sustainability

MDPI

Article

Structured Expert-Rule Pipelines for Analytics Method Selection in Supply Chain Systems

Alessandro Tufano

Department of Industrial Engineering, Alma Mater Studiorum—University of Bologna, Viale Risorgimento 2, 40136 Bologna, Italy; alessandro.tufano@studio.unibo.it

Abstract

Selecting an appropriate analytical method for a given supply chain decision problem is a recurring challenge for both researchers and practitioners. Existing taxonomies classify problems and methods independently but offer limited guidance on how data-availability conditions constrain the feasible set of methods. This paper introduces three structured expert-rule pipelines—descriptive, predictive, and prescriptive—that condition on observable data-availability predicates to prescribe a method family for Supply Chain Systems (SCS). Each pipeline is structured as a binary decision tree whose internal nodes are Boolean predicates on data conditions and whose leaf nodes map to concrete method families. The pipelines are designed to be usable by operations managers with limited statistical training, subject to expert interpretation of the predicates and to validation of the resulting outputs. The framework is domain-independent: while it applies to any supply chain decision context, this paper develops its sustainability dimension by grounding all application examples in environmental, social, and governance (ESG) scenarios—including carbon footprint monitoring compliant with the Greenhouse Gas (GHG) Protocol and ISO 14064, energy consumption forecasting for Corporate Sustainability Reporting Directive (CSRD) and European Sustainability Reporting Standards (ESRS) reporting, and green vehicle routing aligned with ISO 14040/44 life-cycle assessment principles—to demonstrate how the same methodological structure supports circular economy analytics, ESG reporting, and decarbonisation objectives. A retrospective application to a proprietary operational distribution dataset with two use cases shows that the pipelines are executable and discriminating: the same physical system routes to different feasible method sets depending on data content and depth, and each unreachable branch is paired with the specific field whose collection would enable it. The paper contributes an expert-rule-based selection tool that bridges the gap between problem taxonomy and practical method deployment.

Keywords: supply chain systems; decision pipelines; analytics method selection; sustainable supply chain management; circular economy; ESG reporting

Check for updates

Academic Editor: Koichi Murata

Received: 16 September 2026
Revised: 25 October 2026
Accepted: 26 October 2026
Published: 31 October 2026

Copyright: © 2026 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

Supply chain management research has grown substantially in recent decades, driven by the proliferation of data-intensive technologies under the Industry 4.0 and 5.0 paradigms [1,2]. The expanding methodological repertoire available to researchers and practitioners, ranging from classical mathematical programming to deep learning, creates a paradox: while the diversity of tools increases the potential for tailored solutions, it also raises the barrier to selecting the most appropriate method for a given decision context [3,4].

The existing literature addresses this challenge through two complementary lenses. On the one hand, taxonomies of decision problems in Supply Chain Systems (SCS) have been proposed to classify the types of decisions that arise in production facilities, warehousing systems, and distribution networks [5]. On the other hand, methodological classifications group solving approaches by their epistemological role: optimisation, simulation, prediction, or empirical analysis [6,7]. However, the connection between these two dimensions remains largely implicit: practitioners must independently assess which method is suitable for which problem under specific data conditions.

This gap has practical consequences. A production manager who needs to forecast demand may not know whether time-series methods or supervised machine learning (ML) is more appropriate given the available data. Similarly, a logistics planner facing a vehicle routing problem may be unaware that the absence of a validated cost function precludes certain optimisation approaches and necessitates simulation instead. The absence of a structured, data-conditioned selection framework creates an efficiency gap that limits the uptake of advanced analytical methods in industrial practice [8,9].

The selection challenge has intensified with recent technological developments. Generative AI and large language models (LLMs) are entering supply chain management, offering new capabilities for demand forecasting, scenario generation, and autonomous decision-making [10,11]. At the same time, causal AI and digital twin technologies are enabling prescriptive analytics that go beyond correlation-based prediction to identify intervention effects under uncertainty [12,13]. These developments expand the feasible set of analytical methods but also deepen the selection problem: practitioners must now assess not only data availability but also the maturity of causal models, the availability of validated digital twins, and the organisational readiness to deploy autonomous agents [14,15]. A structured, data-conditioned selection framework is therefore more urgently needed in the current technological landscape.

This paper addresses this gap by introducing three structured expert-rule pipelines—one per analytics objective (Descriptive, Predictive, Prescriptive)—that translate observable data-availability conditions into recommended method families. The research question is: How should practitioners select analytics methods to support ESG reporting, circularity tracking, and decarbonisation targets when data-availability conditions vary across sustainability contexts? The framework is domain-independent: the pipelines apply to any supply chain decision context regardless of the specific objective. This paper develops the sustainability dimension of the framework by grounding all application examples in environmental, social, and governance (ESG) scenarios—including carbon footprint monitoring compliant with the GHG Protocol and ISO 14064 [16] energy consumption forecasting for CSRD/ESRS reporting, and green vehicle routing aligned with ISO 14040/44 [17] life-cycle assessment principles—to demonstrate how the same methodological structure supports circular economy analytics, ESG reporting, and decarbonisation objectives.

The remainder of this paper is organised as follows. Section 2 reviews the relevant literature on decision-making frameworks, method-selection in SCS, and the role of analytics in sustainable supply chains. Section 3 presents the three structured expert-rule pipelines with their structure, predicates, and leaf-level method prescriptions. Section 4 applies the pipelines to an operational distribution dataset with two use cases, judging every predicate field by field, checking which branches are reachable, and naming the data investments that would unlock each higher leaf. Section 5 discusses managerial implications, sustainability relevance, limitations, and directions for future research. Section 6 concludes the paper.

2. Background

2.1. Decision-Making Frameworks in Supply Chain Systems

The classification of decision problems in SCS has a long tradition in operations research and supply chain management. Anthony [18] proposed one of the earliest hierarchical classifications, distinguishing strategic, tactical, and operational decisions. This tri-level framework has been widely adopted and extended in logistics and supply chain modelling [19]. Jahani et al. [5] provided a comprehensive review of problem classification and method evaluation criteria across production, distribution, and logistics phases.

These frameworks offer valuable vocabularies for categorising decisions but do not systematically link problem types to solving methods under explicit data-availability conditions. The present work builds on these foundations by constructing decision pipelines that bridge the gap between problem classification and method selection.

