeys
                if hh == h and tt == t
            )
            for h in H
        }

    # Batteries in operation / non-op
    batt_in_op = {}
    nonop_batt = {}
    for t in T:
        batt_in_op[t] = sum(
            safe_val(W, (j, tt))
            for (j, tt) in Wkeys
            if tt == t
        )
        nonop_batt[t] = sum(
            safe_val(V, (j, tt))
            for (j, tt) in Vkeys
            if tt == t
        )

    return {
```

```
        'Infrastructure': infra_op,
        'Batteries in Operation': batt_in_op,
        'NonOperational Batteries': nonop_batt
    }

def extract_salvaged_batteries_data(J, T, X=None):
    return {
        t: sum(safe_val(X, (j, t)) for j in J if (j, t) in (X or {}))
        for t in T
    }

def unify_acq_op_salv_data(T, H, J, acq_data, op_data, sal_data):
    recs = []
    for t in T:
        # infra acquisition
        for h in H:
            recs.append(dict(
                Time=t, Type="Acquisition", Asset="Infra", Name=h,
                Value=acq_data["Infrastructure"][t][h]
            ))
        # battery acquisition
        recs.append(dict(
            Time=t, Type="Acquisition", Asset="Battery", Name=None,
            Value=acq_data["Batteries"][t]
        ))
        # infra operation
        for h in H:
            recs.append(dict(
                Time=t, Type="Installed", Asset="Infra", Name=h,
                Value=op_data["Infrastructure"][t][h]
            ))
        # battery operation
        recs.append(dict(
            Time=t, Type="Operation", Asset="Battery", Name=None,
            Value=op_data["Batteries in Operation"][t]
        ))
        # non-op batteries
        recs.append(dict(
            Time=t, Type="NonOperational Batteries", Asset="Battery", Name=None,
            Value=op_data.get("NonOperational Batteries", {}).get(t, 0)
        ))
        # salvaged batteries
        recs.append(dict(
            Time=t, Type="Salvaged", Asset="Battery", Name=None,
            Value=sal_data[t]
        ))
    return pd.DataFrame(recs)

def flatten_cost_dict_to_df(cost_dict):
    rows = []
    for cat, series in cost_dict.items():
        if isinstance(series, dict):
            for t, v in series.items():
                rows.append({"Time": t, "Category": cat, "Value": v})
        else:
            rows.append({"Time": "final", "Category": cat, "Value": series})
    return pd.DataFrame(rows)

def flatten_emissions_dict_to_df(em_dict):
    rows = []
    for cat, data in em_dict.items():
        if isinstance(data, dict):
            for t, v in data.items():
                rows.append({"Time": t, "Category": cat, "Value": v})
        else:
            # pollutant-specific nested
            for pol, series in data.items():
                for t, v in series.items():
                    rows.append({"Time": t, "Category": f"wtw_{pol}", "Value": v})
    return pd.DataFrame(rows)

```

```python
def run_all_data_extract_for_fleet(base_case, variables, vessel_names, fleet_label, class_label=None):
    df_vessels = build_vessel_state_dataframe(
        base_case["I"], base_case["J"], base_case["T"],
        variables["P"], variables["O"], variables["R"],
        variables["S"], variables["U"],
        vessel_names, class_label
    )
    df_vessels["model"] = fleet_label
    # grab raw H from the config
    raw_H = MODEL_CONFIG[fleet_label].get("H", [])
    # normalize to always be a list of infra names (or empty)
    if not raw_H:
        H = []
    elif isinstance(raw_H, list):
        H = raw_H
    else:
        H = [raw_H]

    df_cost      = extract_cost_data(
        I=base_case["I"], J=base_case["J"], T=base_case["T"], H=H,
        P=variables["P"], O=variables["O"], R=variables["R"],
        S=variables["S"], U=variables["U"],
        Q=variables.get("Q"), V=variables.get("V"),
        W=variables.get("W"), X=variables.get("X"),
        Y=variables.get("Y"), Z=variables.get("Z"),
        v_acq_c=base_case["v_acq_c"],
        v_res_c=base_case["v_res_c"],
        v_om_c=base_case["v_om_c"],
        v_ins_c=base_case["v_ins_c"],
        v_maint_c=base_case["v_maint_c"],
        f_c=base_case["f_c"],
        i_acq_c=base_case.get("i_acq_c"),
        i_om_c=base_case.get("i_om_c"),
        i_res_c=base_case.get("i_res_c"),
        b_acq_c=base_case.get("b_acq_c"),
        b_om_c=base_case.get("b_om_c"),
        b_res_c=base_case.get("b_res_c"),
        initial_age=base_case["v_i_a"],
        b_eol_a=base_case.get("b_eol_a")
    )

    df_emissions = extract_emissions_data(
        E=base_case["E"], I=base_case["I"], J=base_case["J"], T=base_case["T"],
        P=variables["P"], O=variables["O"], R=variables["R"], S=variables["S"],
        Q=variables.get("Q"), X=variables.get("X"), Y=variables.get("Y"),
        f_e=base_case["f_e"],
        v_ctg_e=base_case["v_ctg_e"],
        v_gtc_e=base_case["v_gtc_e"],
        i_ctg_e=base_case.get("i_ctg_e"),
        i_gtc=base_case.get("i_gtc"),
        b_ctg_e=base_case.get("b_ctg_e"),
        b_gtc_e=base_case.get("b_gtc_e"),
    )
    df_acqinfra = extract_acqinfra_data(
     T=base_case["T"],
     H=H,                          # guaranteed iterable
     Q=variables.get("Q"),
     Y=variables.get("Y")
)
    df_opinfra = extract_opinfra_data(
        T=base_case["T"],
        H=H,
        V=variables.get("V"),
        W=variables.get("W"),
        Z=variables.get("Z")
)
    df_salvaged = extract_salvaged_batteries_data(
        J=base_case["J"], T=base_case["T"], X=variables.get("X")
    )
    df_assets   = unify_acq_op_salv_data(
```

```
        T=base_case["T"], H=H, J=base_case["J"],
        acq_data=df_acqinfra, op_data=df_opinfra, sal_data=df_salvaged
    )

    return {
        "df_vessels":      df_vessels,
        "df_cost":         df_cost,
        "df_emissions":    df_emissions,
        "df_acqinfra":     df_acqinfra,
        "df_opinfra":      df_opinfra,
        "df_salvaged":     df_salvaged,
        "df_assets":       df_assets,
        "df_cost_flat":    flatten_cost_dict_to_df(df_cost),
        "df_emissions_flat": flatten_emissions_dict_to_df(df_emissions),
    }
```

combine_fleet_data.py

```
# combine_fleet_data.py
import pandas as pd
from collections import defaultdict
from data_extract import unify_acq_op_salv_data
def combine_fleet_data(fleet_datasets, base_H):
    """
    fleet_datasets: list of dicts, each containing keys
      df_vessels, df_cost, df_emissions, df_acqinfra, df_opinfra, df_salvaged
    base_H: list of infrastructure names (e.g. ["BSM","Shorepower",...])
    """
    # === Combine vessel states ===
    df_all_vessels = pd.concat(
        [fleet["df_vessels"] for fleet in fleet_datasets],
        ignore_index=True
    )

    # === Combine cost data ===
    cost_data_combined = {}
    for fleet in fleet_datasets:
        for key, series in fleet["df_cost"].items():
            if isinstance(series, dict):
                cost_data_combined.setdefault(key, {})
                for t, value in series.items():
                    cost_data_combined[key][t] = cost_data_combined[key].get(t, 0.0) + value
            else:
                cost_data_combined[key] = cost_data_combined.get(key, 0.0) + series

    # === Combine emissions data ===
    emissions_data_combined = defaultdict(dict)
    for fleet in fleet_datasets:
        for key, series in fleet["df_emissions"].items():
            if key != "wtw":
                for t, value in series.items():
                    emissions_data_combined[key][t] = (
                        emissions_data_combined[key].get(t, 0.0) + value
                    )

    # Combine WTW emissions separately
    wtw_combined = defaultdict(lambda: defaultdict(float))
    for fleet in fleet_datasets:
        for pollutant, series in fleet["df_emissions"].get("wtw", {}).items():
            for t, val in series.items():
                wtw_combined[pollutant][t] += val
    emissions_data_combined["wtw"] = {
        pollutant: dict(series) for pollutant, series in wtw_combined.items()
    }

    # === Combine acquisition infrastructure & batteries ===
    acquisition_data = {
        "Infrastructure": defaultdict(dict),
        "Batteries":     defaultdict(float)
    }
```

```
    for fleet in fleet_datasets:
        infra = fleet["df_acqinfra"]["Infrastructure"]
        for t, h_vals in infra.items():
            for h, val in h_vals.items():
                acquisition_data["Infrastructure"][t][h] = (
                    acquisition_data["Infrastructure"][t].get(h, 0.0) + val
                )
        for t, val in fleet["df_acqinfra"]["Batteries"].items():
            acquisition_data["Batteries"][t] += val

    # === Combine operational infra, batteries, salvage ===
    operation_data = {
        "Infrastructure": defaultdict(lambda: defaultdict(float)),
        "Batteries in Operation": defaultdict(float),
        "NonOperational Batteries": defaultdict(float),
        "Salvaged": defaultdict(float)
    }
    for fleet in fleet_datasets:
        # infra operation
        for t, h_vals in fleet["df_opinfra"]["Infrastructure"].items():
            for h in base_H:
                operation_data["Infrastructure"][t][h] += h_vals.get(h, 0.0)
        # batteries in operation
        for t, val in fleet["df_opinfra"].get("Batteries in Operation", {}).items():
            operation_data["Batteries in Operation"][t] += val
        # non-operational batteries
        for t, val in fleet["df_opinfra"].get("NonOperational Batteries", {}).items():
            operation_data["NonOperational Batteries"][t] += val
        # salvaged batteries
        for t, val in fleet["df_salvaged"].items():
            operation_data["Salvaged"][t] += val

    # === Convert all defaultdicts to plain dicts ===
    # cost_data_combined: series of dicts or scalars
    cost_data_combined = {
        k: dict(v) if isinstance(v, dict) else v
        for k, v in cost_data_combined.items()
    }

    emissions_data_combined = {
        k: dict(v) if isinstance(v, dict) else v
        for k, v in emissions_data_combined.items()
    }

    acquisition_data = {
        "Infrastructure": dict(acquisition_data["Infrastructure"]),
        "Batteries":     dict(acquisition_data["Batteries"])
    }

    operation_data = {
        "Infrastructure": {
            t: dict(hvals)
            for t, hvals in operation_data["Infrastructure"].items()
        },
        "Batteries in Operation": dict(operation_data["Batteries in Operation"]),
        "NonOperational Batteries": dict(operation_data["NonOperational Batteries"]),
        "Salvaged": dict(operation_data["Salvaged"])
    }

 # === NEW: Build df_assets ===
    # T is just the sorted -timepoints you have in acq infra:
    T = sorted(acquisition_data["Infrastructure"].keys())

    # the salvage series you already accumulated lives in operation_data["Salvaged"]
    salvaged_data = operation_data["Salvaged"]

    df_assets = unify_acq_op_salv_data(
        T    = T,
        H    = base_H,
        J    = None,                # `unify_acq_op_salv_data` doesn't actually use J
        acq_data = acquisition_data,
```

```
        op_data  = operation_data,
        sal_data = salvaged_data
    )

    return (
        df_all_vessels,
        cost_data_combined,
        emissions_data_combined,
        df_assets                  # ← return it as the sixth element
    )
```

plotter.py

```
# plotter.py
"""
Created on Thu Jun 12 13:26:21 2025

@author: penti
"""

# plotter.py
# -*- coding: utf-8 -*-
"""
Revised plotting routines for the modular -fleetmodel workflow.
"""

import os
import matplotlib.pyplot as plt
import pandas as pd
from scipy.io import savemat
plt.rcParams.update({
    'font.size':          14,   # default text size
    'axes.titlesize':     18,   # axes title
    'axes.labelsize':     16,   # x/y labels
    'xtick.labelsize':    12,   # tick labels
    'ytick.labelsize':    12,
    'legend.fontsize':    14,
    'legend.title_fontsize': 14,
    'figure.titlesize':   18    # if you ever use fig.suptitle()
})

# these two dicts are populated by run_all_plots_for_fleet()
figures = {}
figure_data = {}

# Global color mappings
STATE_COLORS = {
    "Purchased":   "#6699cc",
    "Operational": "#2ca02c",
    "Reserve":     "#ffe680",
    "Maintained":  "#ff8c00",
    "Salvaged":    "#d62728"
}

def _year_ticks(T, start_year=2026):
    # pick every 4th quarter (including t=0)
    years = sorted(t for t in T if t % 4 == 0)
    labels = [str(start_year + (t // 4)) for t in years]
    return years, labels
def plot_vessel_gantt_and_age(df_vessels, fleet_label, start_year=2026):
    """
    1) -brokenbarh Gantt of states over time
    2) line plot of age vs time
    """
    # collect -timepoints and labels
    T = sorted(df_vessels["Time"].unique())
    labels = df_vessels["Label"].unique()

    # identify the "year "boundaries in T (every 4 quarters)
    year_ticks = [t for t in T if t % 4 == 0]
```

```
    year_labels = [str(start_year + t // 4) for t in year_ticks]

    # --- Gantt ---
    fig, ax = plt.subplots(figsize=(16, 8))
    for idx, lbl in enumerate(labels):
        sub = df_vessels[df_vessels["Label"] == lbl]
        y0 = idx * 10
        for state, grp in sub.groupby("State"):
            times = grp["Time"].values
            if times.size:
                ax.broken_barh(
                    [(t, 1) for t in times],
                    (y0, 8),
                    facecolors=STATE_COLORS.get(state, "#999999")
                )

    # y-axis
    year_ticks = year_ticks[::2]
    year_labels = year_labels[::2]

    ax.set_xticks(year_ticks)
    ax.set_xticklabels(year_labels)
    ax.set_xlim(year_ticks[0], T[-1])
    ax.set_yticks([i*10 + 4 for i in range(len(labels))])
    ax.set_yticklabels(labels)

    # x-axis: years only
    ax.set_xlim(year_ticks[0], T[-1])
    ax.set_xlabel("Year")
    ax.set_title(f"{fleet_label} Vessel State Timeline")

    # legend
    handles = [plt.Line2D([0],[0], color=c, lw=6) for c in STATE_COLORS.values()]
    ax.legend(handles, STATE_COLORS.keys(), title="State",
              bbox_to_anchor=(1.05,1), loc="upper left")

    ax.grid(True)
    fig.tight_layout(rect=[0,0,0.85,1])
    figures["gantt_vessel_timeline"] = fig

    # --- Age vs Time ---
    # fig2, ax2 = plt.subplots(figsize=(16, 8))
    # for lbl in labels:
    #     sub = df_vessels[df_vessels["Label"] == lbl]
    #     ax2.plot(sub["Time"], sub["Age"], marker="o", markersize=3, label=lbl)
    # ax2.set_xlabel("Time (quarters)")
    # ax2.set_ylabel("Age (quarters)")
    # ax2.set_title("Vessel Age Over Time")
    # ax2.set_xlim(year_ticks[0], year_ticks[-1])
    # ax2.legend(loc="upper left", bbox_to_anchor=(1.05,1))
    # ax2.grid(True)
    # fig2.tight_layout(rect=[0,0,0.85,1])
    # figures["vessel_age_over_time"] = fig2
def plot_costs_over_time(T, cost_data, fleet_label, start_year=2026):
    """
    Stacked CAPEX/OPEX/Fuel Cost over time + scatter at final residual.
    """

    # helper to pull out either a dict[t] or a constant
    def series(key):
        v = cost_data.get(key, 0)
        if isinstance(v, dict):
            return [v.get(t, 0) for t in T]
        else:
            return [v] * len(T)

    df = pd.DataFrame({
        "Time":        T,
        "Vessel CAPEX": series("vessel_capex"),
        "Infra CAPEX":  series("infra_capex"),
        "Battery CAPEX":series("batt_capex"),
```

```
        "Vessel OPEX":  series("vessel_opex"),
        "Infra OPEX":   series("infra_opex"),
        "Battery OPEX": series("batt_opex"),
        "Fuel Cost":    series("fuel_cost"),
    })

    df["CAPEX"] = df["Vessel CAPEX"] + df["Infra CAPEX"] + df["Battery CAPEX"]
    df["OPEX"] = df["Vessel OPEX"] + df["Infra OPEX"] + df["Battery OPEX"]
    df["Total Cost"] = df["CAPEX"] + df["OPEX"] + df["Fuel Cost"]

    # draw the six -timeseries
    fig, ax = plt.subplots(figsize=(16,8))
    for col in ["CAPEX","OPEX","Fuel Cost","Total Cost"]:
        ax.plot(df["Time"], df[col], marker="o", markersize=3, label=col)

    # scatter the residual value at T_max
    T_max = max(T)
    residual = (
        cost_data.get("vessel_residual", 0)
      + cost_data.get("infra_residual", 0)
      + cost_data.get("batt_residual", 0)
    )
    ax.scatter([T_max], [residual], s=60,
               facecolor="black", edgecolor="k", clip_on=False,
               label="Residual value")

    ax.set_title(f"{fleet_label} - Cost Components Over Time")
    years, labels = _year_ticks(T, start_year)
    ax.set_xticks(years)
    ax.set_xticklabels(labels)
    ymin = min(df["CAPEX"].min(),0)
    ax.set_ylim(bottom=ymin * 1.1)
    ax.set_xlabel("Year")
    ax.set_xlim(years[0], T[-1])
    ax.set_ylabel("Cost €( thousands)")
    ax.tick_params(axis='x', labelsize=12)   # ← override to 16pt for this plot
    ax.legend(loc="upper right")
    ax.grid(True)
    fig.tight_layout(rect=[0,0,0.85,1])

    figures["cost_components_over_time"] = fig
def plot_cumulative_tco_lca(T, cost_data, emissions_data, fleet_label, start_year=2026):
    """
    Dual-axis: cumulative total cost vs cumulative LCA ( CO only).
    """

    # helper to fetch cost_data[key][t] if 'its a dict, else return scalar
    def cval(key, t):
        v = cost_data.get(key, 0)
        if isinstance(v, dict):
            return v.get(t, 0)
        return v

    running_cost = 0.0
    cum_cost     = []
    running_co2  = 0.0
    cum_co2      = []

    for t in T:
        # accumulate cost
        running_cost += (
            cval("vessel_capex", t)
          + cval("infra_capex",  t)
          + cval("batt_capex",   t)
          + cval("vessel_opex",  t)
          + cval("infra_opex",   t)
          + cval("batt_opex",    t)
          + cval("fuel_cost",    t)
        )
        cum_cost.append(running_cost)

```

```
        # accumulate  CO from the various series
        running_co2 += (
            emissions_data.get("vessel_ctg", {}).get(t, 0)
          + emissions_data.get("wtw", {}).get("CO2", {}).get(t, 0)
          + emissions_data.get("vessel_gtc", {}).get(t, 0)
          + emissions_data.get("infra_ctg", {}).get(t, 0)
          + emissions_data.get("infra_gtc", {}).get(t, 0)
          + emissions_data.get("batt_ctg", {}).get(t, 0)
          + emissions_data.get("batt_gtc", {}).get(t, 0)
        )
        cum_co2.append(running_co2)

    # subtract the final residual from the last point
    residual = (
        cost_data.get("vessel_residual_value", 0)
      + cost_data.get("infra_residual_value", 0)
      + cost_data.get("batt_residual_value", 0)
    )
    cum_cost_net = cum_cost.copy()
    cum_cost_net[-1] -= residual

    fig, ax1 = plt.subplots(figsize=(16,8))
    ax1.plot(
        T, cum_cost,
        marker="o", markersize=3, label="Cumulative TCO",
        color="blue"               # ← force blue for cost
    )
    years, labels = _year_ticks(T, start_year)
    ax1.set_xticks(years)
    ax1.set_xticklabels(labels)
    ax1.tick_params(axis='x', labelsize=12)   # ← override to 16pt for this plot
    ax1.set_xlabel("Year")
    ax1.set_xlim(years[0], T[-1])
    ax1.set_ylabel("Cumulative TCO €(k)")
    ax1.grid(True)
    ax1.set_ylim(bottom=0)
    ax2 = ax1.twinx()
    ax2.plot(
        T, cum_co2,
        marker="s", markersize=3, label="Cumulative LCA",
        color="green"              # ← force green for LCA
    )
    ax2.set_ylabel("Cumulative LCA (t  CO-eq)")

    # combined legend
    h1, l1 = ax1.get_legend_handles_labels()
    h2, l2 = ax2.get_legend_handles_labels()
    ax1.legend(h1+h2, l1+l2, loc="upper left")
    ax2.set_ylim(bottom=0)
    ax1.set_title(f"{fleet_label} - Cumulative TCO & LCA")
    fig.tight_layout()

    figures["cumulative_tco_lca"] = fig


def plot_wtw_emissions_over_time(T, emissions_data, fleet_label, start_year=2026):
    """
    Plot each pollutant in emissions_data['wtw'] over time,
    scaling NOx to "per 10 "kg and PM to "per 100 "kg.
    """
    # define how we want to scale each series
    scale = {
        "CO2": 1,     # already in t-CO2-eq
        "NOX": 10,    # show "per 10 "kg
        "PM": 1     # show "per 100 "kg
    }
    # and what unit-label to display for each
    unit_label = {
        "CO2": "t  CO-eq",
        "NOX": "× 10 kg  NO",
        "PM": "kg PM",
```

```python
    }

    fig, ax = plt.subplots(figsize=(16,8))

    for pol, series in emissions_data["wtw"].items():
        divisor = scale.get(pol, 1)
        vals = [series[t] / divisor for t in T]
        ax.plot(T, vals, marker="o", markersize=3, label=f"{pol} ({unit_label[pol]})")