2.2. Analytics Families and Method Classification

The classification of analytical methods into descriptive, predictive, and prescriptive families is widely attributed to Delen [20] and has become a standard reference in both academic and industrial contexts [7]. Descriptive analytics characterises phenomena through statistical and empirical methods; predictive analytics forecasts future outcomes using statistical and machine-learning models; prescriptive analytics identifies actions that optimise a defined objective.

In the SCS domain, Wang et al. [3] observe that the research corpus still lacks a comprehensive perspective on analytics, and the mapping from problem type to method remains underexplored. Mentzer [4] and Lambert and Enz [21] highlight the need to balance academic rigour with industrial applicability, a tension that a structured selection framework can help resolve. Recent systematic reviews confirm that while AI and machine learning applications in supply chain management have grown substantially, the integration of descriptive, predictive, and prescriptive analytics into unified decision pipelines remains underdeveloped [10,14].

2.3. Data-Driven Versus Model-Driven Prescriptive Analytics

The prescriptive pipeline operates within a rapidly evolving landscape of data-driven prescriptive analytics. Bertsimas and Kallus [22] provide a comprehensive framework bridging predictive and prescriptive analytics, showing how machine-learning predictions can be integrated into optimisation models. Elmachtoub and Grigas [23] formalise the "predict, then optimise" paradigm, where a prediction model is trained jointly with a downstream optimisation objective. The present pipeline simplifies this landscape by conditioning on observable data states rather than on the predictive-optimisation coupling; it acknowledges that practitioners must first determine whether they have a model, data, or neither—before deciding how to integrate prediction and optimisation. The pipeline is thus complementary to, rather than a substitute for, the predict-then-optimise literature.

2.4. Data-Driven Versus Model-Driven Decision Making

The debate between data-driven and model-driven approaches is central to the present study. Provost and Fawcett [24] and Chen et al. [25] analyse whether data-driven technologies should integrate with or replace traditional model-driven engineering approaches. Trkman et al. [26] and Hofmann [27] confirm that integration can extend the benefits of both methodologies, provided that specific conditions—such as data readiness and stakeholder willingness to share data—are met.

This paper contributes to this debate by making data-availability conditions an explicit conditioning variable in the method-selection process. The decision pipelines do not advocate for one approach over the other; instead, they map data states to the feasible set of methods, enabling informed choices that respect both data constraints and problem structure.

2.5. Analytics for Sustainable Supply Chain Management

The integration of analytics into sustainable supply chain management has gained significant attention. Mustafa et al. [28] provide a systematic review of AI applications in sustainability reporting, highlighting how machine learning algorithms can process extensive ESG data, recognise patterns, and produce actionable insights more rapidly than conventional approaches. Strazzullo et al. [29] examine the role of AI in enhancing transparency and accountability in ESG reporting, particularly in the European fashion industry.

For circular economy applications, Kokkinou and Mitas [30] conduct a systematic review of AI applications in circular supply chains, finding that AI contributes to waste reduction, resource efficiency, and reverse logistics optimisation. Farshadfar et al. [31] review machine learning applications across circular supply chain stages—from supply and manufacturing to waste management—and find that ML contributes significantly to supplier selection, operational optimisation, and waste reduction. Kumar and Khedlekar [32] propose an analytics-driven circular supply chain framework integrating economic, environmental, and behavioral dimensions.

Digital twin technologies are enabling new sustainability pathways. Phonthanukitithaworn and Ketkaew [33] investigate how digital twin adoption enhances green supply chain performance through real-time optimisation and resource efficiency. Arij and Areej [34] demonstrate the transformative role of digital twins in enabling data-driven decisions for net-zero supply chain goals.

2.6. Sustainability-Specific Data Challenges

Sustainability analytics face distinctive data-availability challenges that generic method-selection frameworks do not address. Under the GHG Protocol and ISO 14064, Scope 1 and Scope 2 emissions are typically measurable from utility bills and fuel records, but Scope 3 emissions, which often represent the largest share of a firm's carbon footprint, depend on supplier-provided estimates, industry-average factors, and spend-based calculations that vary in accuracy and granularity [28]. This heterogeneity across emission scopes is a first-class data-availability condition that the descriptive pipeline must accommodate.

The Corporate Sustainability Reporting Directive (CSRD) and European Sustainability Reporting Standards (ESRS) require firms to disclose forward-looking ESG targets and transition plans, creating demand for predictive analytics on sustainability metrics. However, historical ESG data series are often short, fragmented, or non-existent for emerging impact categories (e.g., biodiversity, water stress), limiting the applicability of classical time-series methods and necessitating alternative approaches such as LLM-based zero-shot forecasting from knowledge graphs.

Circular economy tracking under extended producer responsibility (EPR) regulations requires linking product-level material composition data to end-of-life recovery rates—a prescriptive problem where the objective function (maximise recovery) and constraints (minimum recycled content, landfill limits) must be jointly optimised. When material flow models are available, mathematical programming applies; when they are not, the prescriptive pipeline must fall back to simulation or LLM-based reasoning over textual case studies.

2.7. Existing Method-Selection Tools and Related Work

Decision-tree-style method selection is not new, and this subsection positions the present contribution against existing aids. Three families of tools are relevant. First, practitioner flowcharts such as the scikit-learn algorithm cheat sheet [35] map data characteristics to estimators, but they cover only predictive modelling and are tied to one library. Second, algorithm-selection research formalises the choice of a solver for a given instance as a learning problem over features of the problem and the algorithms [36], a line of work that recent surveys have extended to automated, per-instance selection across discrete and continuous problems [37]; it is rigorous but assumes a fixed, pre-established benchmark set and rarely addresses data-availability constraints or prescriptive formulations. Third, forecasting-method selection guidance and decision-support frameworks for supply chain analytics [3,38] offer domain advice but do not expose explicit, checkable data predicates. Table 1 compares these tools with the present pipelines along four dimensions: applicable analytics scope, use of data-availability conditions, transparency of the selection rules, and operational usability of the output.

Table 1. Comparison of existing method-selection tools with the proposed expert-rule pipelines.

Tool	Analytics Scope	Data-Availability Conditions	Rule Transparency	Output Usability
scikit-learn cheat sheet [35]	Predictive only	Implicit (label and feature presence)	Flowchart, library-specific	Estimator recommendation
Algorithm-selection methods [36,37]	Any solver	Instance features, not data states	Formal but opaque to non-experts	Best algorithm, benchmarked
Forecasting-method guidance [38]	Predictive (time series)	Series properties	Heuristic guidelines	Forecast method
Supply chain analytics frameworks [3]	Descriptive–prescriptive (conceptual)	Not formalised as predicates	Conceptual	Research agenda
Present pipelines	Descriptive–predictive–prescriptive	Explicit Boolean predicates	Rule-by-rule justification (Section 3.5)	Method family per data state

The comparison clarifies the contribution: unlike library-specific flowcharts and formal algorithm-selection methods, the present pipelines span all three analytics families, make data-availability conditions explicit and checkable, and expose their ordering for inspection. The comparison also delimits the claim—the framework does not replace principled algorithm selection, but provides a transparent, domain-independent front end that routes a decision context to a method family before finer algorithm selection takes place.

3. Structured Expert-Rule Pipelines

This section introduces three structured expert-rule pipelines—one per analytics objective (Descriptive, Predictive, Prescriptive)—that condition on data-availability state to prescribe a method family. The pipelines are structured as binary decision trees whose internal nodes are Boolean predicates on data conditions and whose leaf nodes map to concrete method families (e.g., mathematical optimisation, simulation, time-series forecasting, supervised machine learning, inferential statistics).

3.1. Notation and Variables

Table 2 introduces the notation used throughout the pipelines.

Table 2. Notation of the parameters of the decision pipelines.

Variable	Description
X	Any available input data useful to solve the decision problem
y	Variable to describe or predict
$\hat{y}$	Observed/measured values of the variable y to use as validation dataset (i.e., ground truth)
D	Decision variables to set in a prescriptive problem
f	Mathematical function of X, such as $y = f(X)$
g	Mathematical function of D, such as $y = g(D)$
$\mathcal{T}$	Unstructured text corpus or document collection relevant to the decision context (e.g., maintenance logs, supplier reports, incident descriptions)
$\mathcal{K}$	Knowledge graph or structured ontology linking entities and relationships within the supply chain domain

Each pipeline poses questions on:

- The need to set up decision variables D to solve the problem (e.g., a set of values indicating a specific configuration of entities and processes considered as a feasible solution);
- The availability of observations on previous realisations of the solution;
- The completeness and accuracy of the previous realisations dataset;
- The knowledge of the boundary conditions of X.

The framework begins with a preliminary classification step: the practitioner determines whether the problem requires setting decision variables D (Prescriptive), forecasting a target variable y (Predictive), or characterising y without prediction (Descriptive). This classification is a user judgement based on the decision context, not part of the tree itself. Once classified, the corresponding pipeline is entered.

The three trees form a partition of the space of decision contexts: a context is assigned to exactly one pipeline based on whether decision variables D must be set (Prescriptive), whether the target variable y is to be forecast (Predictive), or whether it is to be described (Descriptive). Within each tree, the predicates are evaluated in a stated priority order (Section 3.5): the tree returns the highest-priority feasible method family, while the lower-priority branches that remain jointly feasible are reported as alternatives rather than being ruled out. The ordering is therefore an explicit design choice, justified rule by rule in Section 3.5, and not a claim of strict mutual exclusivity.

3.2. Descriptive Expert-Rule Pipeline

The descriptive decision pipeline (Figure 1) focuses on characterising a random variable y, often representing a key performance indicator. No decision variables D are defined. Algorithm 1 formalises the decision logic.

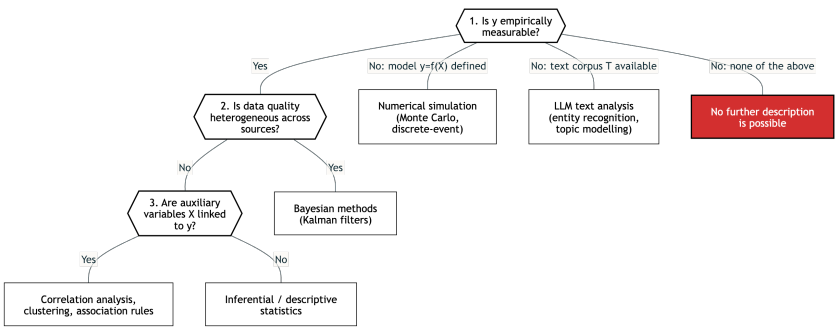


Figure 1. Descriptive decision pipeline. Numbered nodes correspond to Algorithm 1. Red nodes indicate terminal states where no viable method is available.

Algorithm 1 Descriptive Expert-Rule Pipeline

Require: Target variable y; auxiliary variables X; text corpus $\mathcal{T}$; analytical model f
Ensure: Method family for descriptive analytics
if y is empirically measurable **then**
  if data quality is heterogeneous across sources **then**
    return Bayesian methods (e.g., Kalman filters)
  else if auxiliary variables X linked to y are available **then**
    return Correlation analysis, clustering, or association rules
  else
    return Inferential/descriptive statistics
  end if
else if analytical model $y = f(X)$ is defined **then**
  return Numerical simulation (Monte Carlo, discrete-event)
else if text corpus $\mathcal{T}$ is available **then**
  return LLM-based text analysis (entity recognition, topic modelling)
else
  return No further description is possible
end if

The pipeline begins at Node 1 by asking whether the target variable y is empirically measurable. If y cannot be measured directly but an analytical model $y = f(X)$ can be defined from domain knowledge, numerical simulation (e.g., Monte Carlo or discrete-event simulation) is used to analyse y's behaviour under the assumed functional form. If neither measurements nor a model are available, the pipeline reaches a terminal node indicating that no further description is possible.

When y is measurable, the pipeline proceeds to Node 2, which evaluates whether data quality is heterogeneous across observers or sources. If multiple measurement sources exist with varying accuracy—for instance, two sensors at the end of a production line with different calibration profiles—Bayesian methods such as Kalman filters are applied to combine the data sources and produce a fused estimate with improved accuracy.

If data quality is homogeneous, the pipeline proceeds to Node 3, which asks whether additional related variables X are linked to y. If no auxiliary variables are available, inferential or descriptive statistics (e.g., confidence intervals, hypothesis tests, distribution fitting) are used to characterise y. If auxiliary variables are present, correlation analysis, clustering, or association rules are employed to uncover hidden patterns in the joint distribution of (X, y).

Another pathway emerges when unstructured text data $\mathcal{T}$—such as maintenance logs, supplier correspondence, or incident reports—are available but no structured measurements of y exist. In this case, large language models (LLMs) can be applied to extract descriptive signals from text corpora through entity recognition, sentiment analysis, or topic modelling [10]. This pathway does not replace the need for empirical measurements but provides a fallback when structured data collection is infeasible, enabling qualitative characterisation of phenomena that are documented only in natural language. As with all LLM-based pathways in this framework, the output carries no accuracy guarantee and should be validated against any available structured measurements; where additional structured data collection is feasible, it should be preferred over text-based inference.