    # tick formatting unchanged
    years, labels = _year_ticks(T, start_year)
    ax.set_xticks(years)
    ax.set_xticklabels(labels)

    ax.set_xlabel("Year")
    ax.set_xlim(years[0], T[-1])
    ax.set_ylabel("Emissions")
    ax.set_title(f"{fleet_label} - WtW Fuel Emissions")
    ax.set_ylim(bottom=0)
    ax.tick_params(axis='x', labelsize=12)   # ← override to 16pt for this plot
    ax.grid(True)
    ax.legend(loc= "upper right")
    fig.tight_layout(rect=[0,0,0.85,1])

    figures["wtw_emissions_over_time"] = fig
    # store the *raw* data if you like, or store the scaled if that's what downstream wants:
    figure_data["wtw_emissions_over_time"] = {
        **{pol: [emissions_data["wtw"][pol][t] for t in T] for pol in emissions_data["wtw"]},
        "Time": T
    }

def plot_emission_breakdowns_over_time(T, emissions_data, fleet_label, start_year=2026):
    """
    Plot CTG vs GTC for vessel, infra, batt, plus totals and
    include WtW  CO emissions as its own line.
    """
    fig, ax = plt.subplots(figsize=(16,8))

    # --- CTG series ---
    ax.plot(T, [emissions_data["vessel_ctg"][t] for t in T],
            marker="o", markersize=3, label="Vessel CTG")
    ax.plot(T, [emissions_data["infra_ctg"][t] for t in T],
            marker="o", markersize=3, label="Infra CTG")
    ax.plot(T, [emissions_data["batt_ctg"][t] for t in T],
            marker="o", markersize=3, label="Batt CTG")
    total_ctg = [emissions_data["vessel_ctg"][t]
               + emissions_data["infra_ctg"][t]
               + emissions_data["batt_ctg"][t] for t in T]
    ax.plot(T, total_ctg, linestyle="--", color="red", label="Total CTG")

    # --- GTC series ---
    ax.plot(T, [emissions_data["vessel_gtc"][t] for t in T],
            marker="x", markersize=3, label="Vessel GTC")
    ax.plot(T, [emissions_data["infra_gtc"][t]  for t in T],
            marker="x", markersize=3, label="Infra GTC")
    ax.plot(T, [emissions_data["batt_gtc"][t]   for t in T],
            marker="x", markersize=3, label="Batt GTC")
    total_gtc = [emissions_data["vessel_gtc"][t]
               + emissions_data["infra_gtc"][t]
               + emissions_data["batt_gtc"][t] for t in T]
    ax.plot(T, total_gtc, linestyle="--", color="purple", label="Total GTC")

    # --- NEW: WtW  CO ---
    # emissions_data["wtw"]["CO2"] is a dict →tvalue (t  CO-eq)
    ax.plot(T, [emissions_data["wtw"]["CO2"].get(t, 0) for t in T],
            marker="s", markersize=4, linestyle="-.",
            color="green", label="WtW  CO")

    # formatting
    years, labels = _year_ticks(T, start_year)
    ax.set_xticks(years)
```

```
    ax.set_xticklabels(labels)
    ax.set_xlabel("Year")
    ax.set_xlim(years[0], T[-1])
    ax.tick_params(axis='x', labelsize=12)   # ← override to 16pt for this plot
    ax.set_ylabel("Emissions (t  CO-eq)")
    ax.set_title(f"{fleet_label} - CTG, GTC & WtW  CO Emissions")
    ax.legend(loc="upper right")
    ax.grid(True)
    fig.tight_layout(rect=[0,0,0.85,1])

    figures["emission_breakdown_over_time"] = fig
    figure_data["emission_breakdown_over_time"] = {
        "Time":     T,
        "VesselCTG":   [emissions_data["vessel_ctg"][t] for t in T],
        "InfraCTG":    [emissions_data["infra_ctg"][t]  for t in T],
        "BatteryCTG":  [emissions_data["batt_ctg"][t]   for t in T],
        "VesselGTC":   [emissions_data["vessel_gtc"][t] for t in T],
        "InfraGTC":    [emissions_data["infra_gtc"][t]  for t in T],
        "BatteryGTC":  [emissions_data["batt_gtc"][t]   for t in T],
        "WtW_CO2":     [emissions_data["wtw"]["CO2"].get(t, 0) for t in T]
    }

def plot_acquisition_and_operation(df_assets, fleet_label, start_year=2026):
    """
    df_assets: DataFrame with columns [Time, Type, Asset, Name, Value]
    Type   {"Acquisition","Operation","NonOperational Batteries","Salvaged"}
    Asset   {"Infra","Battery"}
    Name  = infra name (e.g. "BSM") or None for batteries
    """

    # ensure Time is sorted
    df_assets = df_assets.sort_values("Time")
    T = sorted(df_assets["Time"].unique())

    fig, (ax1, ax2) = plt.subplots(2,1, figsize=(16,8), sharex=True)

    # --- Top: Acquisition & Salvage ---
    df_acq    = df_assets[df_assets.Type == "Acquisition"]
    df_salv   = df_assets[df_assets.Type == "Salvaged"]

    # Infra acquisitions (one line per Name)
    for name, sub in df_acq[df_acq.Asset=="Infra"].groupby("Name"):
        ax1.plot(sub["Time"], sub["Value"], '-o', markersize=3, label=f"Acquire {name}")

    # Battery acquisitions
    batt_acq = df_acq[df_acq.Asset=="Battery"]
    ax1.plot(batt_acq["Time"], batt_acq["Value"], '-o', markersize=3, label="Acquire Batteries")

    # Salvaged batteries
    df_salv_batt = df_salv[df_salv.Asset=="Battery"]
    ax1.plot(df_salv_batt["Time"], df_salv_batt["Value"], '-o', markersize=3, label="Batteries Salvaged")

    ax1.set_ylabel("Quantity")
    ax1.set_title(f"{fleet_label} - Acquisition & Salvage of Batteries and Infrastructure")
    ax1.legend(bbox_to_anchor=(1.05,1), loc="upper left")
    ax1.grid(True)

    # --- Bottom: Operation Status (4 series) ---
    df_op      = df_assets[df_assets.Type == "Operation"]
    df_inins   = df_assets[df_assets.Type == "Installed"]
    df_nonop   = df_assets[df_assets.Type == "NonOperational Batteries"]

    # 1) Batteries in operation
    batt_op = df_op[df_op.Asset=="Battery"]
    ax2.plot(batt_op["Time"], batt_op["Value"],
             '-o', markersize=3, label="Batteries Op")

    # 2) Non-operational batteries
    df_nonop_batt = df_nonop[df_nonop.Asset=="Battery"]
```

```
    ax2.plot(df_nonop_batt["Time"], df_nonop_batt["Value"],
             '--x', markersize=3, label="Batteries Non-Op")

    # 3) Installed BSM
    bsm = df_inins[(df_inins.Asset == "Infra")
                   & (df_inins.Name  == "BSM")]
    if not bsm.empty:
        ax2.plot(bsm["Time"], bsm["Value"],
                 '-s', markersize=3, label="BSM Installed")

    # 4) Installed Shorepower
    shore = df_inins[(df_inins.Asset == "Infra")
                     & (df_inins.Name  == "Shorepower")]
    if not shore.empty:
        ax2.plot(shore["Time"], shore["Value"],
                 '-^', markersize=3, label="Shorepower Installed")

    ax2.set_ylabel("Quantity")
    ax2.set_title(f"{fleet_label} - Operation Status Batteries and Infrastructure")
    ax2.legend(bbox_to_anchor=(1.05,1), loc="upper left")
    ax2.grid(True)

    years, labels = _year_ticks(T, start_year)
    years = years[::2]
    labels = labels[::2]


    ax2.set_xlabel("Year")
    ax1.set_xlim(years[0], T[-1])
    plt.tight_layout(rect=[0,0,0.85,1])
    ax1.set_xticks(years)
    ax1.set_xticklabels(labels)
    ax2.set_xlabel("Year")
    ax1.set_xlim(years[0], T[-1])
    plt.tight_layout(rect=[0,0,0.85,1])
    figures["acquisition_and_operation_assets"] = fig
def save_figures_and_data(plots_dir):
    """
    Write out all figures
    """
    os.makedirs(plots_dir, exist_ok=True)
    for name, fig in figures.items():
        # save as PDF
        fig.savefig(
            os.path.join(plots_dir, f"{name}.pdf"),
            format='pdf',
            bbox_inches="tight"
        )
        plt.close(fig)
    # for name, data in figure_data.items():
    #     savemat(os.path.join(plots_dir, f"{name}.mat"), {name: data})


def run_all_plots_for_fleet(
    fleet_label,
    df_vessels,
    cost_data,
    emissions_data,
    df_assets,
    base_case
):
    """
    Orchestrate *all* of the above for one fleet/run.
    Returns the dict of matplotlib.Figure objects.
    """
    figures.clear()
    figure_data.clear()

    T = base_case["T"]

    plot_vessel_gantt_and_age(df_vessels, fleet_label)
```

```
    plot_costs_over_time(T, cost_data, fleet_label)
    plot_cumulative_tco_lca(T, cost_data, emissions_data, fleet_label)
    plot_wtw_emissions_over_time(T, emissions_data, fleet_label)
    plot_emission_breakdowns_over_time(T, emissions_data, fleet_label)
    plot_acquisition_and_operation(df_assets, fleet_label)

    return figures
```

F

TOPSIS Source Code

The following Python script implements the TOPSIS method for MCDA used in the decision support layer of the model. The code handles normalisation, weighting, and ranking of Pareto-optimal solutions based on stakeholder preferences.

```
import pandas as pd
import numpy as np
import os

# === USER CONFIGURATION ===
directory = r"C:\Users\penti\OneDrive\Results\Modelv5\Pathway2"  # Change this to the
    appropriate result folder
# Stakeholder-defined weights: TCO, LCA, LPnox
# === TOPSIS Ranking ===

# Stakeholder-defined weights: TCO, LCA, LPnox
weights = [0, 0, 1]

# Reuse `topsis` name with vector normalization internally

def topsis(df, criteria_cols, benefit_criteria, weights):
    """
    Vector-normalized TOPSIS:
      1) Normalize each criterion by Euclidean norm
      2) Apply stakeholder weights (sum to 1)
      3) Invert cost criteria
      4) Compute distances to ideal and nadir
      5) Score = D_minus / (D_plus + D_minus)
    """
    X = df[criteria_cols].values.astype(float)
    norms = np.linalg.norm(X, axis=0)
    X_norm = X / norms

    w = np.array(weights, dtype=float)
    w /= w.sum()
    V = X_norm * w

    for i, is_benefit in enumerate(benefit_criteria):
        if not is_benefit:
            V[:, i] = 1 - V[:, i]

    y_plus = V.max(axis=0)
    y_minus = V.min(axis=0)

    D_plus = np.linalg.norm(V - y_plus, axis=1)
    D_minus = np.linalg.norm(V - y_minus, axis=1)

    C_s = D_minus / (D_plus + D_minus)

    out = df.copy().reset_index(drop=True)
```

```
    out['TOPSIS_Score'] = C_s
    out['TOPSIS_Rank']  = out['TOPSIS_Score'].rank(ascending=False).astype(int)
    return out.sort_values('TOPSIS_Rank')
# === MAIN EXECUTION ===
summary_path = os.path.join(directory, "epsilon_combined_summary_cleaned.csv")

if not os.path.exists(summary_path):
    print(f"␣File␣not␣found:␣{summary_path}")
else:
    df = pd.read_csv(summary_path)

    # Filter for feasible solutions only
    df = (df[df["Status"] == "Feasible"].copy() if "Status" in df.columns else df)

    if df.empty:
        print("␣No␣feasible␣results␣found.")
    else:
        # Criteria definitions:
        # f1: LCA (min), f2: TCO_res (min), f3: LP (min)
        criteria = ["TCO", "LCA", "LPnox"]
        benefit_criteria = [False, False, False]  # All are to be minimized

        # Perform TOPSIS
        df_topsis_result = topsis(df, criteria, benefit_criteria, weights)

        # Save results
        output_path = os.path.join(directory, "topsis_ranked_results.csv")
        df_topsis_result.to_csv(output_path, index=False)
        print(f"␣TOPSIS␣completed.\nResults␣saved␣to:␣{output_path}")