3.3. Predictive Expert-Rule Pipeline

The predictive decision pipeline (Figure 2) focuses on forecasting a random variable y. No decision variables D are defined. Algorithm 2 formalises the decision logic.

Algorithm 2 Predictive Expert-Rule Pipeline

Require: Target variable y; auxiliary variables X; knowledge graph $\mathcal{K}$; text corpus $\mathcal{T}$
Ensure: Method family for predictive analytics

```
if y is empirically measurable from historical records then
    if auxiliary variables X are available then
        if text corpus T is available for feature augmentation then
            return Hybrid: supervised ML on (X, y) augmented with LLM-extracted features from T
        else
            return Supervised machine learning on (X, y)
        end if
    else
        return Time-series forecasting (ARIMA, exponential smoothing)
    end if
else if knowledge graph K is available then
    return LLM with RAG: zero-shot/few-shot forecasting
else
    return Prediction not feasible
end if
```

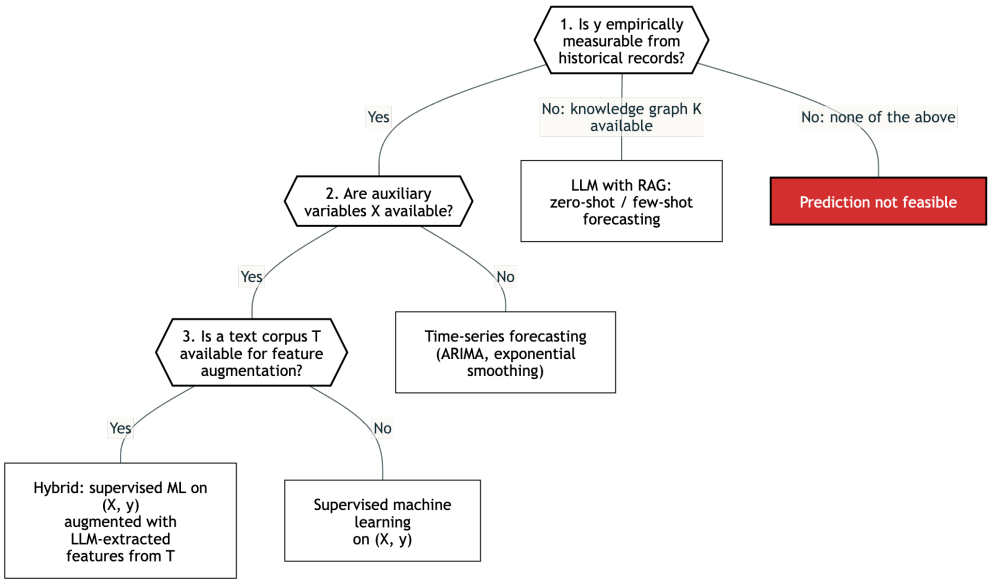


Figure 2. Predictive decision pipeline. Numbered nodes correspond to Algorithm 2. Red nodes indicate terminal states where no viable method is available.

The pipeline begins at Node 1 by asking whether the target variable y is empirically measurable from historical records. If no historical observations exist, classical statistical forecasting is not feasible. However, when a knowledge graph $\mathcal{K}$ or a domain-specific ontology is available, large language models (LLMs) equipped with retrieval-augmented generation (RAG) can produce zero-shot or few-shot forecasts by reasoning over structured domain knowledge and analogous historical contexts [11]. This pathway enables prediction even in data-sparse settings, provided that relevant textual or relational knowledge is accessible. Zero-shot or few-shot LLM forecasts provide no calibration or optimality guarantee; they should be treated as exploratory and, where possible, benchmarked against time-series baselines or superseded once historical data can be collected.

When historical measurements of y are available, the pipeline proceeds to Node 2, which asks whether additional related variables X are available. If no auxiliary variables exist, time-series forecasting methods—such as ARIMA or exponential smoothing—are applied to predict future values of y from its own temporal structure. If auxiliary variables

X are available (e.g., weather conditions, day of the week, promotional events), the pipeline proceeds to Node 3, which asks whether unstructured text data $\mathcal{T}$ are available for feature augmentation. If no text data exist, supervised machine learning methods—including linear regression, decision trees, support vector machines, or neural networks—are trained on the joint dataset (X, y) to enhance predictive accuracy. If text data are available, a hybrid pathway becomes feasible: LLMs extract synthetic feature representations from $\mathcal{T}$ (e.g., equipment condition indicators from maintenance logs, sentiment indices from news feeds), which are then fed as additional covariates into supervised models [10]. This augmentation strategy extends the predictive pipeline to incorporate information sources not captured in structured databases.

3.4. Prescriptive Expert-Rule Pipeline

In prescriptive decision-making, the objective is to determine a set of decision variables D that optimises a specific outcome for y (e.g., minimising cost or maximising efficiency). The prescriptive pipeline (Figure 3) depends on the availability of empirical measurements and mathematical models. Algorithm 3 formalises the decision logic.

Algorithm 3 Prescriptive Expert-Rule Pipeline

Require: Outcome variable y; decision variables D; analytical model g; validation data $\hat{y}$; historical tuples (D, y); text corpus $\mathcal{T}$; knowledge graph $\mathcal{K}$
Ensure: Method family for prescriptive analytics
1: **if** analytical model $y = g(D)$ is available **then**
2: **if** validation dataset $\hat{y}$ is available **then**
3: **return** Model-based optimisation (LP, IP, heuristics, metaheuristics)
4: **else if** dataset of feasible (D, y) tuples is available **then**
5: **return** Data-driven prescription (learned policies, predict-then-optimise)
6: **else**
7: **return** Model uncalibrated—collect validation data or use simulation-based optimisation
8: **end if**
9: **else if** y is empirically measurable **then**
10: **if** dataset of feasible (D, y) tuples is available **then**
11: **return** Data-driven prescription (learned policies, predict-then-optimise)
12: **else**
13: **return** Simulation-based optimisation
14: **end if**
15: **else if** text corpus $\mathcal{T}$ is available **then**
16: **return** LLM-based qualitative reasoning from case histories, grounded in $\mathcal{K}$ when available
17: **else**
18: **return** No viable prescription is possible
19: **end if**

The pipeline begins at Node 1 by asking whether an analytical model $y = g(D)$ relating decision variables to the outcome is available. If a model exists, the pipeline proceeds to Node 2, which asks whether a validation dataset $\hat{y}$ is available. If validation data are available, the model is calibrated and model-based optimisation methods—including linear programming, integer programming, heuristics, or metaheuristics—are applied to identify the optimal prescription. If no validation data are available, but a dataset of historical (D, y) tuples exists, data-driven prescription (learned policies or predict-then-optimise) is returned. Otherwise, the model remains uncalibrated and the practitioner is directed to collect validation data or to use simulation-based optimisation.