        # Display top results
        print(df_topsis_result.head())
```

G
Framework verification

G.1. Preprocessing layer

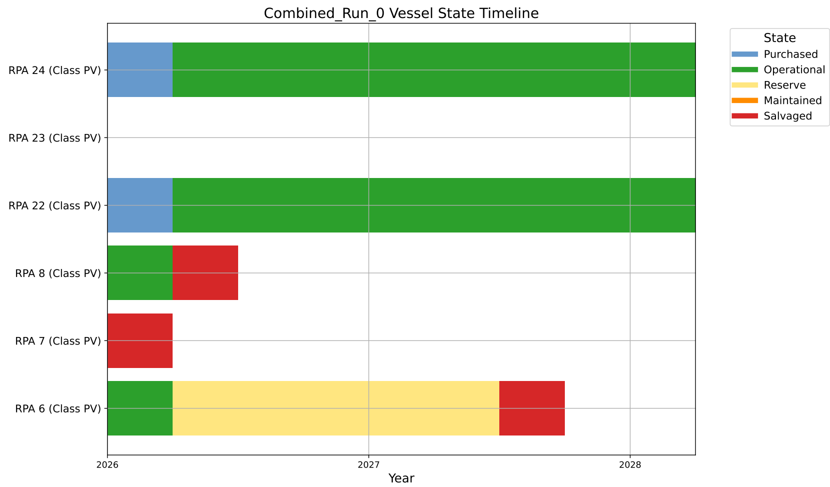


Figure G.1: Fleet schedule preprocessing verification.

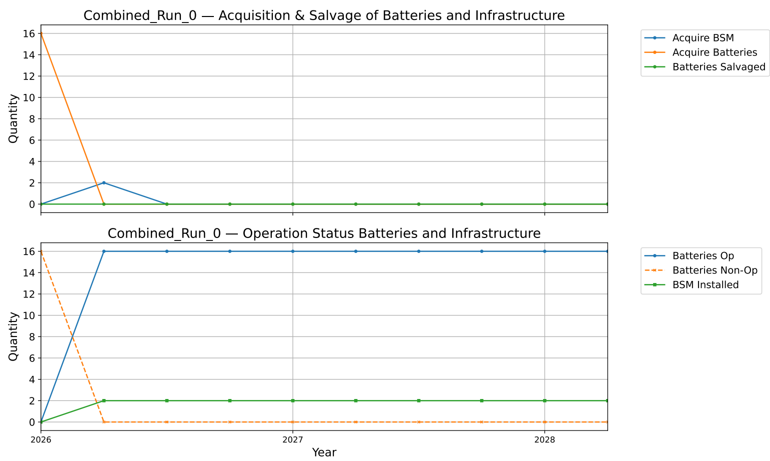


Figure G.2: Battery and infrastructure schedule preprocessing verification.

Cost:														
Model output:		vessel_capex		vessel_opex	infra_capex		infra_opex		batt_capex	batt_opex		fuel_cost		
	0	**41525**		**49**		0	0		**10720**	0			**45.974**	
	1	**-3473.75**		**151.755**		**16301.1**	**102.51**		0	**64.32**			**68.9832**	
Hand calculations:	T0		T1		Initial		T1		T0	T1	Fuel consumption	T0		T1
RPA 7 residual value		-875		Infra acquisiton		8110	8150.55	Battery acquisition	670		RPA 6		**22.2**	
RPA 8 residual value		-3500	-3473.8	Units			2	Units	16		RPA 8		**14**	
RPA 22/24 acquistion		21200		**Total infra CAPEX**			***16301.1***	**Total battery CAPEX**	***10720***		HVO cost		**1.27**	
Total vessel capex		***41525***	***-3473.8***								RPA 22/24		143	**143**
				Infra OPEX		51	51.255	Battery OPEX	4	4.02	Electricity cost		0.24	0.2412
RPA 6 OPEX		**17**	17.085	Units			2			16				
RPA 8 OPEX		**32**		**Total infra OPEX**			***102.51***	**Total batteyr OPEX**		***64.32***	**Fuel cost**		***45.974***	***68.9832***
RPA 22/24 OPEX		67	67.335											
Total vessel opex		***49***	***151.755***											

Figure G.3: Preprocessing layer cost verification.

Model:

Emissions:

Time	Category	Value	Notes
0	vessel_ctg	**80.1407**	RPA 22+24
1	infra_ctg	**2.56218**	2x BSM
0	batt_ctg	**1124.27**	16x
0	vessel_gtc	**4.66511**	RPA 7

	Time	0	1
CO2	wtw	**14.842**	0
NOX	wtw	**1371.65**	0
PM	wtw	**69.7173**	0
Notes		RPA 6+8	

Hand calculations:

Vessel CTG:		
Hull weight (22+24)	190	ton CO2/MWh
Hull material	secondary aluminium	
EU grid	0.237	ton CO2/MWh
Distance proc-prod	6300	km
EU grid	0.237	ton CO2/MWh
Distance prod-use	0	km
Emission factors:		
Transport	7.9	g/tonkm
Direct processing factor	0.23	tCO2/ton
Indirect processing fact	0.03	MWh/ton
Direct production factor	0.000771	tCO2/ton
Indirect production fact	0.566	MWh/ton
Direct processing emiss	43.7	ton CO2
Indirect processing emi	1.3509	ton CO2
Transport proc-prod	9.4563	ton CO2
Direct production emiss	0.14649	ton CO2
Indirect production emi	25.48698	ton CO2
Total CTG	**80.14067**	ton CO2

Vessel GTC		
Hull weight (7)	38	ton
EU grid	0.237	ton CO2/MWh
Distance prod-use	0	

Vessel WTW		
fuel consumption RPA	22.2	ton HVO
fuel consumption RPA	14	ton HVO
HVO emission factor	0.41	ton CO2/ ton HVO
WTW CO2 emissions	**14.842**	
RPA 6 NOX specific em	10.05	
RPA 6 PM specific emis	0.54	
RPA 8 NOX specific em	4.57	
RPA 8 PM specific emis	0.186	
LHV	11.94444444	MWh/tonHVO