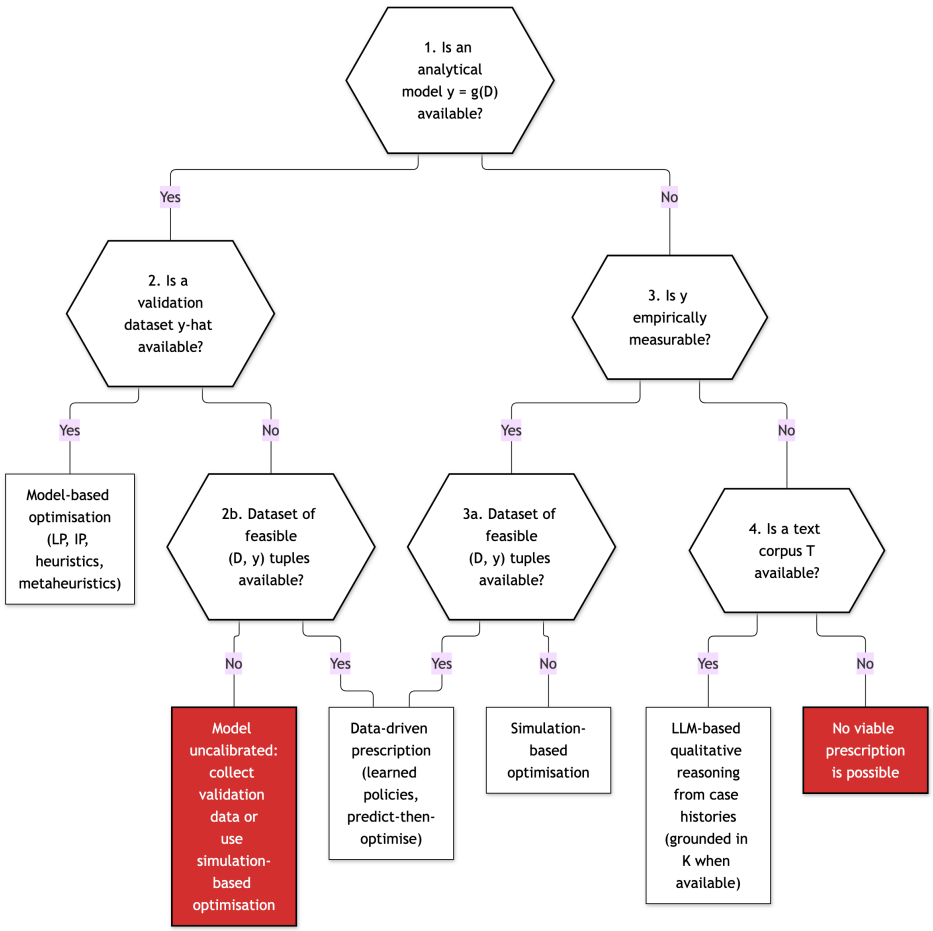


Figure 3. Prescriptive decision pipeline. Numbered nodes correspond to Algorithm 3. Red nodes indicate terminal or fallback states where no viable method is currently available.

If no analytical model exists, the pipeline proceeds to Node 3, which asks whether the outcome variable y can be empirically measured. If y is measurable, the pipeline asks whether a dataset of historical (D, y) tuples—recording past decisions and their observed outcomes—is available. If such a dataset exists, data-driven prescriptive methods are applied: learned policies, predict-then-optimise frameworks, or empirical risk minimisation can identify good prescriptions from historical patterns. If no historical (D, y) tuples are available, simulation-based optimisation is used to evaluate alternative configurations by generating synthetic outcomes from a simulator.

If neither a model nor measurable outcomes exist, the pipeline proceeds to Node 4, which asks whether a text corpus $\mathcal{T}$ describing past decision outcomes is available. When unstructured text data $\mathcal{T}$ are present—for example, textual case histories documenting decision-action-outcome patterns—LLM-based agents can perform qualitative prescriptive reasoning by extracting decision heuristics from these textual records [39]. This pathway does not yield mathematically optimal solutions and its reliability depends on the quality and completeness of the text corpus; it should be treated as an exploratory complement to, rather than a substitute for, quantitative methods.

The model-based leaf explicitly includes heuristics and metaheuristics because sustainable and closed-loop supply chain design problems are frequently combinatorial and multi-objective, and are solved in current practice by metaheuristic search over economic and environmental criteria rather than by exact methods. Recent work on perishable and closed-loop networks illustrates this pattern: metaheuristic algorithms are used to design poultry and agri-food closed-loop supply chains and are evaluated against both solution

quality and computational effort [40]. The pattern recurs across the recent literature: systematic reviews of sustainable supply chain network design report metaheuristic solution methods as a dominant approach for the multi-objective, nonlinear, and uncertain models that characterise this problem class [41,42], and dedicated studies design closed-loop agri-food networks with multi-objective metaheuristics evaluated on the same two axes [43]. The pipeline therefore returns a coarse method family—heuristics or metaheuristics—that the practitioner refines into a specific algorithm, for example a multi-objective evolutionary method, with the objective set and constraint structure determining the choice.

3.5. Operational Definitions and Justification of the Predicates

To make the pipelines reproducible, this subsection gives an operational (checkable) definition for each predicate and justifies, rule by rule, the priority ordering introduced in Section 3. The rules were not derived algorithmically: they were formulated by the authors through domain-expert reasoning grounded in the reviewed literature and in established practice, each leaf being selected as the method family best supported under the corresponding data condition, and Table 3 records the alternatives that were considered and not prioritised.