engine efficiency	0.4	
RPA 6 NOX emissions	1065.97	
RPA 6 PM emissions	57.276	
RPA 8 NOX emissions	305.6822222	
RPA PM emissions	12.44133333	
Total NOX	**1371.652222**	
Total PM	69.71733333	

Infra CTG		
Total weight (2 units)	8	ton
Material	EAF SCRAP	steel
EU grid	0.237	ton CO2/MWh
Distance proc-prod	1000	km
Distance prod-use	0	km
Direct processing factor	0.04	tCO2/ton
Indirect processing factor	0.58	MWh/ton
Direct production factor	0.000771	tCO2/ton
Indirect production factor	0.566	MWh/ton
Direct processing emissions	0.32	
Indirect processing emissions	1.09968	
Transport proc-prod	0.0632	
Direct production emissions	0.006168	
Indirect production emissions	1.073136	
Total GTC	**2.562184**	

Battery CTG		
Total capacity (16 units)	16	MWh
Total weight	94.4	Ton
Production direct factor	56	ton/MWh capacity
Production indirect factor	60	MWh/MWh capacity
Distance proc-prod	1000	km
EU grid	0.237	ton CO2/MWh

Figure G.4: Preprocessing layer emission verification.

G.2. Multi-objective optimisation layer

Baseline

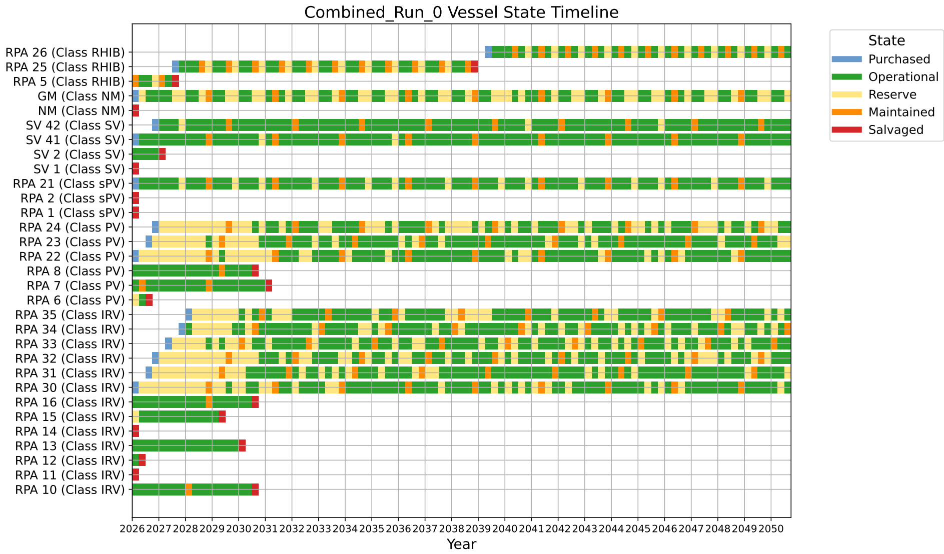


Figure G.5: Baseline verification: Fleet schedule.

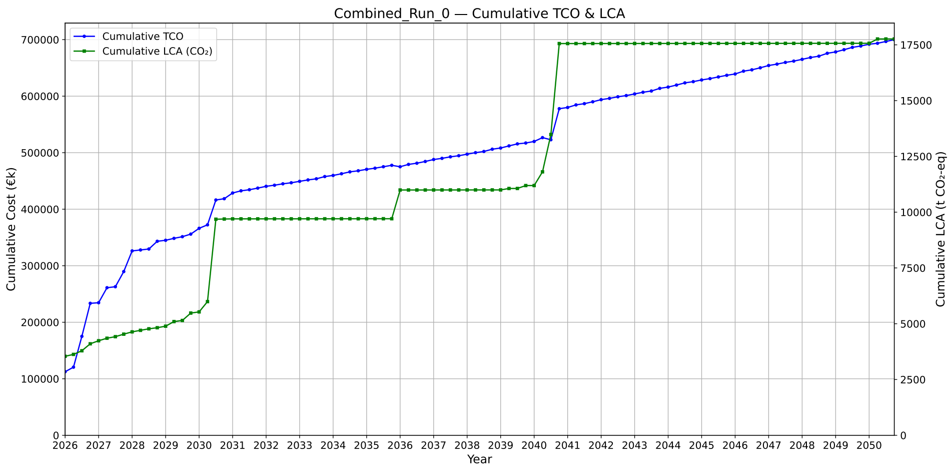


Figure G.6: Baseline verification: Cumulative TCO and LCA.

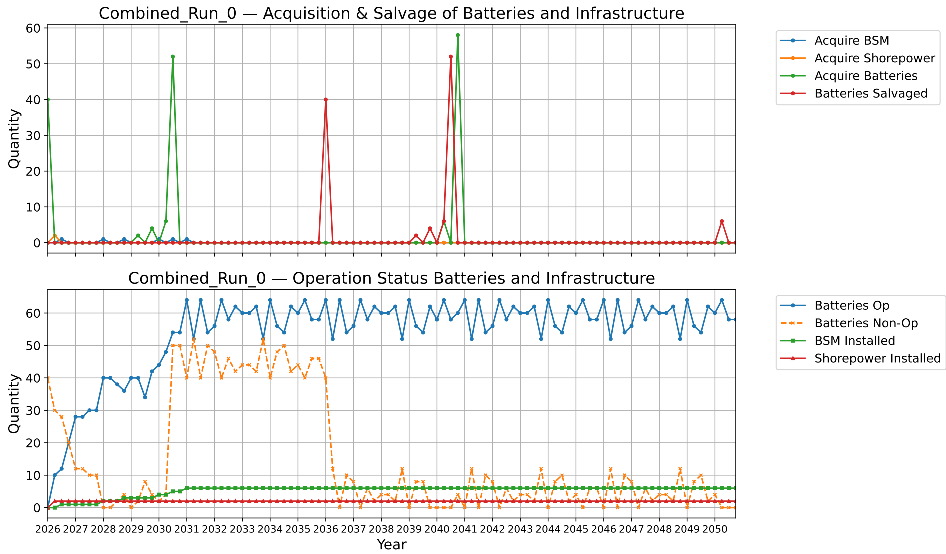


Figure G.7: Baseline verification: Asset acquisition, operation and salvage.

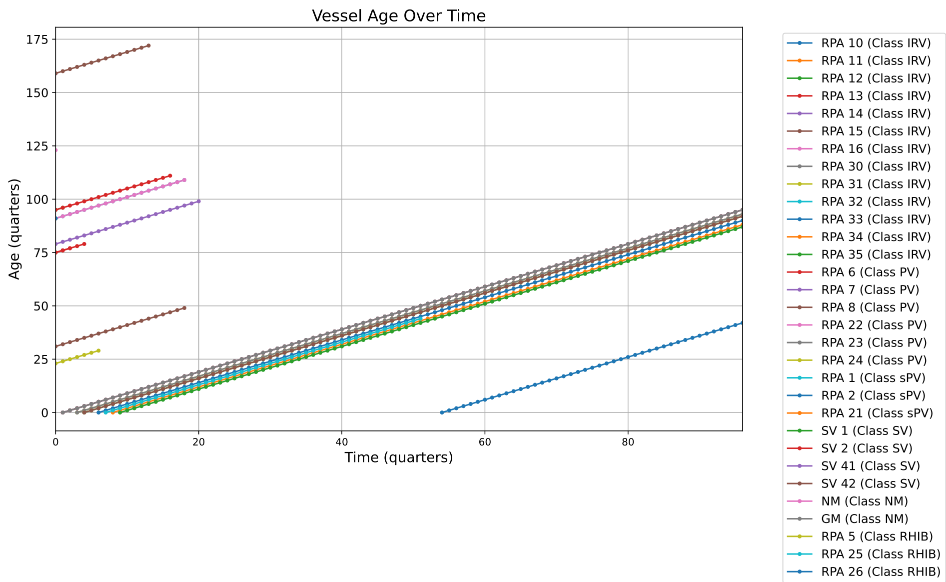


Figure G.8: Baseline verification: Vessel age over time.

Demand

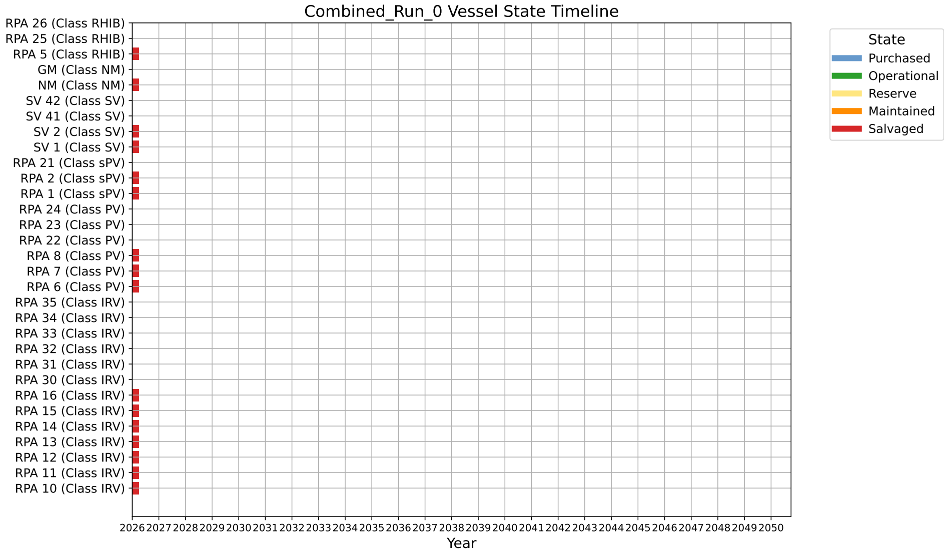


Figure G.9: Demand verification: Fleet schedule.

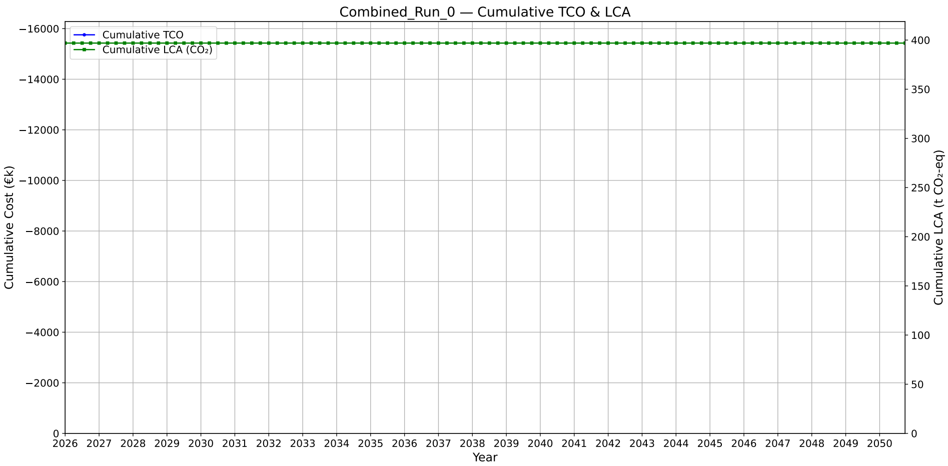


Figure G.10: Demand verification: Cumulative TCO and LCA.

Battery and infrastructure

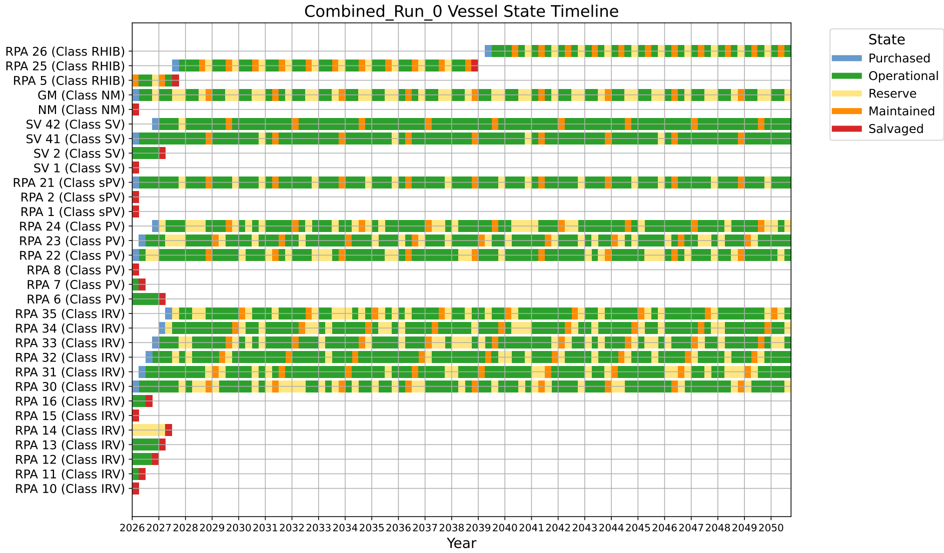


Figure G.11: Battery and infrastructure verification: Fleet schedule.

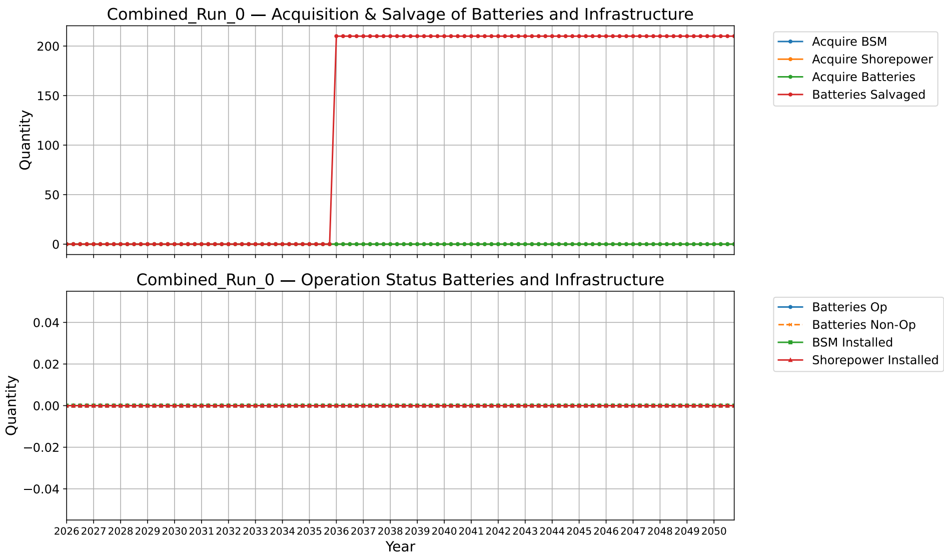


Figure G.12: Battery and infrastructure verification: Asset acquisition, operation and salvage.

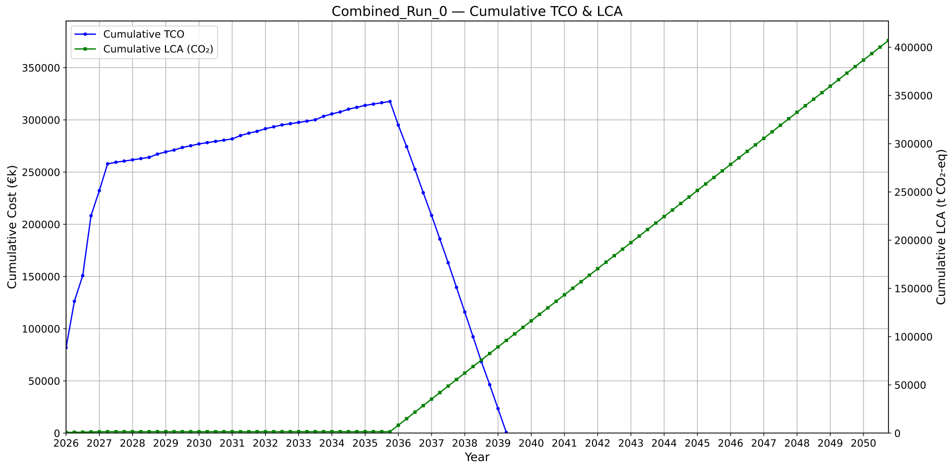


Figure G.13: Battery and infrastructure verification: Cumulative TCO and LCA.

Salvage and maintenance

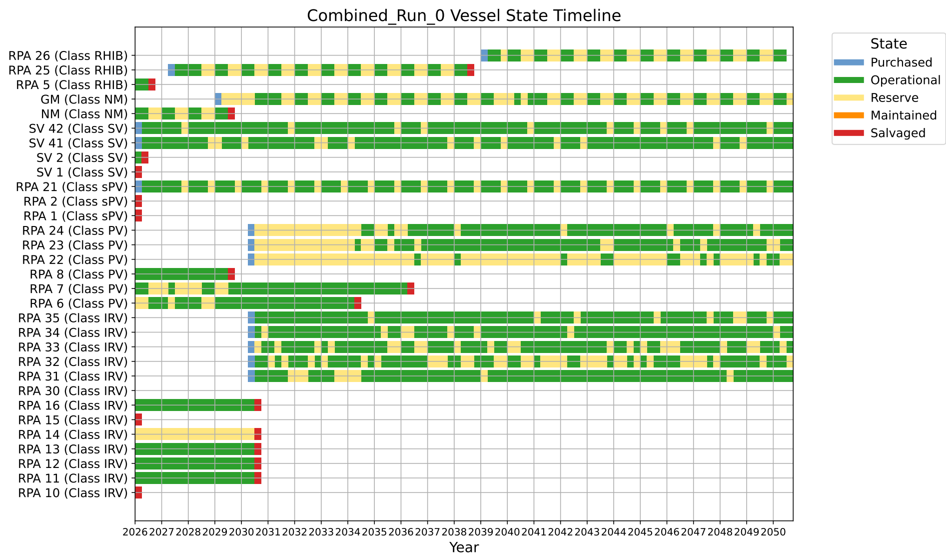


Figure G.14: Fleet composition verification: Fleet schedule.

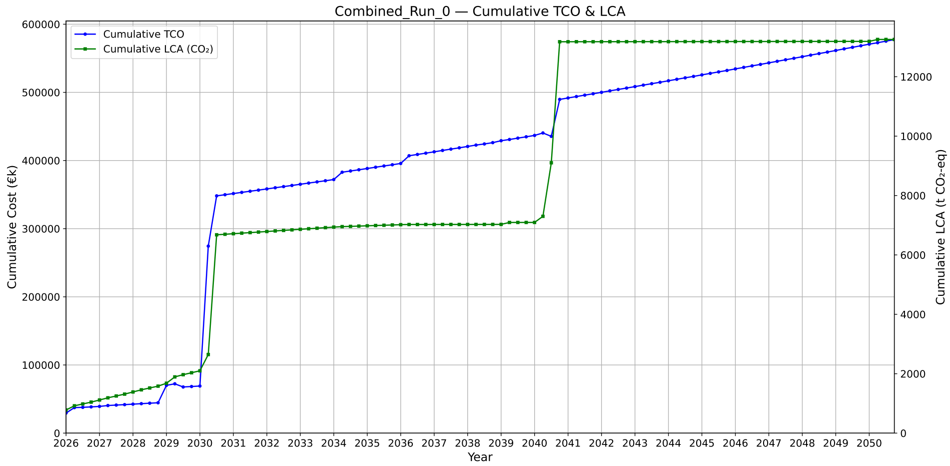


Figure G.15: Fleet composition verification: Cumulative TCO and LCA.

Ageing

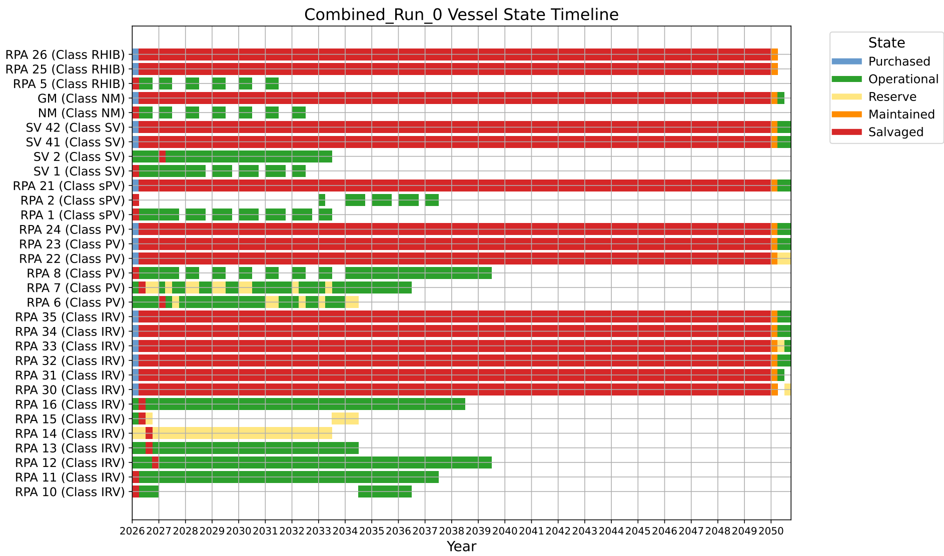


Figure G.16: Ageing verification: Fleet schedule.

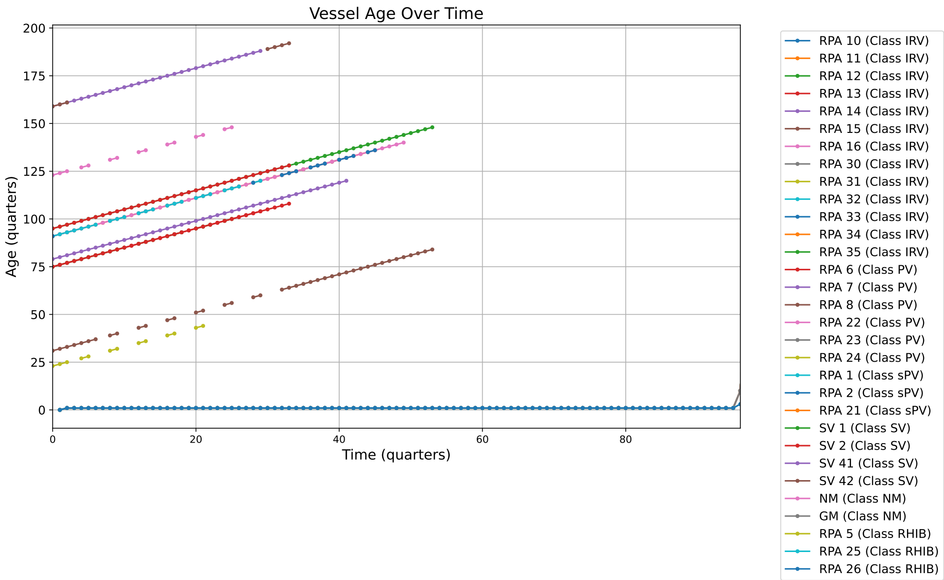


Figure G.17: Ageing verification: Vessel age over time.

G.3. Multi-criteria decision analysis layer

TOPSIS verification

Run		TCO	TCO_res	LCA	LPnox	LPpm
	0	67776.52639	30256.66639	1233.59615	1371.652	69.71733
	1	67588.91334	30069.05334	1282.46815	7461.79	400.932

TOPSIS criteria

Run		TCO	LCA	LPnox
	0	67776.52639	1233.59615	1371.652222
	1	67588.91334	1282.46815	7461.79

Norm TCO	95717.9123
Norm LCA	1779.461721
Norm LP	7586.813549

Run		Normalised TCO	Normalised LCA	Normalised LP
	0	0.708086133	0.693241184	0.180794244
	1	0.706126071	0.720705669	0.983520941

Equal weights

Run		Normalised TCO	Normalised LCA	Normalised LP
	0	0.236028711	0.231080395	0.060264748
	1	0.235375357	0.240235223	0.327840314

Invert criteria

Run		Normalised TCO	Normalised LCA	Normalised LP
	0	0.763971289	0.768919605	0.939735252
	1	0.764624643	0.759764777	0.672159686

Determine ideal ar	Ideal	Nadir
Tco	0.764624643	0.763971289
LCA	0.768919605	0.759764777
LP	0.939735252	0.672159686

Calculate Euclidian distances

Run 0 D+	0.000653354
Run 0 D-	0.267732131
Run 1 D+	0.267732131
Run 1 D-	0.000653354

TOPSIS score		Rank
Run 0	0.997565613	1
Run 1	0.002434387	2

Figure G.18: TOPSIS verification hand calculations result.

df_topsis_result - DataFrame

Index	Run	TCO	TCO_res	LCA	LPnox	LPpm	Status	TOPSIS_Score	TOPSIS_Rank
0	0	67776.5	30256.7	1233.6	1371.65	69.7173	Feasible	0.997566	1
1	1	67588.9	30069.1	1282.47	7461.79	400.932	Feasible	0.00243439	2

Figure G.19: TOPSIS verification framework result.

H

Constraint boundaries

Table H.1: ε-constraint values.

Scenario	Class	Min LCA	Max LCA	Min NOX	Max NOX
Pathway 1	IRV	50763	55055	9551	107498
	PV	22747	26234	1685	24241
	sPV	4940	5008	2	5270
	SV	5643	5643	1307	1307
	NM	4534	4842	2	2
	RHIB	356	357	1871	1871
Pathway 2	IRV	29537	38183	9546	206354
	PV	13714	18188	1682	51522
	sPV	2458	3325	2	31606
	SV	3554	3979	1306	12066
	NM	3370	4044	2	8673
	RHIB	65	65	1871	1871
Pathway 3	IRV	6507	6995	9537	104813
	PV	3148	3453	1678	23013
	sPV	121	121	0	0
	SV	256	256	1305	1305
	NM	637	688	0	5203
	RHIB	34	34	1871	1871
Pathway 4	IRV	10330	12787	9537	107486
	PV	5004	6302	1678	23013
	sPV	290	290	0	0
	SV	464	464	1305	1305
	NM	1012	1264	0	6938
	RHIB	35	35	1871	1871
Pathway 5	IRV	10520	12978	9537	107486
	PV	5090	6389	1678	23624
	sPV	301	322	0	3161
	SV	485	485	1305	1305
	NM	1012	1264	0	6938
	RHIB	35	35	1871	1871
Conservative	IRV	10342	12800	9537	104925
	PV	5012	6311	1678	23319
	sPV	290	290	0	0
	SV	464	464	1305	1305
	NM	1016	1268	0	6938
	RHIB	35	35	1871	1871
Optimistic	IRV	10342	12882	9537	65239

Continued on next page

Table H.1: ε-constraint values.

Scenario	Class	Min LCA	Max LCA	Min NOX	Max NOX
	PV	5012	6311	1678	22707
	sPV	290	290	0	0
	SV	464	464	1305	1305
	NM	1016	1268	0	6938
	RHIB	35	35	1871	1871
CO2	IRV	5711	8078	9537	107486
	PV	2645	3951	1678	29432
	sPV	248	290	0	0
	SV	429	456	1305	1305
	NM	549	810	0	5203
	RHIB	34	34	1871	1912

I

Pareto fronts

Pathway 1

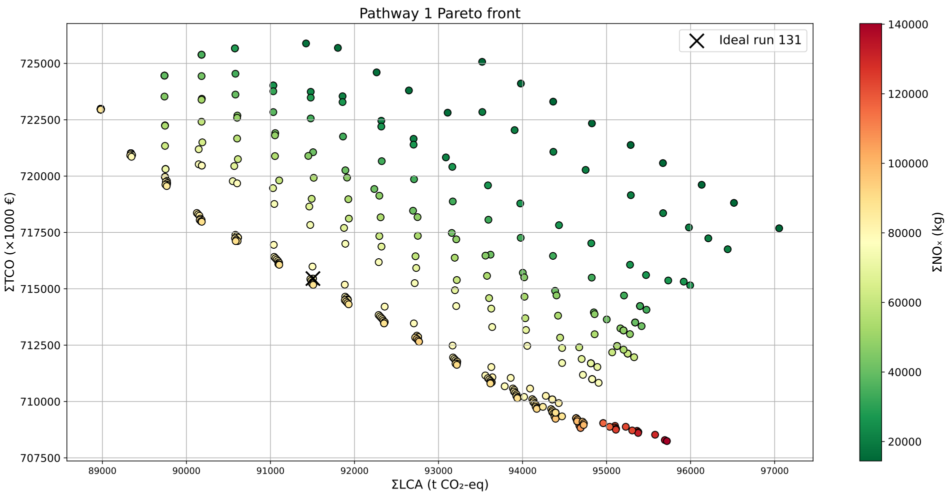