The predicates are defined as follows:

- y is empirically measurable: Numeric values of y with documented units are available for analysis, whether produced by a physical instrument or recorded by an upstream system of the data owner; the provenance (metered or provider-recorded) is recorded as an attribute of the outcome field. Values that the practitioner must herself compute from other populated fields via a documented model do not satisfy this predicate; they trigger the analytical-model branch instead. Two counting rules apply consistently to every use case: the working minimum of 30 observations counts records of y (e.g., shipment legs), not time stamps, while temporal coverage is assessed separately (at least two periods, and a full seasonal cycle where seasonality matters).
- Data quality is heterogeneous: two or more sources provide values for the same quantity whose disagreement exceeds a stated tolerance (e.g., a relative deviation above a threshold δ), an illustrative working value to be set by the practitioner together with the data owner, or whose metadata indicate different measurement principles or accuracy classes.
- Auxiliary variables X linked to y are available: variables X are recorded for every observation of y and are documented, or shown in a preliminary analysis, to have a non-negligible association with y (e.g., an absolute correlation above a threshold, again an illustrative working value rather than an optimised cut-off, or a mechanistic link from domain knowledge).
- An analytical model $y = f(X)$ or $y = g(D)$ is available: an explicit functional or simulation model with defined parameters exists; it is validated when observed values $\hat{y}$ of its output are available for calibration or validation.
- A text corpus $\mathcal{T}$ is available: a sufficiently large and topically relevant document collection exists; a knowledge graph $\mathcal{K}$ provides a structured ontology over the relevant entities and relations that can ground LLM reasoning.

Table 3 summarises the rationale for each leaf-level rule and the alternatives that were not prioritised. The ordering follows three principles: (i) prefer calibrated quantitative methods over model-based approximations, and both over data-driven learning when no validation is possible; (ii) when a probabilistic model of uncertainty is unavailable, prefer methods that require the fewest assumptions (time series before machine learning); and (iii) treat LLM-based pathways as last-resort fallbacks whose outputs are exploratory.

Table 3. Rule-by-rule justification of the pipeline ordering.

Rule	Rationale	Alternatives Not Prioritised
Heterogeneous quality → Bayesian fusion	Quantifies and combines source uncertainty; standard for reconciling inventories	Simple averaging ignores differing accuracy
X linked to y → supervised ML	Leverages covariates when joint (X, y) data exist	Time series ignores available covariates
No X → time series	Only the temporal structure of y is available	Supervised ML requires covariates
No measurements, $\mathcal{K}$ available → LLM/Retrieval-Augmented Generation (RAG)	Reasons over domain knowledge when no series exist	Quantitative forecasting is infeasible
Model + validated $\hat{y}$ → optimisation	Calibrated model supports exact or near-exact search	Heuristic search is unnecessary
Model, no $\hat{y}$ but (D, y) → data-driven	Historical decisions ground a learned policy	Uncalibrated model gives unreliable prescriptions
Measurements + (D, y) → data-driven	Outcomes observed for past decisions	Simulation is only needed without (D, y)
No model, no data, $\mathcal{T}$ → LLM reasoning	Extracts heuristics from case histories	No quantitative method is available

Finally, data availability is necessary but not sufficient for method selection. Scale (data volume and dimensionality), computational cost, interpretability, and auditability determine feasibility in practice. These criteria are complementary to the pipelines and are flagged as an extension of the predicates in Section 5.4.

The numerical values that appear in these definitions—the minimum of 30 observations, the tolerance δ, and the association threshold—are conventional working defaults, not parameters optimised in this work: 30 is a long-standing rule of thumb for the sample size at which asymptotic statistical results become serviceable, while δ and the association threshold are domain-dependent and should be set by the practitioner together with the data owner. The pipelines report which leaf a given setting selects; a different setting may select a different leaf, and this sensitivity is part of the classification-granularity limitation discussed in Section 5.5.

3.6. Worked Example: Tracing One Path Through a Pipeline

To illustrate how a practitioner reads a tree, consider a manufacturer that must forecast monthly energy consumption y for CSRD/ESRS reporting (Section 4). At Node 1 the practitioner checks whether y is measurable from historical records: smart-meter data are available, so the predicate is true and the tree moves to Node 2. At Node 2 the question is whether auxiliary variables X (e.g., outdoor temperature, production schedule) are linked to y: initially only the meter series is available, so X is absent and the tree returns time-series forecasting (ARIMA or exponential smoothing). If the firm later integrates temperature and production-planning variables, Node 2 becomes true; Node 3 then asks whether a text corpus $\mathcal{T}$ exists for feature augmentation. With no such corpus, the tree returns supervised machine learning trained on (X, y). The example shows a single, reproducible traversal and makes explicit the data investment—collecting X—that would shift the recommendation.

4. Application to an Operational Distribution Dataset

A real distribution dataset is used to evaluate the pipelines with real-world data. The dataset records the movement of goods across a logistics network together with its environmental and economic consequences. It is organised into five structures:

1. The shipment-flow structure holds one record per shipment leg: it links an origin and a destination node through a transport mode over a distance, carries a load, and reports the resulting CO_2-equivalent and air-pollutant quantities, fuel and energy consumption, and cost.
2. The product-impact structure repeats these impact fields disaggregated by product.
3. The shipment-record structure holds the logistics metadata of each shipment—order, delivery and arrival dates, carton, weight and unit-load quantities, item and transport mode, and the commercial-assignment fields.
4. The network structure describes the nodes (facilities with attributes and roles) and the directed edges (links with distance, travel time and frequency) of the distribution network.
5. The fleet structure provides dimensional, capacity and emission/cost factors for the available transport types.

The same structures are populated by two use cases identified as "use case A" (UC-A) and "use case B" (UC-B); Table 4 reports, field by field, what each field encodes and whether each use case populates it.

Table 4. Incidence matrix for the distribution dataset: for each field, what it encodes and whether use case UC-A or UC-B populates it. Rows are grouped by the six dataset structures defined in the text. Both use cases belong to the distribution system. ● = populated for all records; ◦ = not populated; † = partially populated (about 80%).