Figure I.1: Pathway 1: Pareto front LCA - TCO.

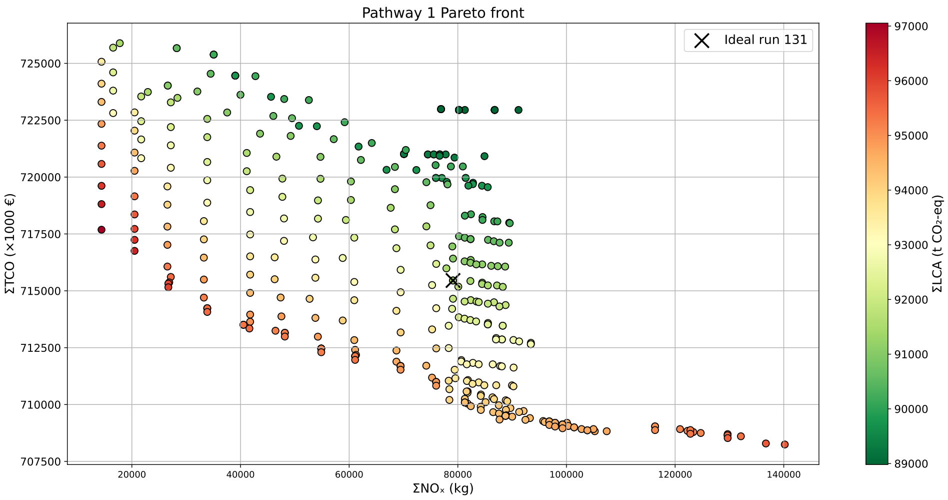


Figure I.2: Pathway 1: Pareto front LP - TCO.

Pathway 2

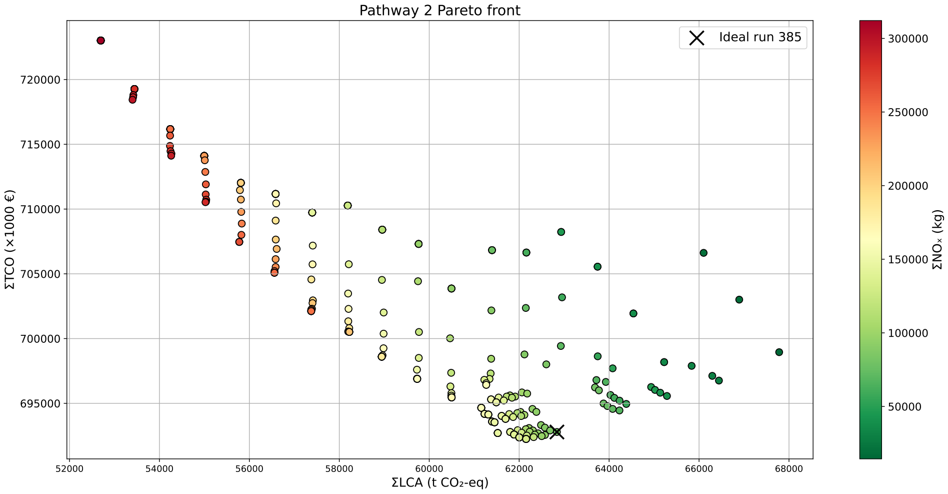


Figure I.3: Pathway 2: Pareto front LCA - TCO.

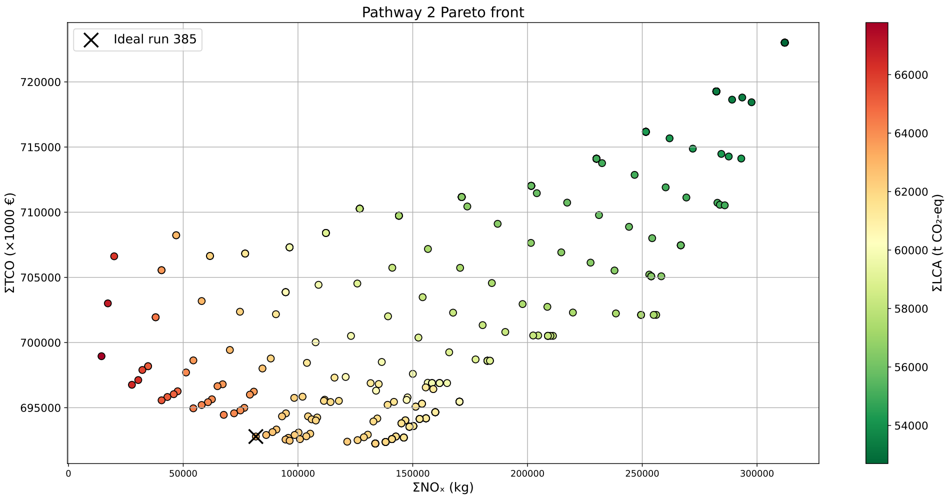


Figure I.4: Pathway 2: Pareto front LP - TCO.

Pathway 3

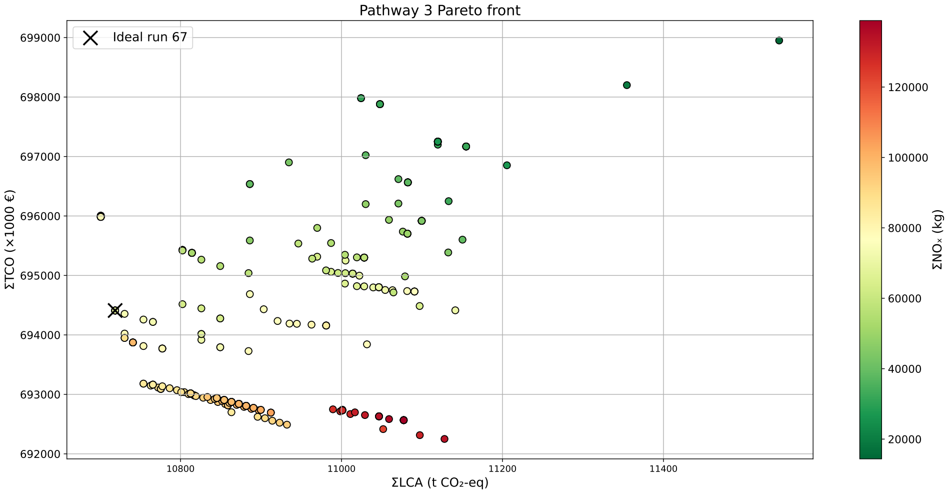


Figure I.5: Pathway 3: Pareto front LCA - TCO.

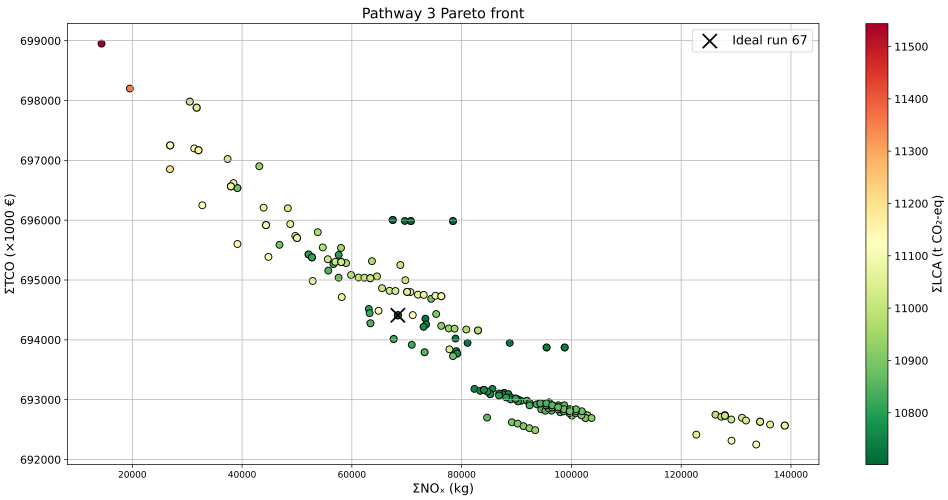


Figure I.6: Pathway 3: Pareto front LP - TCO.

Pathway 4

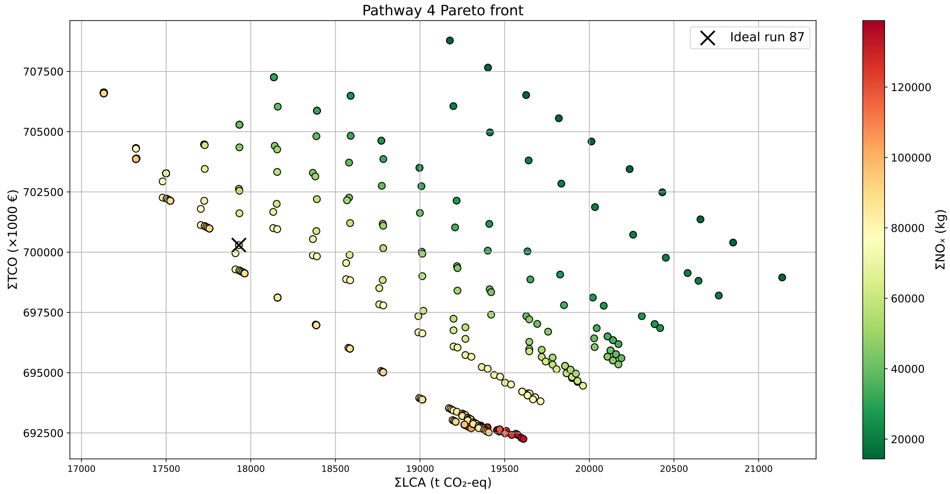


Figure I.7: Pathway 4: Pareto front LCA - TCO.

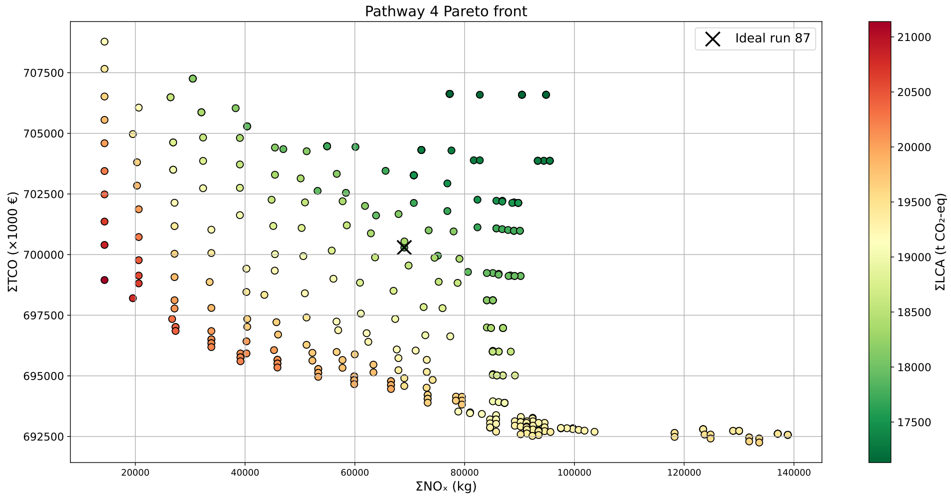


Figure I.8: Pathway 4: Pareto front LP - TCO.

Pathway 5

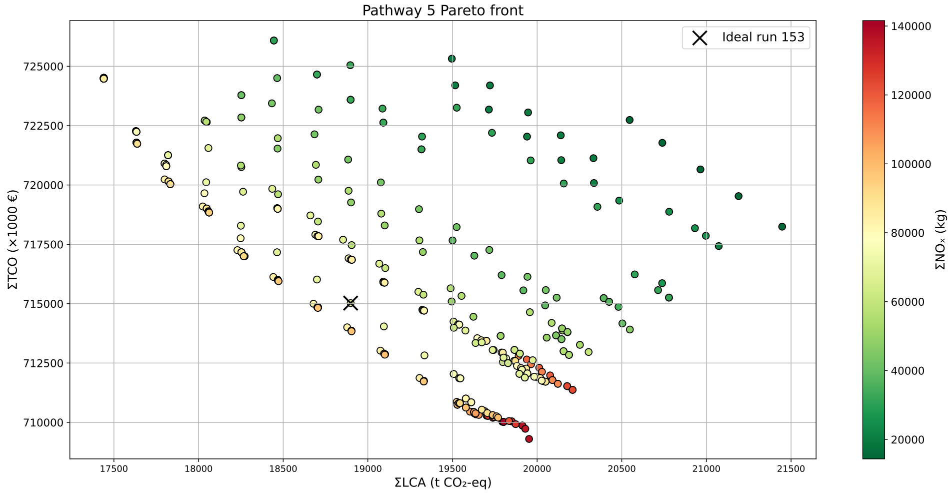


Figure I.9: Pathway 5: Pareto front LCA - TCO.

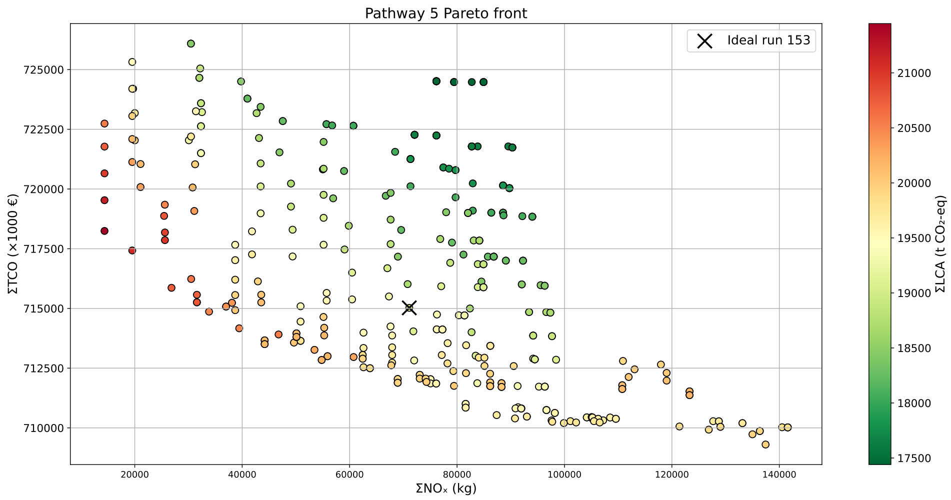


Figure I.10: Pathway 5: Pareto front LP - TCO.

Conservative scenario

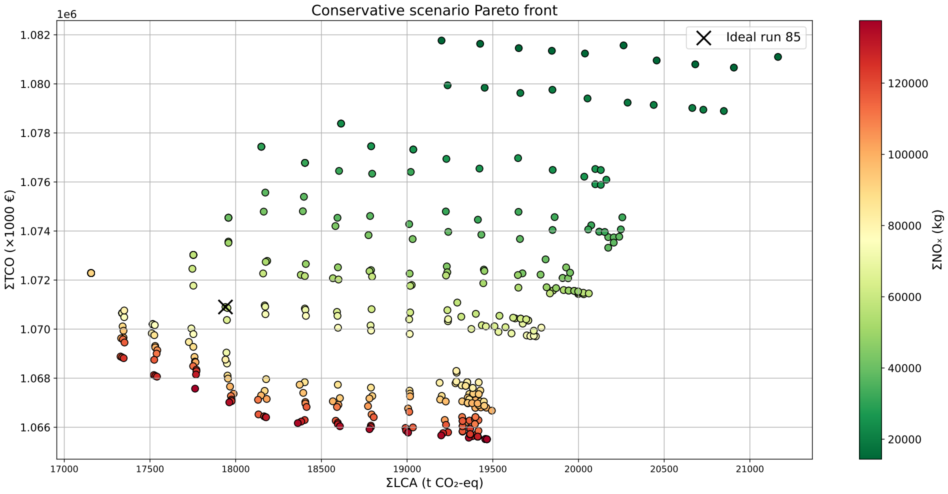


Figure I.11: Conservative scenario: Pareto front LCA - TCO.

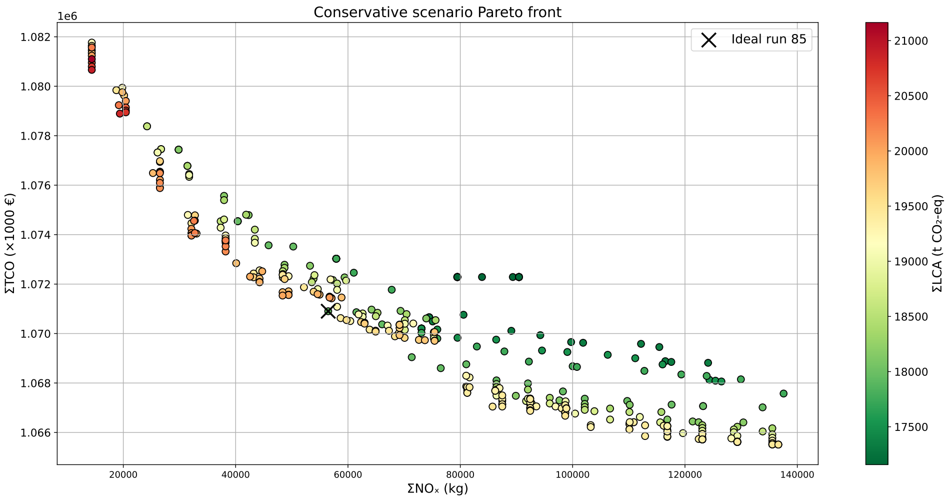


Figure I.12: Conservative scenario: Pareto front LP - TCO.

Optimistic scenario

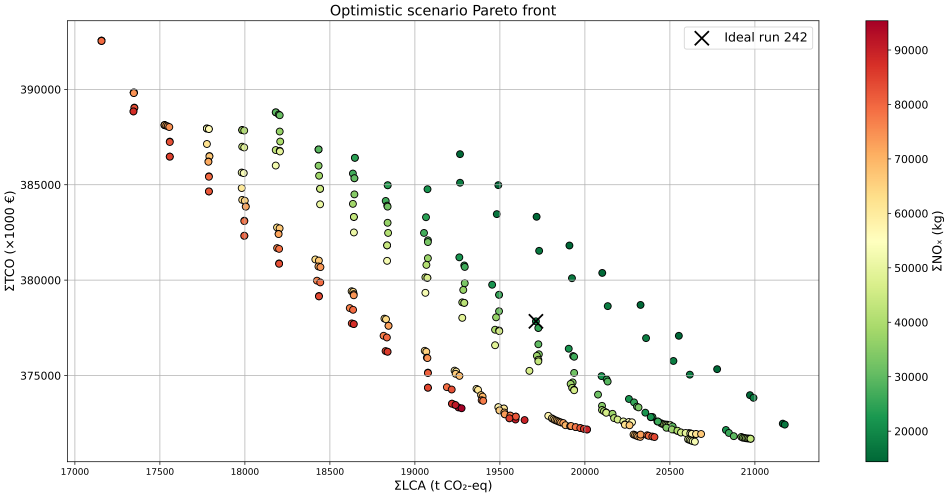


Figure I.13: Optimistic scenario: Pareto front LCA - TCO.

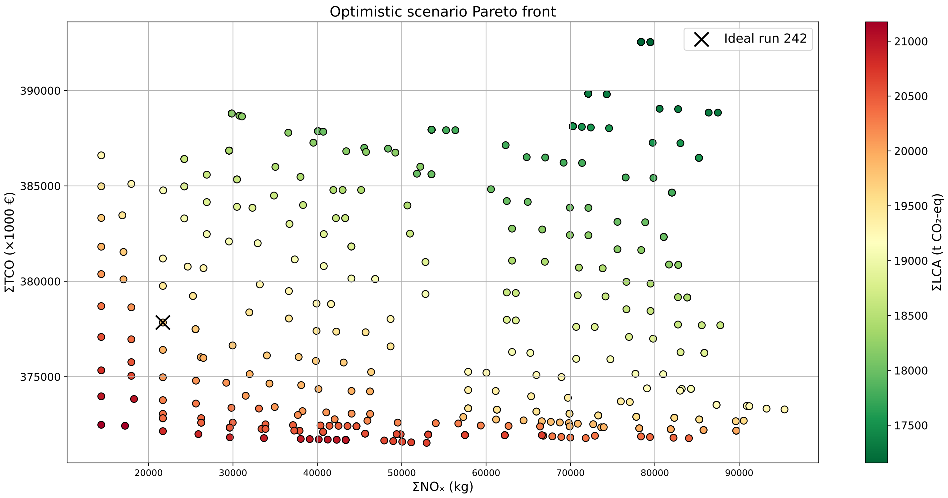


Figure I.14: Optimistic scenario: Pareto front LP - TCO.

CO2 depreciation scenario

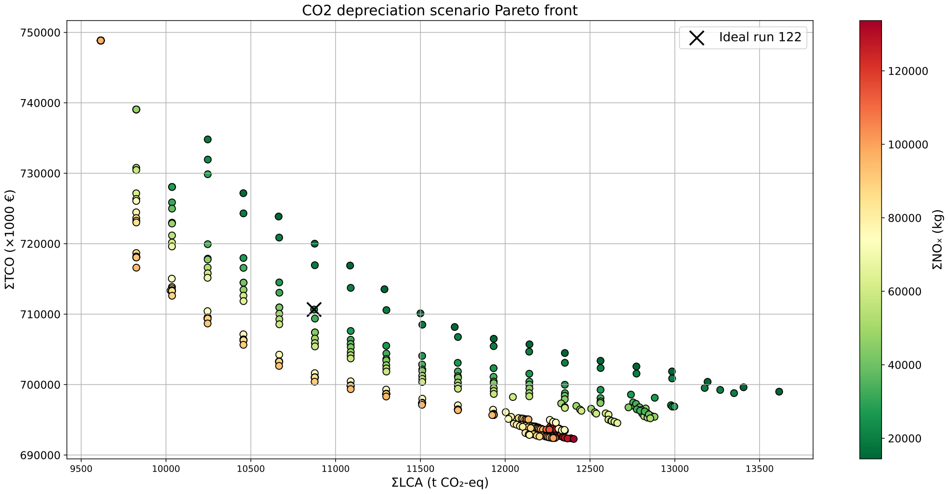


Figure I.15: CO2 depreciation scenario: Pareto front LCA - TCO.

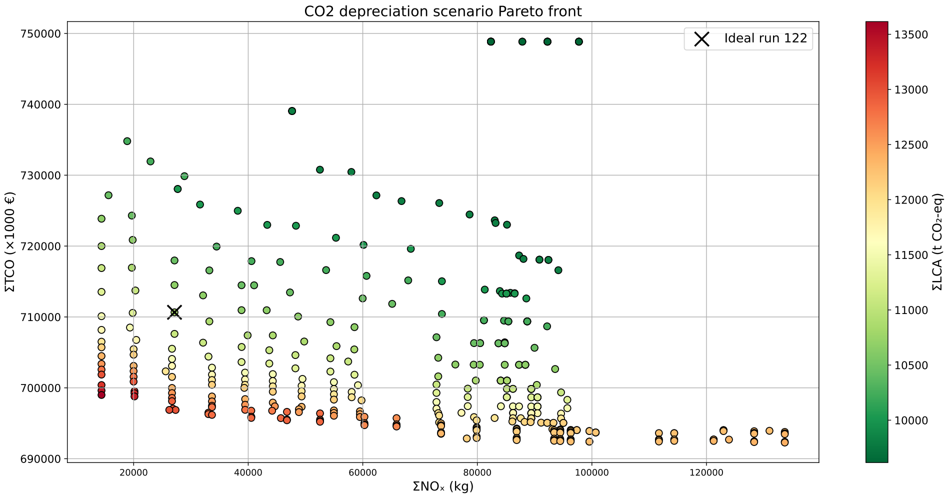


Figure I.16: CO2 depreciation scenario: Pareto front LP - TCO.

J

TOPSIS results

All results are generated using the same objective weights (TCO = 1, LCA = 1, LP_{nox} = 1).

Pathway 1

Run	TCO	LCA	LP_{nox}	TOPSIS Score	TOPSIS Rank
131	715453.70	91505.50	79117.42	0.585649	1
132	715453.70	91505.50	79117.42	0.585649	1
133	715453.70	91505.50	79117.42	0.585649	3
151	714651.00	91889.67	79117.42	0.585521	4
111	716416.95	91044.49	79117.42	0.584517	5

Table J.1: TOPSIS results for Pathway 1.

Pathway 2

Run	TCO	LCA	LP_{nox}	TOPSIS Score	TOPSIS Rank
385	692787.31	62839.57	81648.04	0.645684	1
366	692685.09	62421.69	95784.53	0.645470	2
365	692905.83	62687.95	86141.40	0.645451	3
345	693126.31	62572.21	88900.46	0.645064	4
346	692906.15	62305.96	98543.59	0.644949	5

Table J.2: TOPSIS results for Pathway 2.

Pathway 3

Run	TCO	LCA	LP_{nox}	TOPSIS Score	TOPSIS Rank
67	694410.44	10718.77	68383.19	0.709095	1
24	694410.44	10718.77	68383.19	0.709095	1
60	694410.44	10718.77	68383.19	0.709095	1
61	694410.44	10718.77	68383.19	0.709095	1
27	694410.44	10718.77	68383.19	0.709095	1

Table J.3: TOPSIS results for Pathway 3.

Pathway 4

Run	TCO	LCA	LP_{nox}	TOPSIS Score	TOPSIS Rank
87	700290.40	17930.14	69007.66	0.617352	1
211	693527.43	19170.64	78845.11	0.611954	2
207	697237.58	19197.90	56661.64	0.610353	3
227	696881.95	19268.00	56967.33	0.609565	4
88	699955.02	17909.69	75118.39	0.609113	5

Table J.4: TOPSIS results for Pathway 4.

Pathway 5

Run	TCO	LCA	LP_{nox}	TOPSIS Score	TOPSIS Rank
153	715024.35	18898.36	71106.56	0.615395	1
173	714041.64	19094.79	71866.85	0.615015	2
193	712822.50	19334.52	72020.85	0.614501	3
212	715091.16	19494.92	50867.77	0.614052	4
232	714449.00	19623.64	50867.77	0.613897	5

Table J.5: TOPSIS results for Pathway 5.

Conservative scenario

Run	TCO	LCA	LP_{nox}	TOPSIS Score	TOPSIS Rank
85	1070902.72	17940.40	56482.82	0.705895	1
88	1069043.65	17946.22	71353.84	0.694233	2
45	1070202.13	17516.20	73088.17	0.692074	5
43	1070202.13	17516.20	73088.17	0.692074	5
42	1070202.13	17516.20	73088.17	0.692074	5

Table J.6: TOPSIS results for the conservative scenario.

Optimistic scenario

Run	TCO	LCA	LP_{nox}	TOPSIS Score	TOPSIS Rank
242	377840.00	19711.85	21693.04	0.629473	1
262	376404.32	19904.93	21693.04	0.628672	2
222	379757.73	19454.41	21693.04	0.627165	3
282	374968.66	20098.00	21693.04	0.626074	4
244	377491.20	19726.16	25562.97	0.622880	5

Table J.7: TOPSIS results for the optimistic scenario.

CO2 depreciation scenario

Run	TCO	LCA	LP_{nox}	TOPSIS Score	TOPSIS Rank
122	710653.59	10873.20	27130.69	0.738655	1
142	707611.72	11088.99	27130.69	0.736777	2
123	709369.21	10878.10	33215.23	0.735261	3
143	706369.47	11088.89	32117.97	0.734812	4
124	707408.00	10878.12	39901.38	0.732291	5

Table J.8: TOPSIS results for the CO_2 depreciation scenario.

K

Pathway and scenario figures

Pathway 1

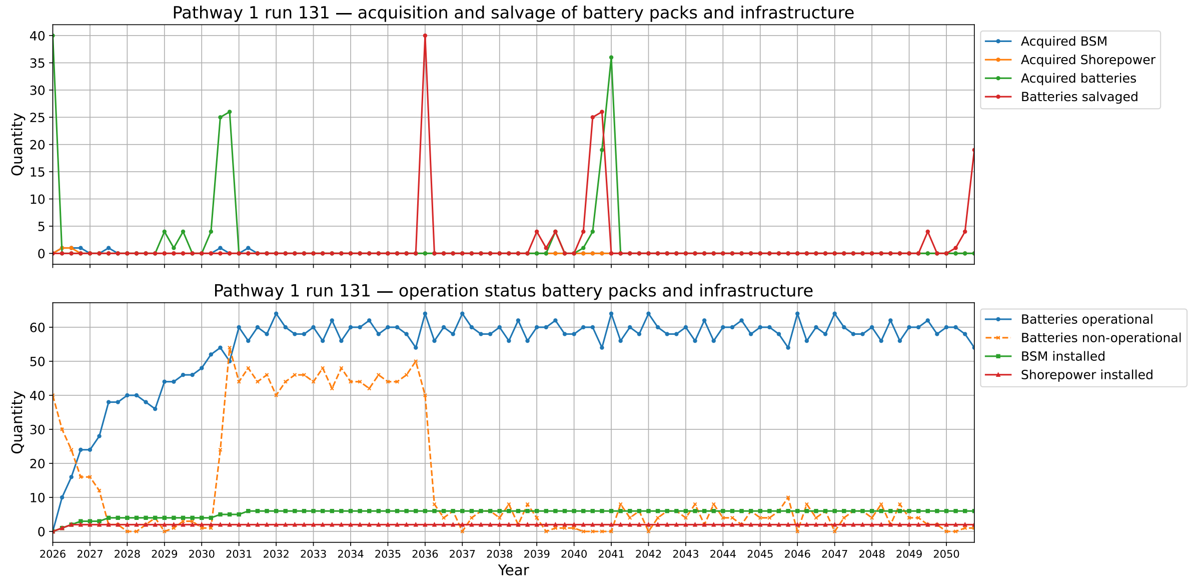


Figure K.1: Pathway 1 run 131: Battery and infrastructure schedule.

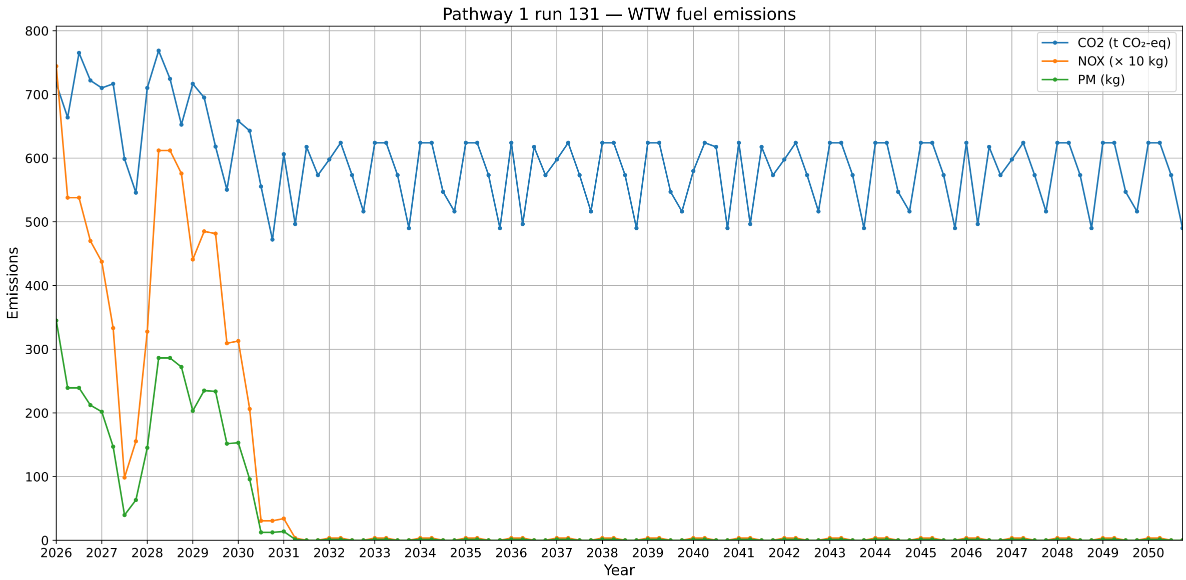


Figure K.2: Pathway 1 run 131: Well to Wake emissions over time.

also reduction 25%

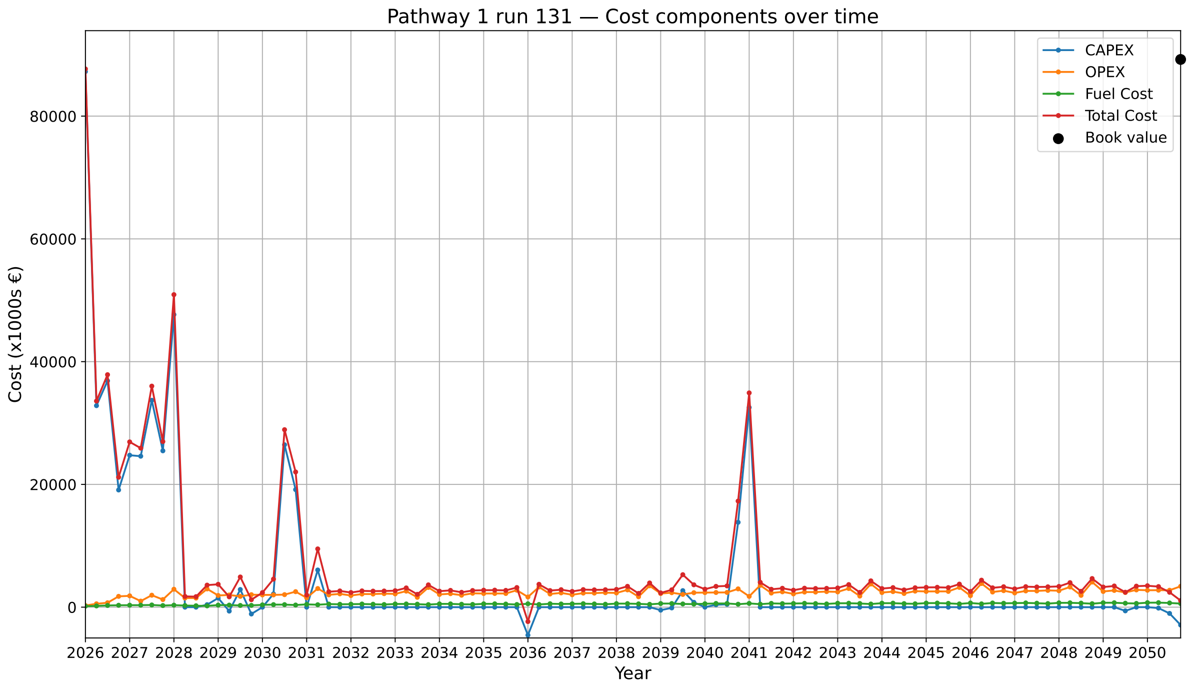


Figure K.3: Pathway 1 run 131: Cost breakdown over time.

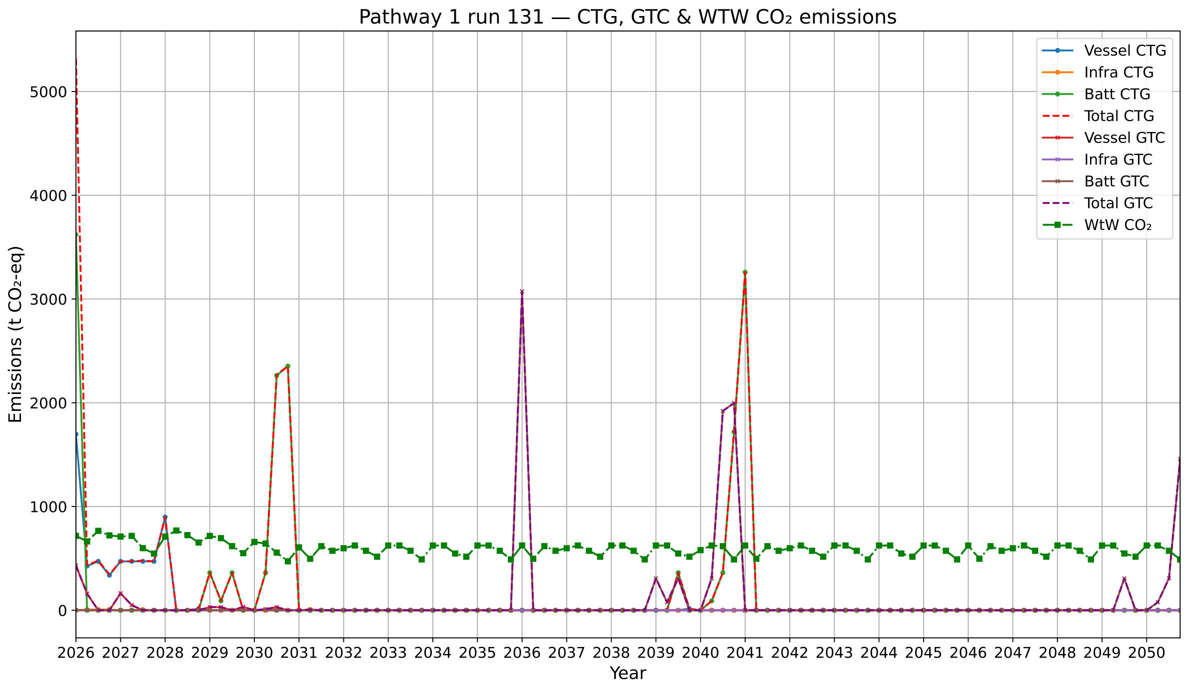


Figure K.4: Pathway 1 run 131: Emission breakdown over time.

Pathway 2

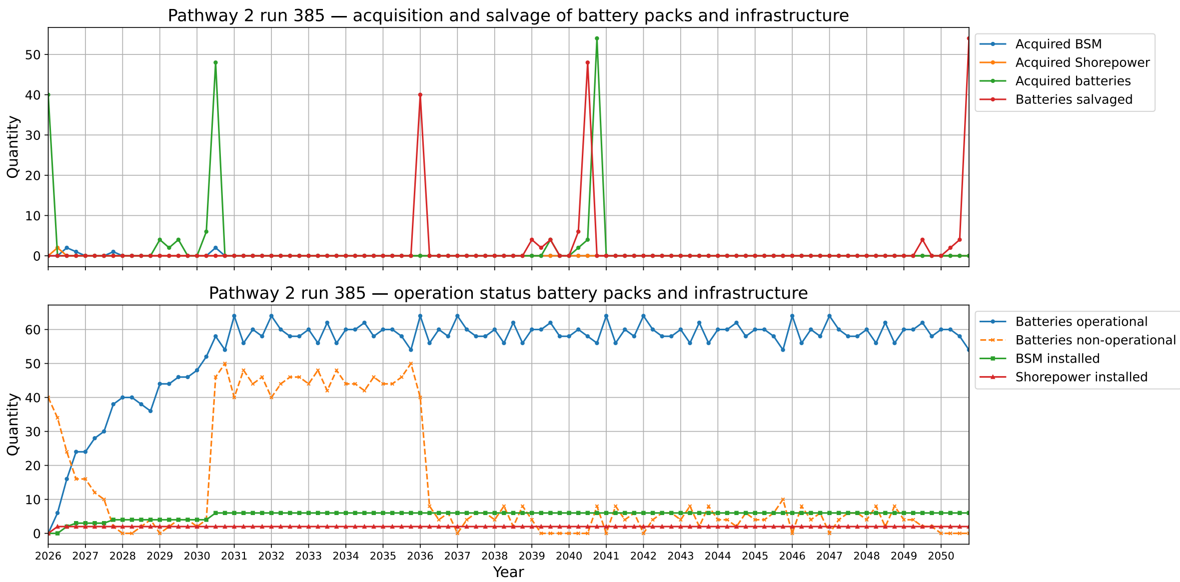


Figure K.5: Pathway 2 run 385: Battery and infrastructure schedule.

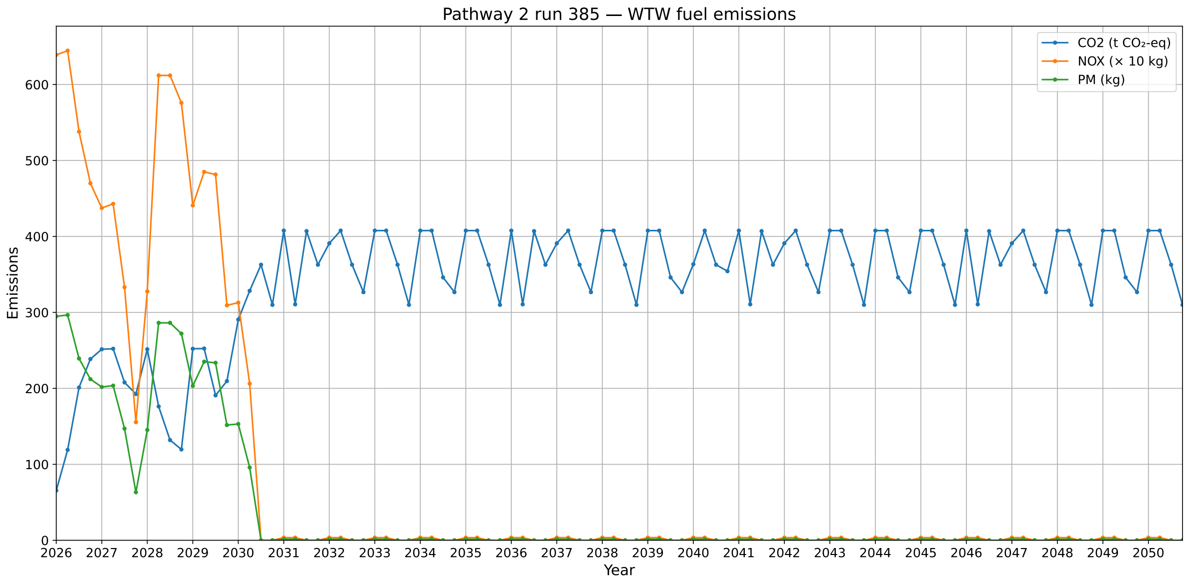


Figure K.6: Pathway 2 run 385: Well to Wake emissions over time.

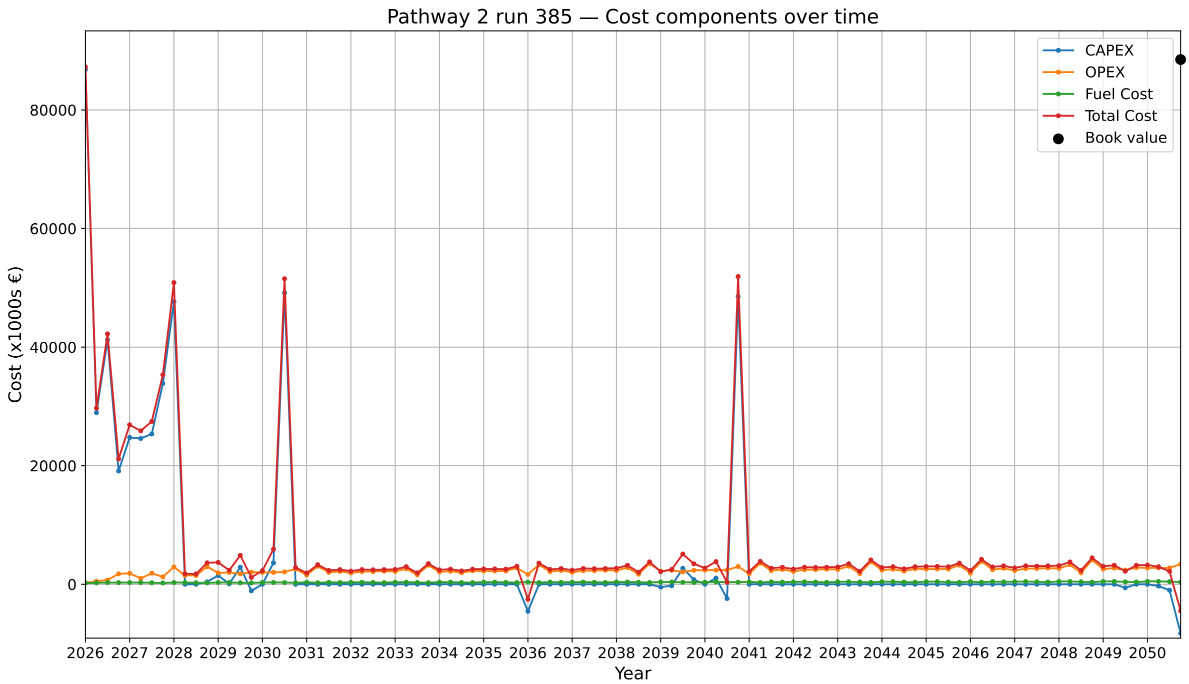


Figure K.7: Pathway 2 run 385: Cost breakdown over time.

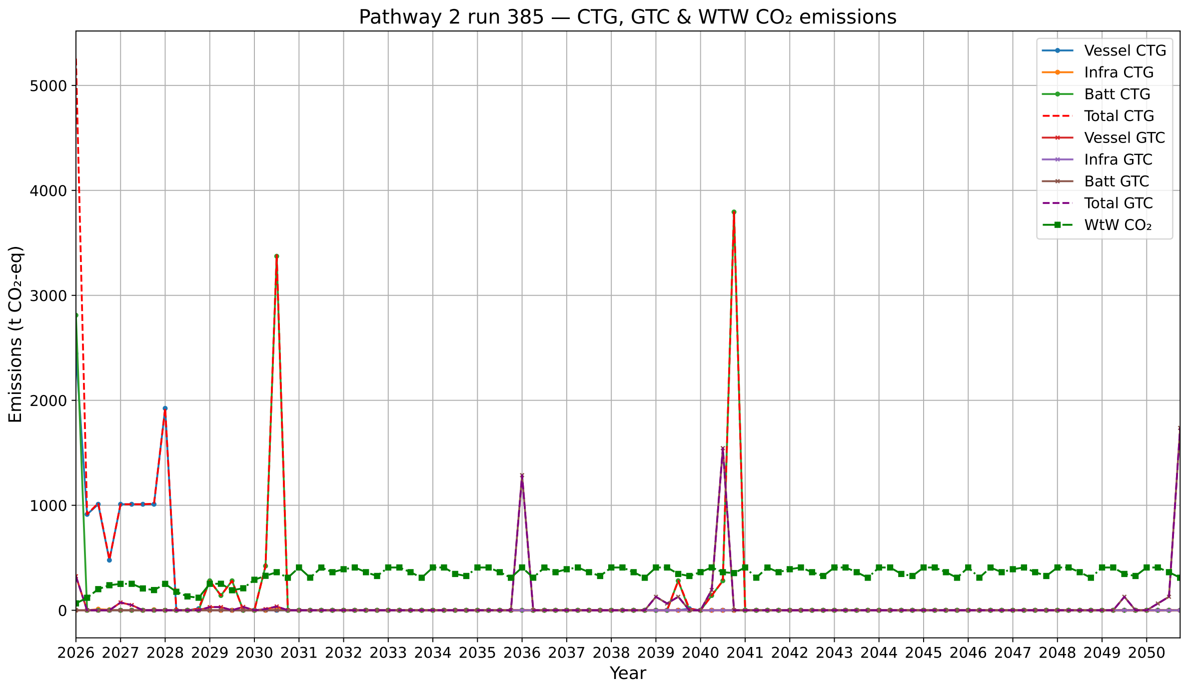


Figure K.8: Pathway 2 run 385: Emission breakdown over time.

Pathway 3

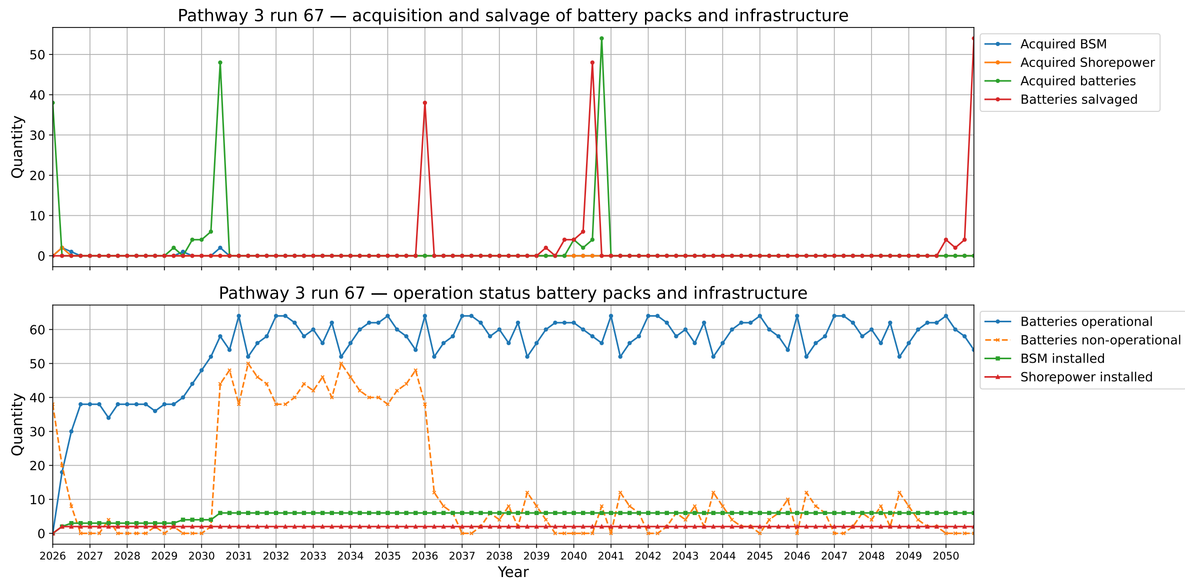


Figure K.9: Pathway 3 run 67: Battery and infrastructure schedule.

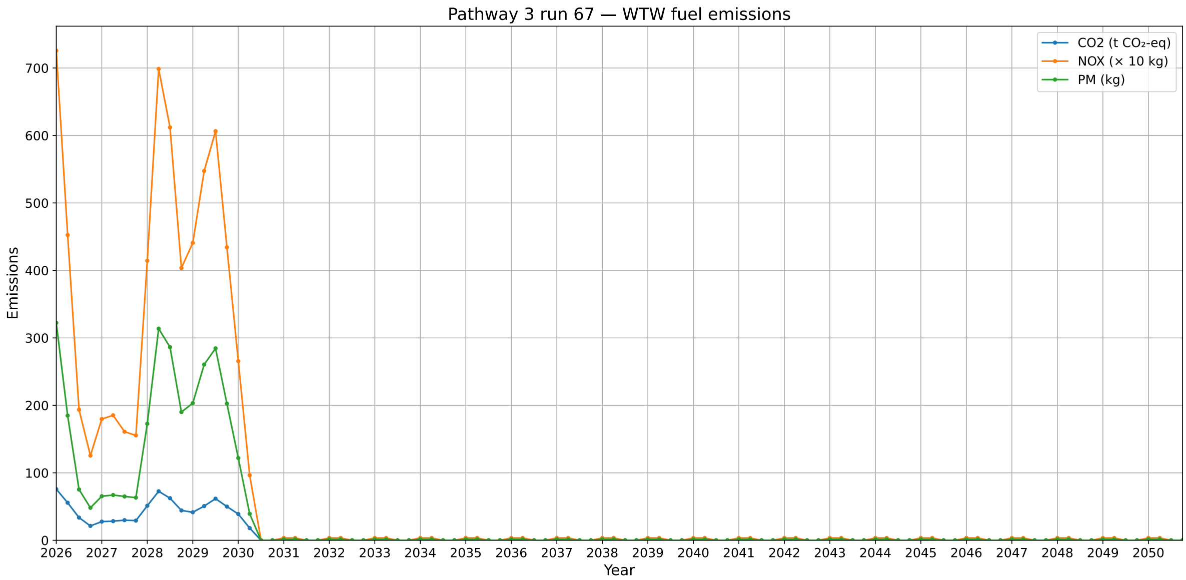