Field	What the Field Encodes	UC-A	UC-B
Shipment-flow			
Shipment/shipping code	Identifier grouping the legs of one shipment	●	●
Origin and destination nodes	Network nodes where a leg begins and ends	●	●
Distance (km)	Length of the leg	◦	●
Transport mode	Transport type used for the leg	●	●
Weight	Mass moved in the leg	◦	●
Pallets	Number of unit loads moved	◦	●
CO_2-equivalent	Total greenhouse-gas impact of the leg	●	◦
Air pollutants	NOx, NH_3	●	◦
Fuel and energy	Fuel and energy consumed in the leg	◦	●
Cost	Monetary transport cost of the leg	●	●
Product-impact			
Product/item identity	Product or item carried	●	●
Shipment-record			
Dataset depth (distinct days)	Number of distinct days available in the records	365	25
Shipment legs (records)	Number of shipment-leg records in the dataset	~58,500	~7500
Order date	Date on which the shipment was requested (lead time)	◦	◦
Delivery date	Date on which a shipment leg is delivered	●	●
Arrival date	Date on which goods arrive at the destination node	●	◦
Carton quantity	Number of cartons in the shipment	● †	●
Unit-load quantity	Number of unit loads per shipment	◦	◦
Network			
Node coordinates	Latitude and longitude of a facility	◦	●
Node role: customer	Facility that receives goods	● †	◦
Node role: distributor	Facility that consolidates or redistributes goods	◦	●
Postal code	Facility location code	● †	●
Edge travel time	Expected travel time along a link	◦	◦
Edge frequency	Number of trips per link per period	◦	◦
Fleet			
Fleet emission and cost factors	Per-transport-type emission, fuel and cost coefficients (14 types)	●	●

The data are proprietary and are drawn from two operational systems of two different companies, used here anonymously. UC-A is the dataset of a third-party freight forwarder whose operational load unit is the pallet; it covers 365 distinct days and about 58,500 shipment legs. UC-B is the dataset of a client company whose operational load unit is the parcel; it covers 25 distinct days and about 7500 shipment legs. The CO_2-equivalent and air-pollutant fields of UC-A are populated from emission parameters produced by the forwarder's own calculation systems, i.e., they are recorded upstream of this study by the data owner, and the per-leg distance used in those calculations is not included in the extract. The provenance of every outcome field (metered, provider-recorded, or analyst-derived) is recorded as an attribute and is used when the measurability predicate is judged (Table 5).

Before tracing the trees, every predicate of Section 3.5 is judged for both use cases under the same counting and provenance rules. Table 5 records the judgement and the evidence for each predicate; the subsections below refer back to it.

Table 5. Judgement of each predicate for the two use cases, applying the counting and provenance rules of Section 3.5 consistently to both.

Predicate	Rule Applied (Section 3.5)	UC-A	UC-B
y empirically measurable	Values of y recorded (metered or provider-recorded); analyst-derived values do not count	True: CO_2-equivalent and pollutants recorded by the forwarder's calculation systems (provider-recorded; 58,500 legs)	False: no outcome field recorded; values only derivable by the analyst from fuel and fleet factors
Data quality heterogeneous	Repeated measurements of the same quantity	False: a single source per quantity	False: a single source per quantity
Auxiliary X linked to y	Covariates recorded jointly with y	True: mode, nodes, product, cost (distance and load absent)	True: distance, weight, mode, network geometry
Model $y = f(X)$ or $y = g(D)$	Explicit functional or simulation model, evaluable from populated fields	False in the extract: the upstream computation cannot be re-instantiated without distance	True: fleet factors applied to distance, weight, and mode
Validation data $\hat{y}$	Observed values against which the model output can be checked	Not applicable (no evaluable model in the extract)	Partial: metered fuel checks the energy component of the model, not the CO_2-equivalent directly
Dataset of (D, y) tuples	Recorded decisions paired with recorded outcomes	True: routing decisions paired with recorded CO_2-equivalent	Weak: 7500 tuples exist, but the outcomes are model-derived rather than recorded and cover only 25 days
Text corpus $\mathcal{T}$ /knowledge graph $\mathcal{K}$	Topically relevant documents or structured ontology	False: structured tables only	False: structured tables only

4.1. Descriptive Pipeline: Emission Monitoring

The problem is to characterise the environmental performance of the distribution operations—to quantify the emissions of each flow and to explain how they vary with route, distance, transport mode and load. The outcome of interest is y, the CO_2-equivalent of a leg.

Use case A. The outcome y is recorded for every record (CO_2-equivalent and air pollutants are populated) by the forwarder's calculation systems (Table 5), from a single, consistent source, so that data quality is homogeneous and the pipeline enters the measured branch. The auxiliary variables that Table 4 reports as populated—origin and destination nodes, transport mode, product identity, delivery date and cost—are linked to y, so that the correlation and clustering leaf is reachable: emissions can be compared across transport modes and routes and legs can be grouped into emission regimes. The two physical drivers that would explain most of the remaining variation, distance and load, are not recorded, so the description must either proceed with the covariates that are present (mode, route, product, cost), which still discriminate emission levels, or be strengthened by supplying the missing drivers. Per-leg distance is recoverable by joining the shipment records to the network edges if those edge lengths are exposed, and weight and pallet counts can be captured at shipment level; populating them would open the distance–emission and load–emission relationships and refine the clustering. With 365 distinct days the characterisation is also seasonally complete.

Use case B. Here the outcome y is not recorded—neither CO_2-equivalent nor air pollutants are populated—so the pipeline does not enter the measured branch. It instead takes the model branch: the fleet table supplies per-transport-type emission factors and the per-leg distance, weight and transport mode are populated, so a functional relation $y = f(X)$ that maps activity data to emissions exists and can be evaluated; because fuel and energy consumption is measured, the outcome can in fact be derived directly by applying the fleet factors to the metered fuel, which is the deterministic limit of the model-based (simulation) leaf. The derived outcome can then be correlated and clustered against the distance, load and network-geometry covariates that UC-B does populate. The limitation here is depth rather than fields: 25 distinct days cannot represent seasonal emission patterns, while the 7500 legs comfortably exceed the 30-observation working minimum, so cross-sectional characterisation over the recorded covariates is statistically supported—what is missing is temporal depth, not observation count.

Bayesian data fusion is unavailable in both use cases, because the dataset contains no repeated measurements of the same quantity that could be combined (for example, metered fuel alongside computed emissions). LLM-based text analysis is likewise unavailable, as there is no unstructured corpus. Two further descriptive extensions are latent in the schema: populating edge travel times would allow origin–destination flows and link-level descriptors to be built, and filling the commercial-assignment fields would allow carrier and container performance to be described. The descriptive pipeline therefore does not merely report which method is admissible today; it names the specific field whose collection would move each use case to a richer leaf.

4.2. Predictive Pipeline: Shipment and Emission Forecasting

The problem is to forecast future shipment volumes and their emissions over the horizon required to plan capacity and decarbonisation actions.