Figure K.10: Pathway 3 run 67: Well to Wake emissions over time.

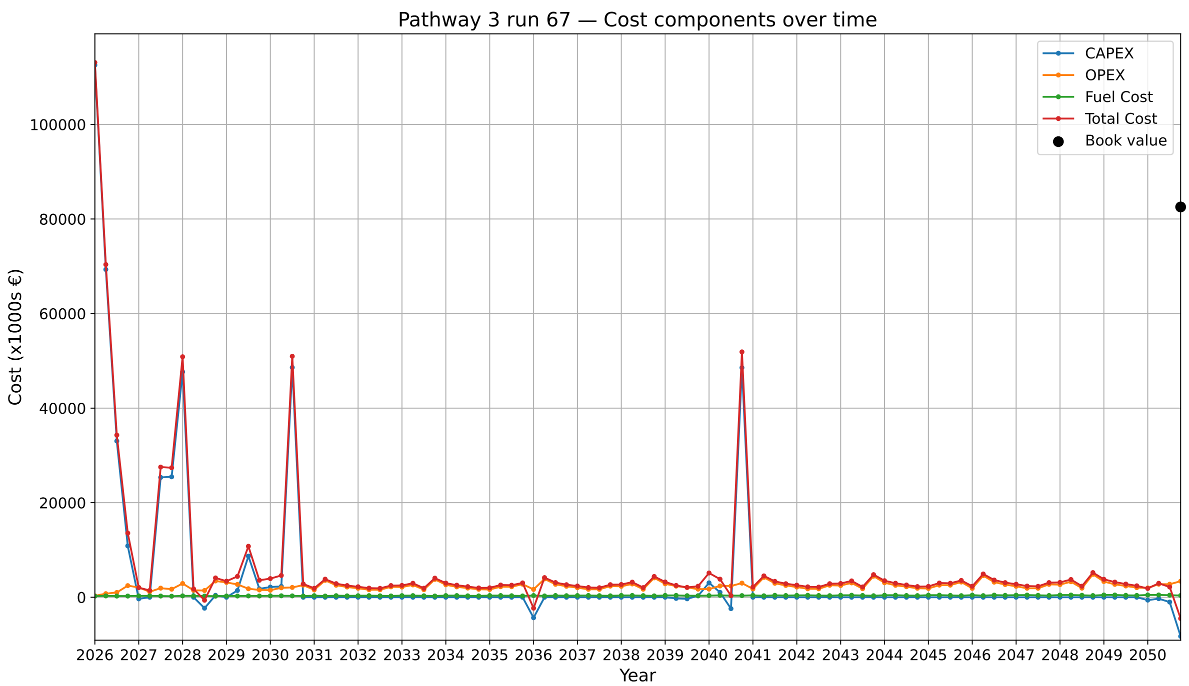


Figure K.11: Pathway 3 run 67: Cost breakdown over time.

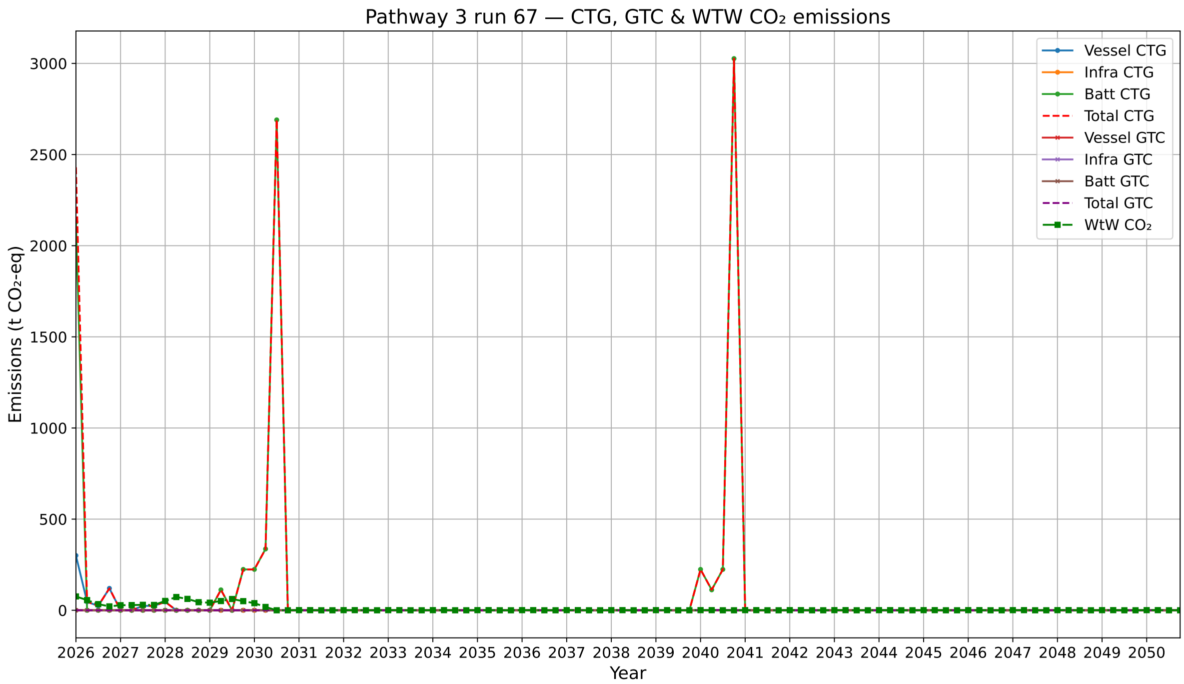


Figure K.12: Pathway 3 run 67: Emission breakdown over time.

Pathway 4

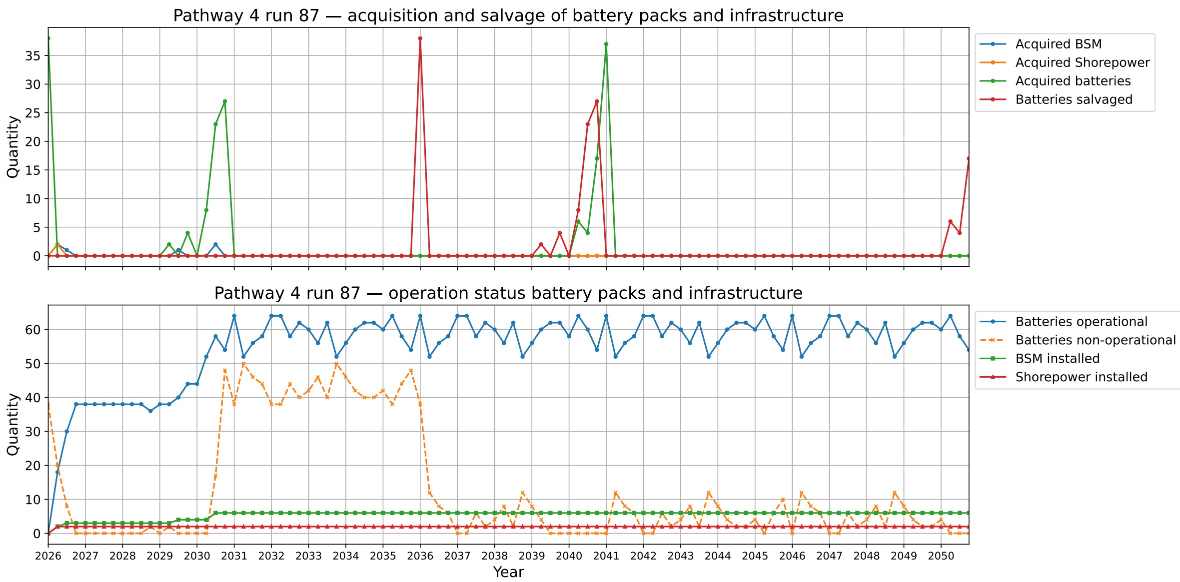


Figure K.13: Pathway 4 run 87: Battery and infrastructure schedule.

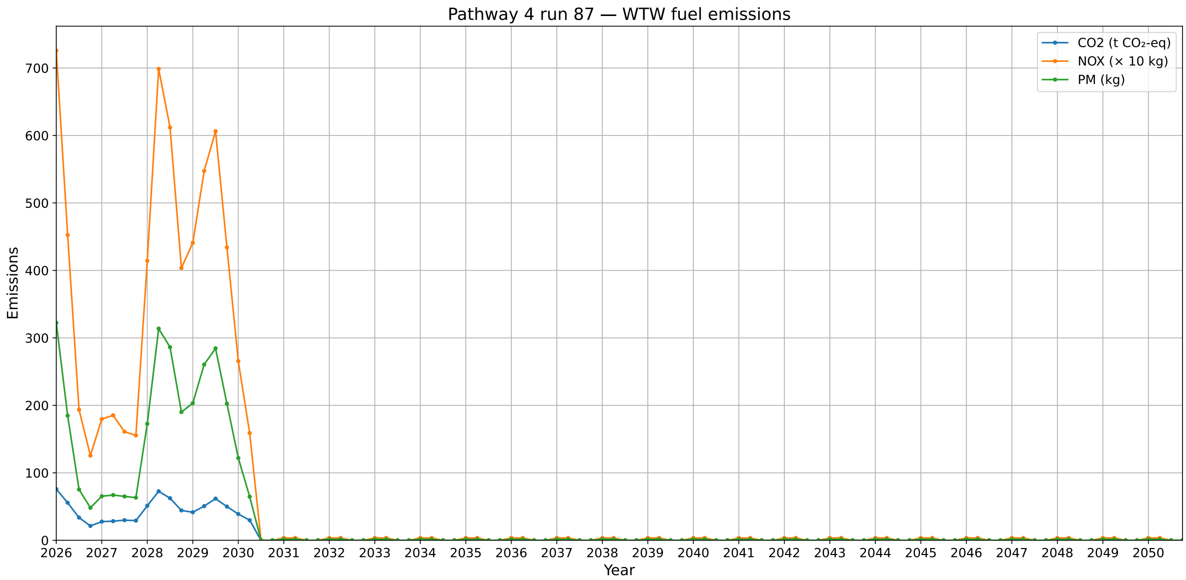


Figure K.14: Pathway 4 run 87: Well to Wake emissions over time.

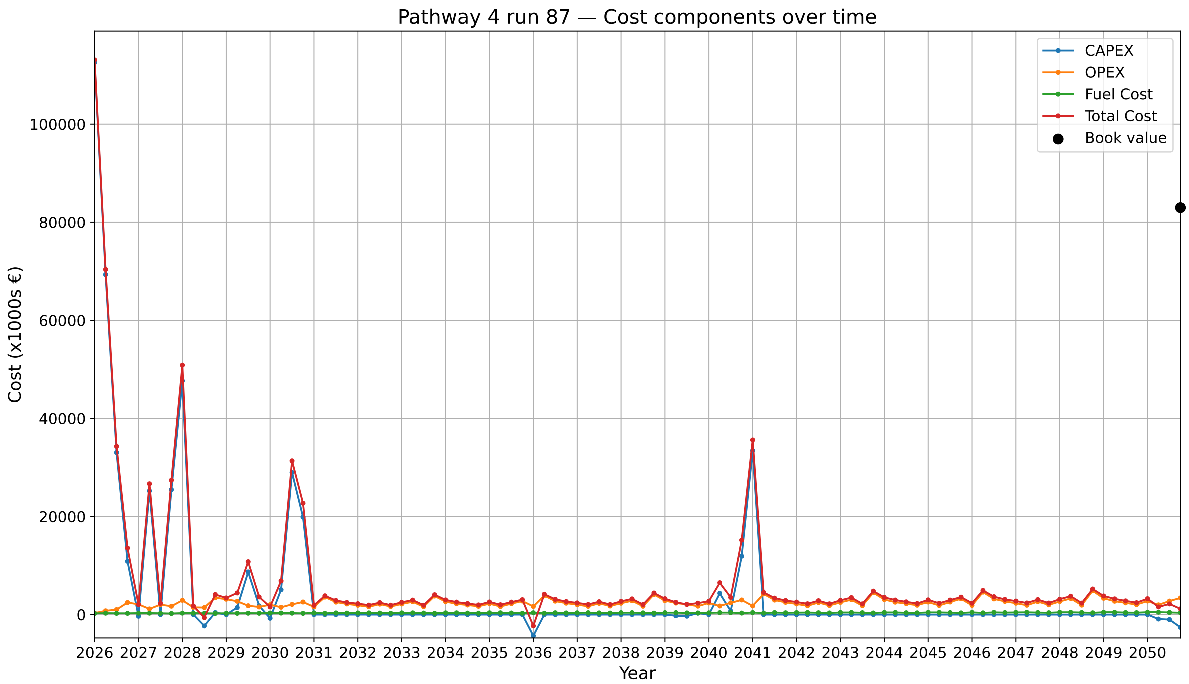


Figure K.15: Pathway 4 run 87: Cost breakdown over time.

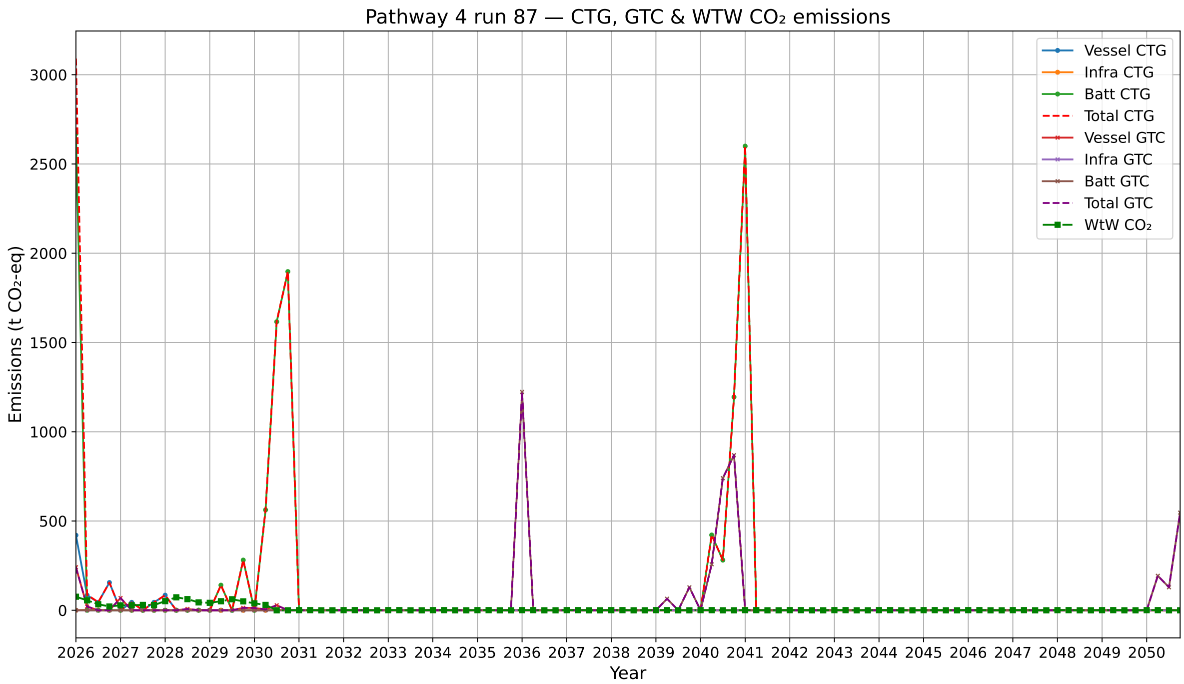


Figure K.16: Pathway 4 run 87: Emission breakdown over time.

Pathway 5

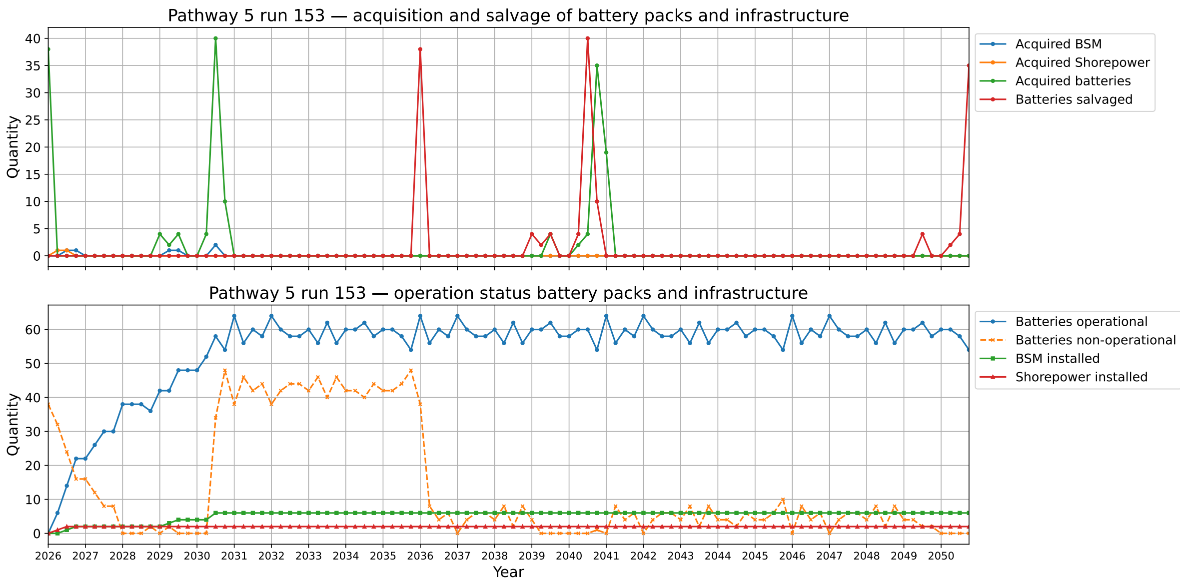


Figure K.17: Pathway 5 run 153: Battery and infrastructure schedule.

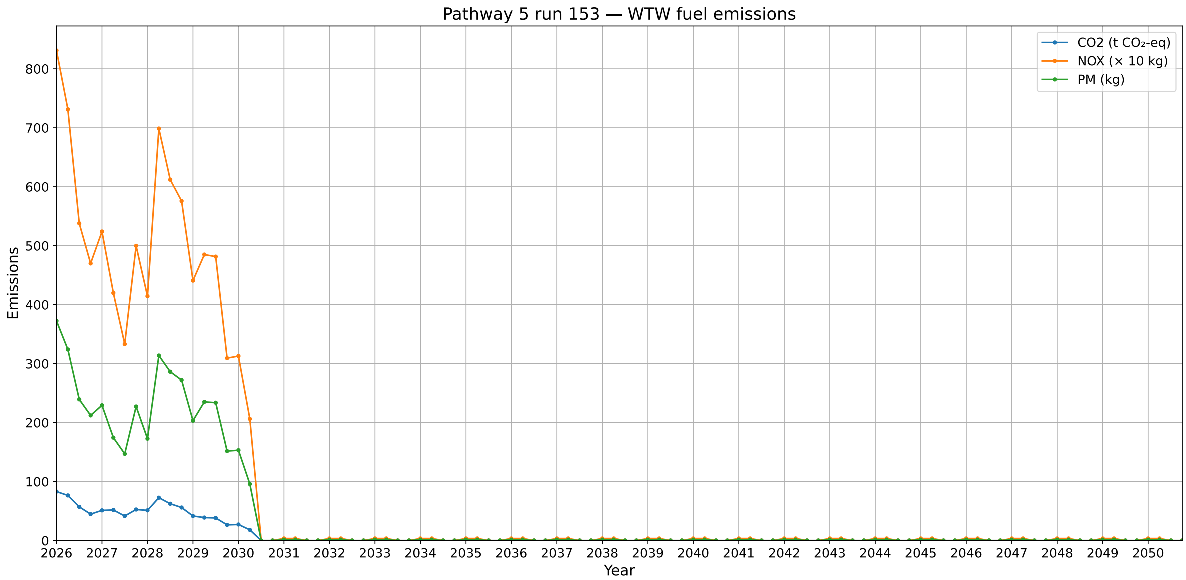


Figure K.18: Pathway 5 run 153: Well to Wake emissions over time.

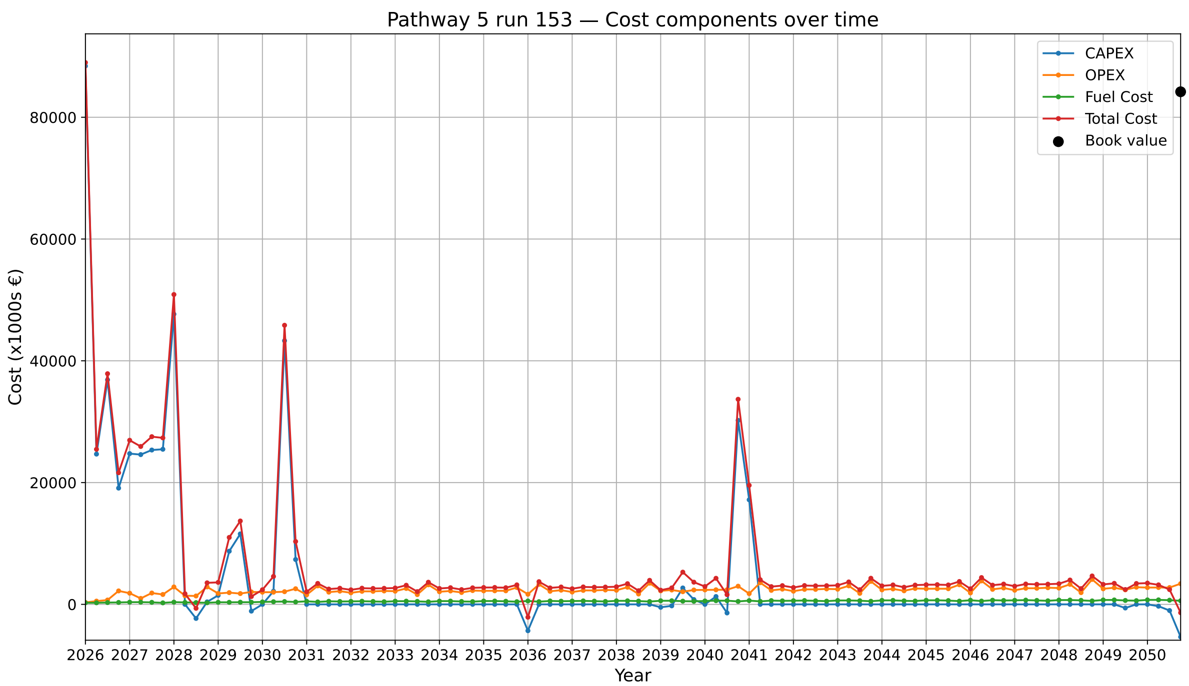


Figure K.19: Pathway 5 run 153: Cost breakdown over time.

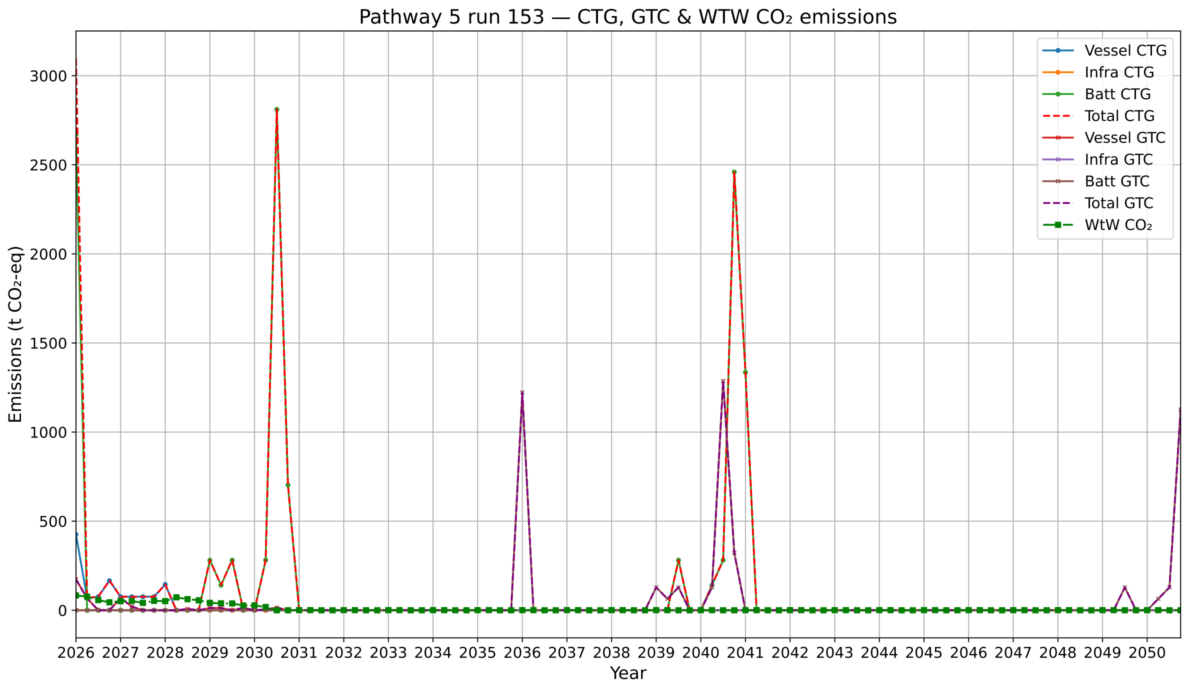


Figure K.20: Pathway 5 run 153: Emission breakdown over time.

Conservative scenario

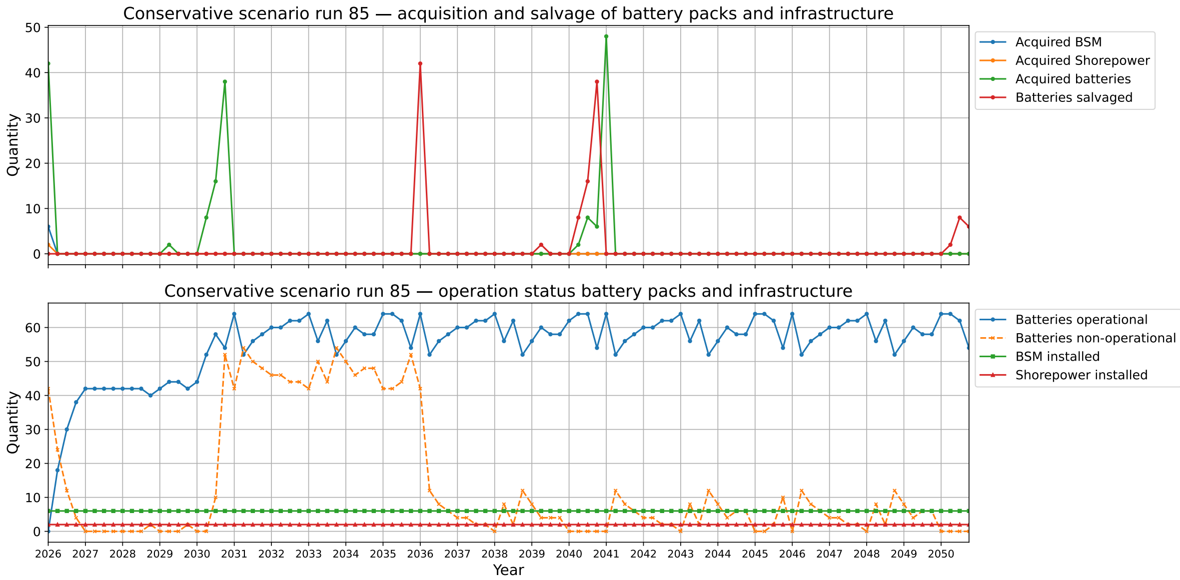


Figure K.21: Conservative scenario run 85: Battery and infrastructure schedule.

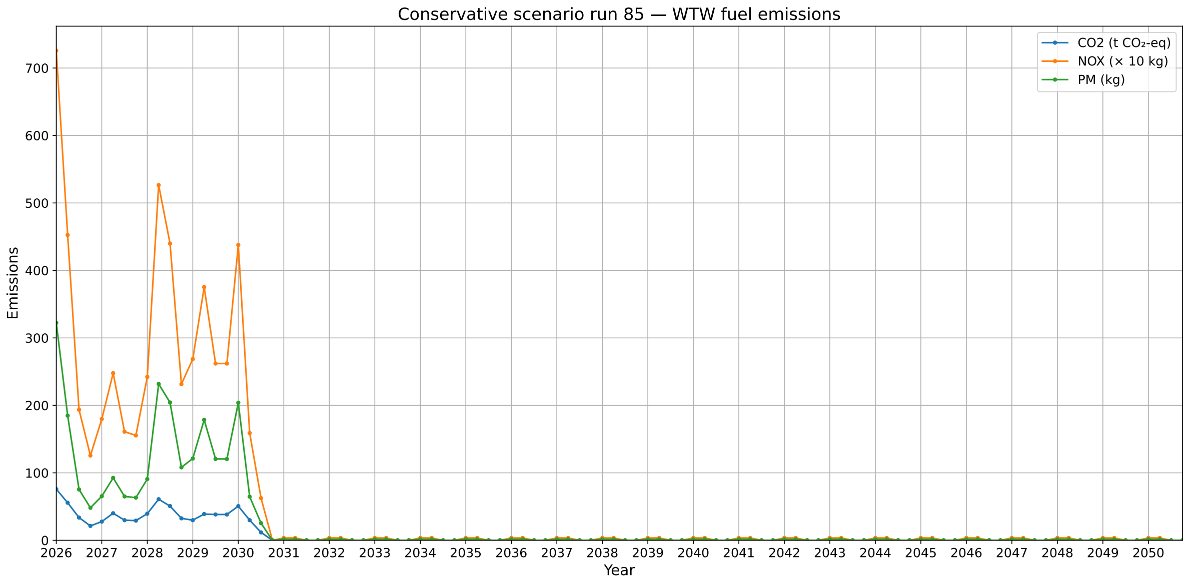


Figure K.22: Conservative scenario run 85: Well to Wake emissions over time conservative scenario.

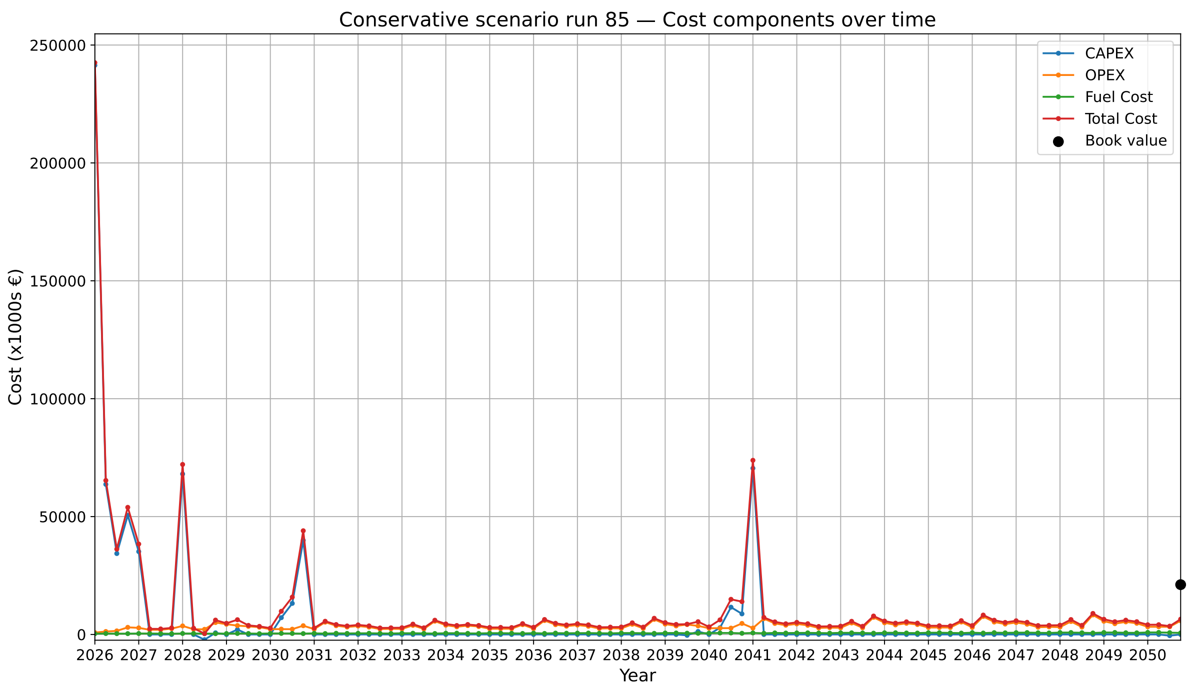


Figure K.23: Conservative scenario run 85: Cost breakdown over time.

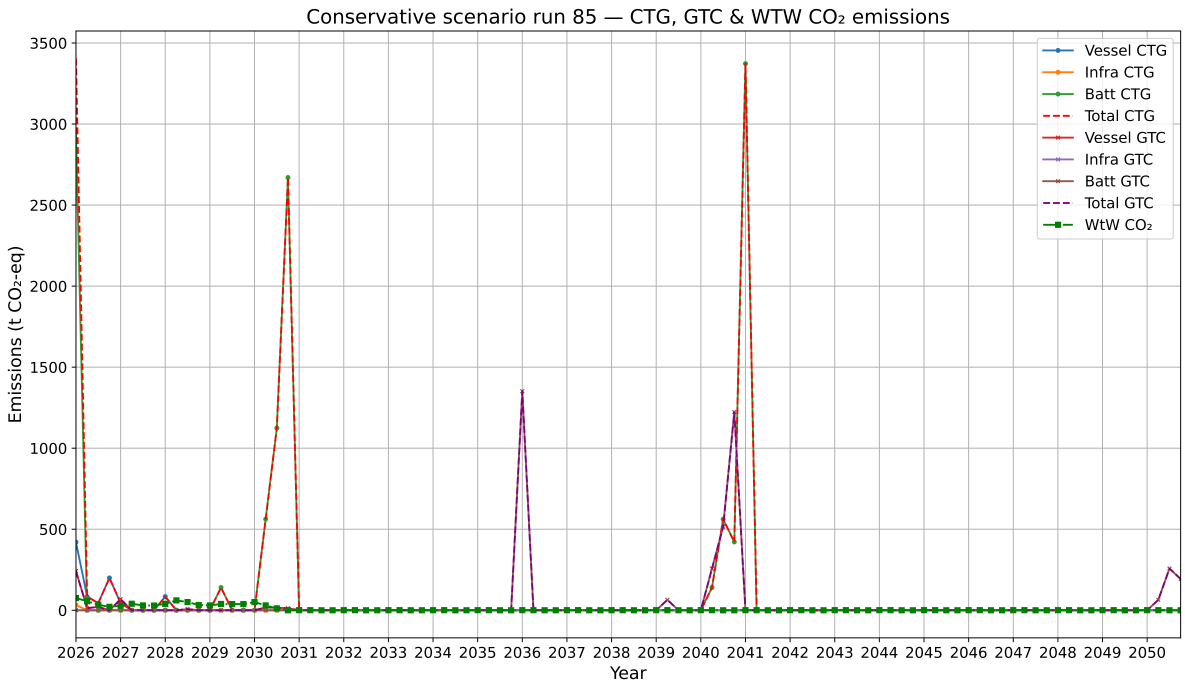


Figure K.24: Conservative scenario run 85: Emission breakdown over time.

Optimistic scenario

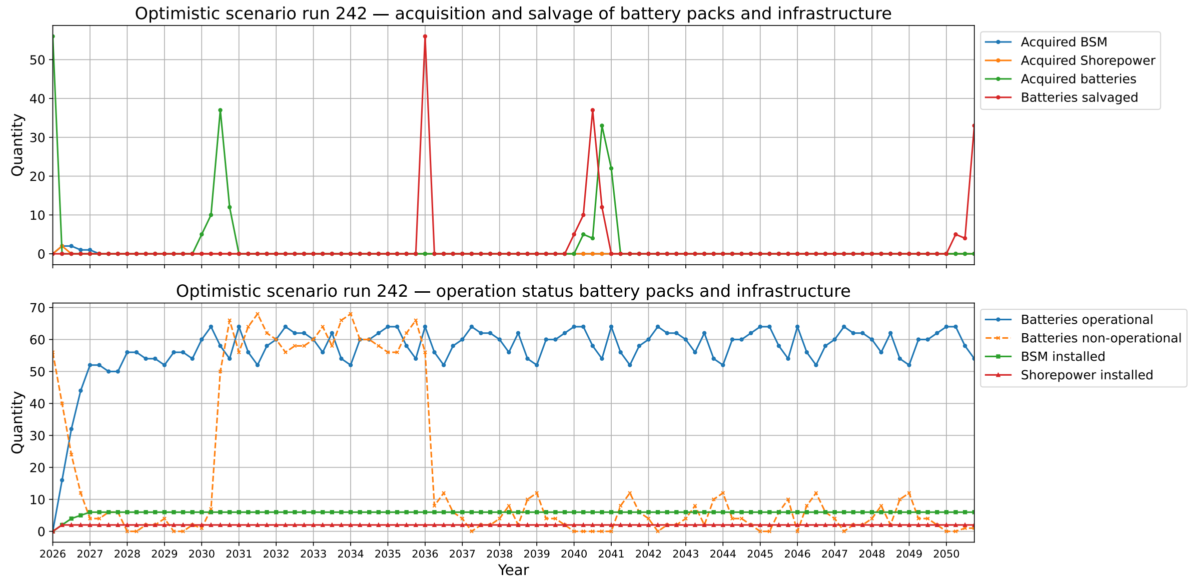


Figure K.25: Optimistic scenario run 242: Battery and infrastructure schedule.

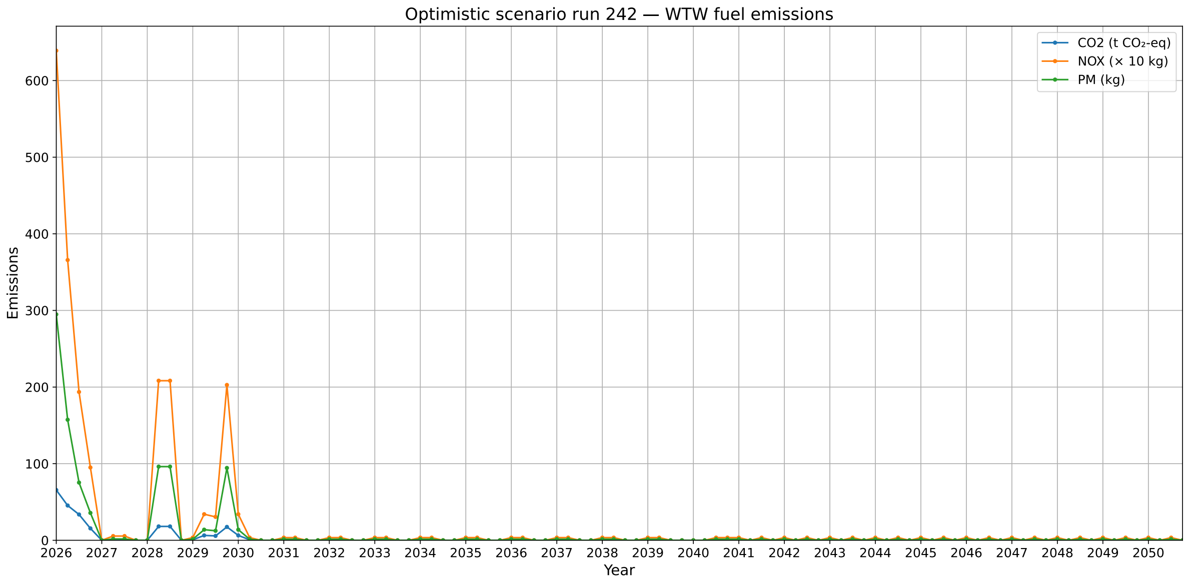


Figure K.26: Optimistic scenario run 242: Well to Wake emissions over time.

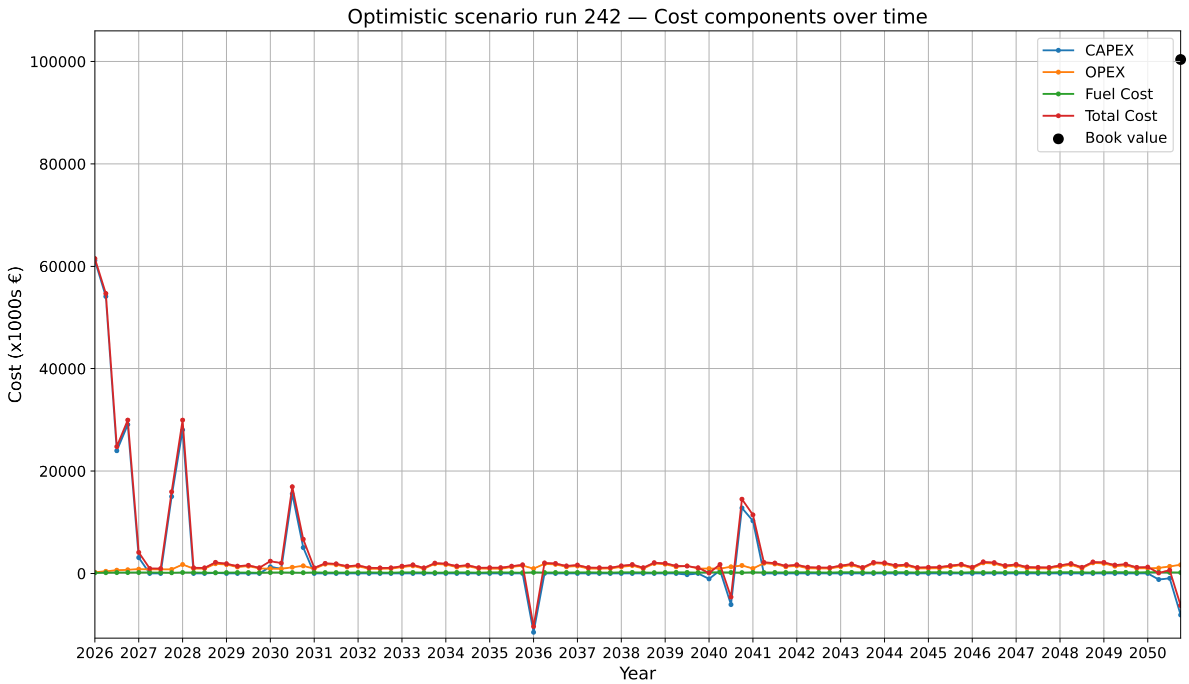


Figure K.27: Optimistic scenario run 242: Cost breakdown over time.

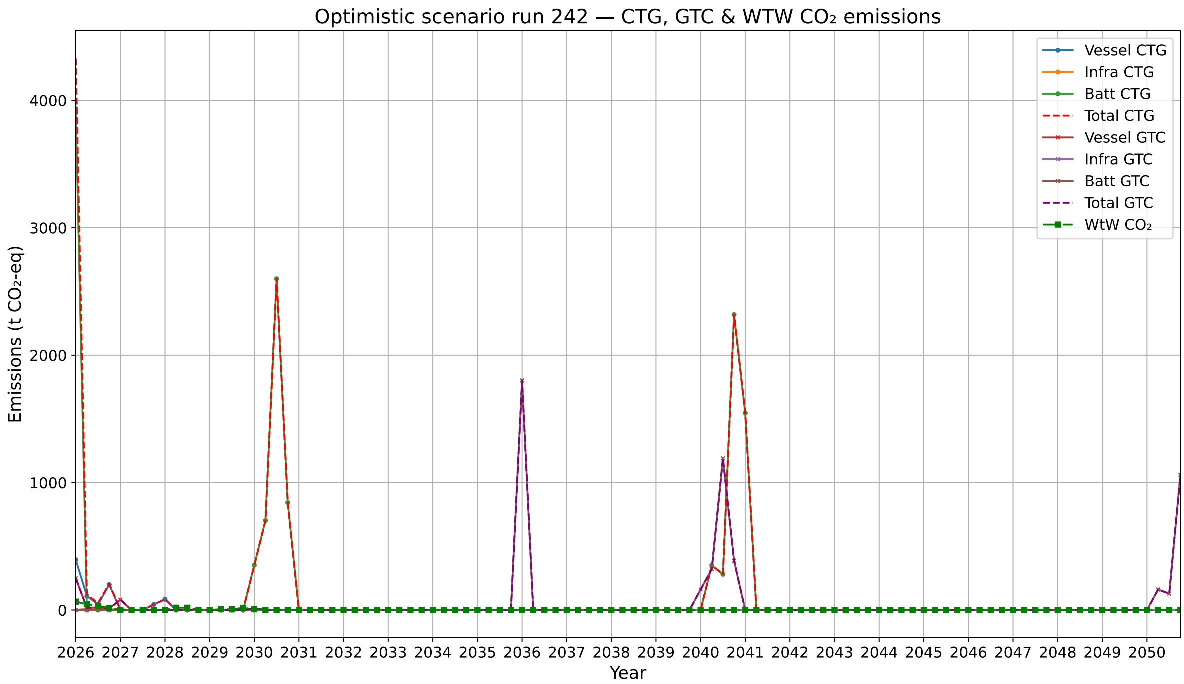


Figure K.28: Optimistic scenario run 242: Emission breakdown over time.

CO2 depreciation scenario

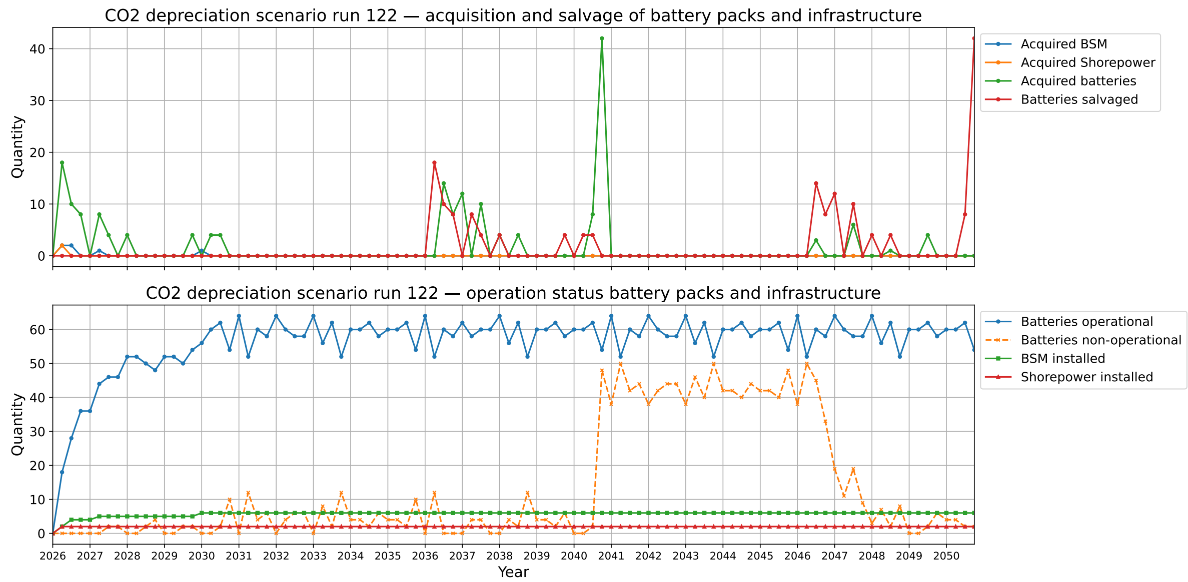


Figure K.29: CO_2 depreciation scenario run 122: Battery and infrastructure schedule.

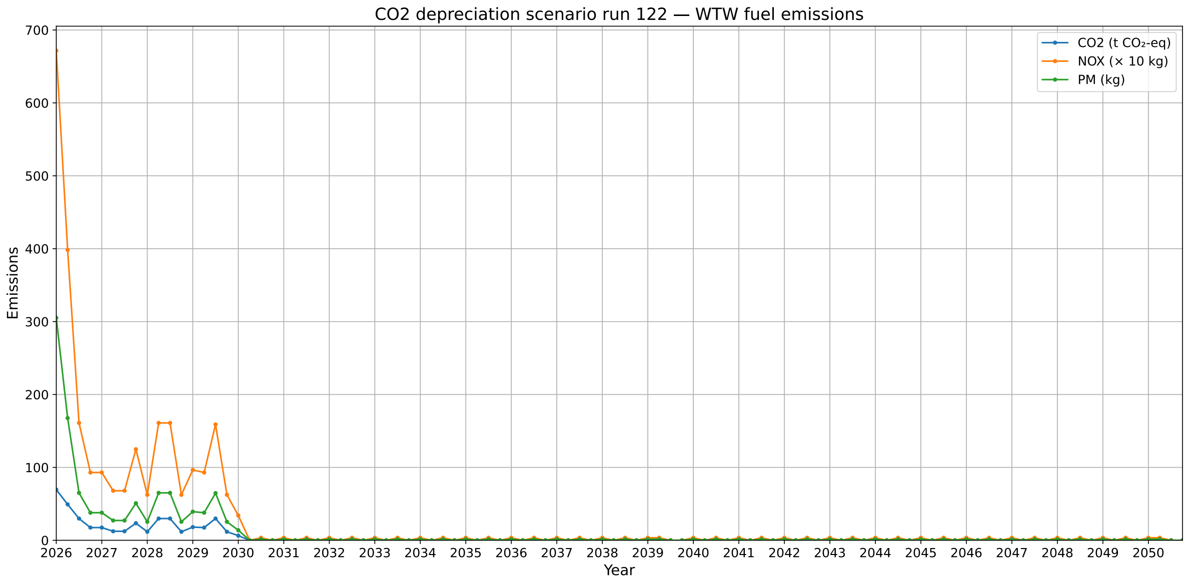


Figure K.30: CO_2 depreciation scenario run 122: Well to Wake emissions over time.

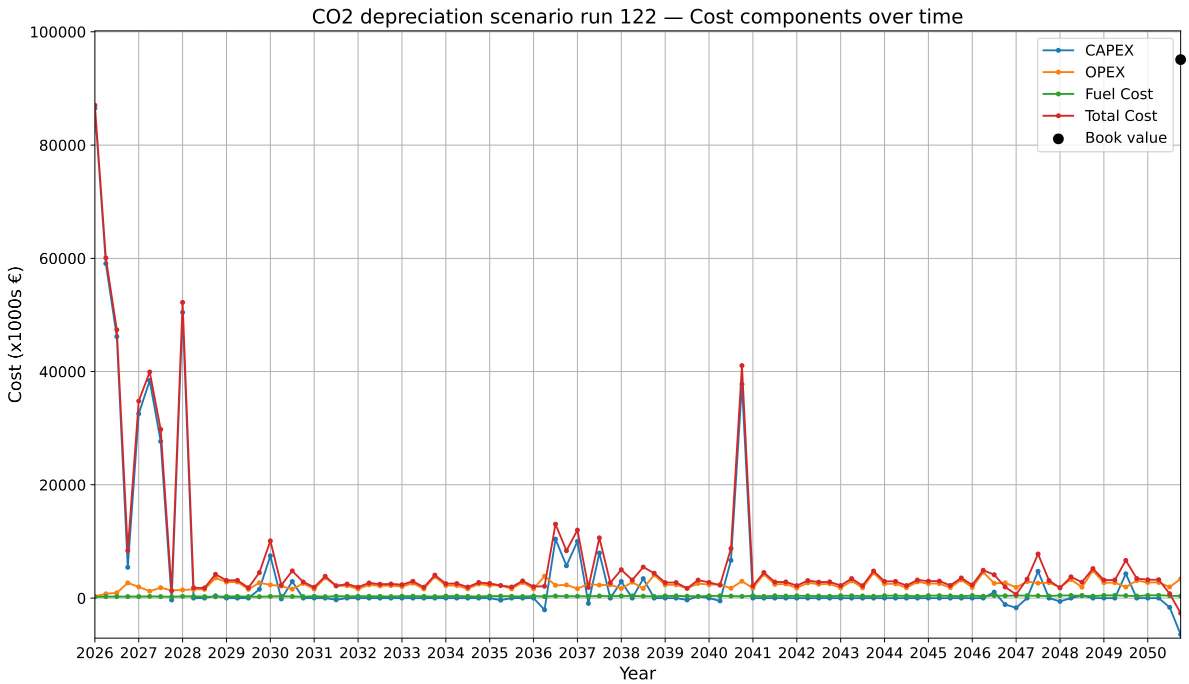


Figure K.31: CO_2 depreciation scenario run 122: Cost breakdown over time.

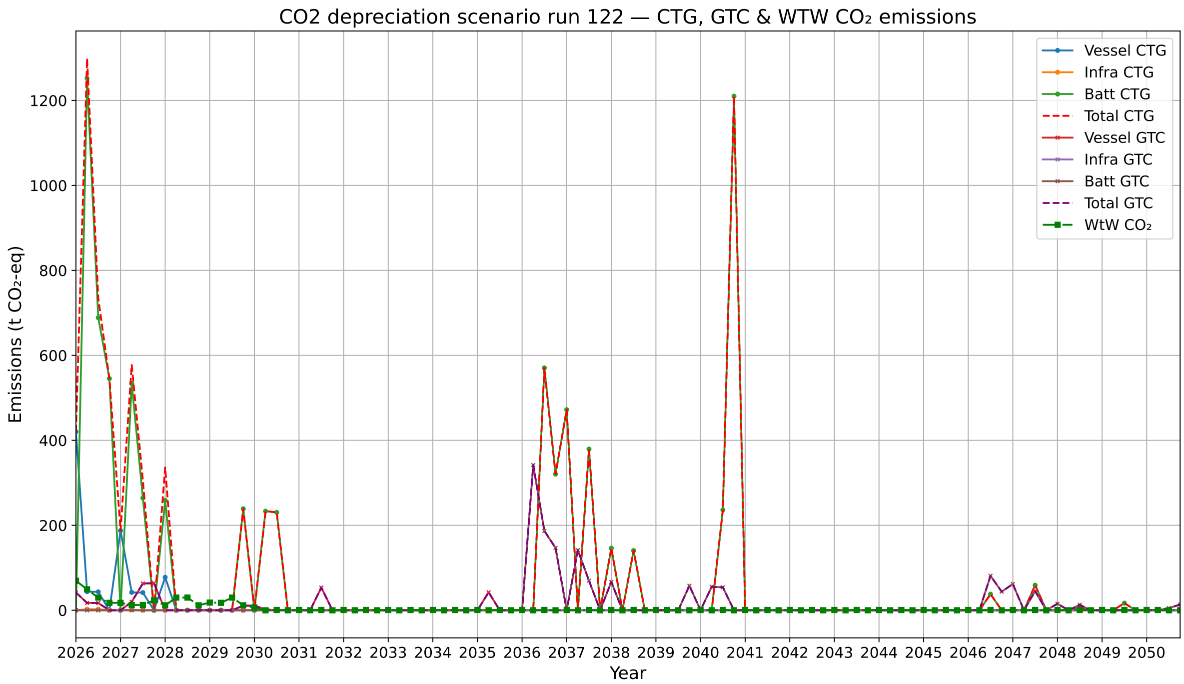


Figure K.32: CO_2 depreciation scenario run 122: Emission breakdown over time.