Use case A. The outcome series exists: CO_2-equivalent is recorded and the delivery and arrival dates are populated, so y is measurable from historical records and the pipeline enters the measured branch. Auxiliary variables X are also populated (transport mode, origin and destination nodes, product identity, cost), so the supervised-learning leaf on (X, y) is reachable; and because the series is dense and daily, univariate time-series models (ARIMA, exponential smoothing) are reachable as well. With 365 distinct days both leaves are supported, seasonality included. Accuracy is limited only by the covariates: distance and load are not recorded, so a supervised model cannot use the two variables that most directly

drive emissions; adding them—distance being recoverable from the network edges—would strengthen the model without changing the leaf.

Use case B. The emission outcome is not recorded, so the literal reading of Node 1 is negative and the pipeline reports prediction as not feasible. This is precisely the diagnostic value of the pipeline: it makes explicit that the outcome series must exist before it can be forecast. The data do, however, support a workaround. Because distance, weight, mode and fuel are populated and the fleet factors are available, a synthetic outcome series can be derived from the model and then forecast; alternatively, the measurable flow variables (tonnes moved, pallets) can be forecast directly and propagated through the model to emissions. Three conditions cap the result. First, the derived outcome does not satisfy the measurability predicate, so every forecast on the derived series remains conditional on the emission model. Second, the two counting rules of Section 3.5 must be read separately: the 7500 legs exceed the 30-observation working minimum, so cross-sectional supervised learning on (X, y) over the recorded legs is statistically supported, whereas 25 distinct days are too few to identify seasonal structure, so univariate seasonal models are not defensible and only short-horizon trends within the observed window are. Third, the forecast inherits any bias in the emission factors. Additional data would change the outcome—a longer history would enable seasonal models, and direct CO_2 metering would move UC-B onto the measured branch alongside UC-A.

The data-sparse branch—zero-shot forecasting through retrieval-augmented generation—is unavailable in both use cases, because it requires a text corpus or a textual knowledge base and the dataset provides only structured tables and a node–edge graph.

4.3. Prescriptive Pipeline: Green Routing

The problem is to decide how shipments are routed and which transport modes are used, so as to minimise the resulting CO_2-equivalent (and, jointly, cost) subject to vehicle-capacity and network constraints. The decision variables D are the routing and mode choices.

Use case A. The outcome is recorded, but its provenance explains why the analytical model is nonetheless absent: the CO_2-equivalent values were produced upstream by the forwarder's calculation systems, which applied fleet emission factors to per-leg distance at source; that distance is not included in the extract, so the computation cannot be re-instantiated here. Equivalently, emissions are in the source system computed from the fleet factors as a function of per-leg distance, and per-leg distance is not recorded (nor are node coordinates, from which it could be derived). With no evaluable $y = g(D)$ the pipeline does not enter the model branch. It does enter the measured branch—CO_2-equivalent is populated—and, because the recorded routing decisions pair with their recorded outcomes to form a dataset of (D, y) tuples, it reaches the data-driven leaf: predict-then-optimise or a learned policy trained on a full year of observed routing decisions. Two data investments would move UC-A to the stronger model-based leaf: populating per-leg distance (recoverable from the network edges) and node coordinates, after which the measured emission could also serve as the validation signal $\hat{y}$. This is the clearest illustration of the guidance value of the pipelines: two use cases that appear equivalent at the level of "emissions can be computed" are separated by a single predicate—model availability—into a learned-policy recommendation and a model-optimisation recommendation.

Use case B. The analytical model $y = g(D)$ is available: the fleet table supplies per-transport-type emission, fuel and cost factors and the per-leg distance, weight and transport mode are populated, so the impact of any routing decision can be computed. The model can also be partially validated, because metered fuel and energy consumption is populated and can be checked against the model prediction $(\hat{y})$; this checks the energy component

of the model rather than the CO_2-equivalent directly, and Table 5 records the $\hat{y}$ predicate as partially satisfied. UC-B therefore reaches the model-validated leaf: model-based optimisation—integer programming, heuristics or metaheuristics—minimising emissions under the capacity constraints of the fleet table and the network topology. The routing decisions paired with their modelled outcomes also form a dataset of (D, y) tuples—7500 legs, well above the 30-observation working minimum—so the data-driven branch (predict-then-optimise or a learned policy) is jointly feasible; it is reported as a lower-priority alternative rather than the recommendation, for two reasons: under the priority ordering of Section 3 the calibrated model-based leaf ranks first, and the tuple outcomes are themselves model-derived, so a learned policy would only re-encode the emission model that can already be optimised directly.

Three prescriptive extensions remain out of reach with the current fields, and each is tied to a specific missing field. The commercial-assignment fields are empty, so the carrier cannot be chosen, container loading cannot be optimised, and customer-level service constraints cannot be imposed. The absence of an order date removes any time-window or delivery-scheduling constraint, and the empty edge travel times prevent time- and speed-constrained routing. Finally, no text corpus exists, so the LLM-based qualitative-reasoning branch used when neither a model nor historical data are available cannot be invoked.

Table 6 summarises the applicability of methods to UC-A and UC-B.

Table 6. Summary of feasible methods per use case. • = applicable; ◦ = not applicable. The last column states why the method is usable, its limiting condition, or the field that would be required to enable it.

Method (Decision-Tree Leaf)	UC-A	UC-B	Why Usable, or What Is Missing
Descriptive			
Inferential statistics	•	•	A: recorded outcome from a single source; B: outcome derived from the model
Correlation/clustering (X)	•	•	A: mode, route, product and cost (distance and load absent); B: distance, load and network geometry
Bayesian data fusion	◦	◦	Missing: repeated measurements of the same quantity
Simulation from $y = f(X)$	◦	•	A: outcome recorded, model not needed; B: model from fleet factors and distance
LLM text analysis	◦	◦	Missing: unstructured text corpus
Predictive			
Univariate time series (ARIMA/ETS)	•	◦	A: 365 daily points; B: outcome not recorded and 25 days too short for seasonal structure
Supervised learning on (X, y)	•	◦	A: covariates populated; B: outcome not recorded; 7500 legs suffice in number, but 25 days limit temporal generalisation
LLM zero-shot forecasting (RAG)	◦	◦	Missing: text corpus or textual knowledge base
Prescriptive			
Model-based optimisation (validated g)	◦	•	B: $g(D)$ from distance and fleet factors, validated on metered fuel (energy component only); A: per-leg distance missing
Simulation-based (unvalidated model)	◦	◦	Not needed: B has a validated model, A has no evaluable model
Data-driven policy/predict-then-optimise	•	◦	A: 365 days of recorded (D, y); B: tuples are model-derived and rank below the validated model leaf
Carrier/container/customer optimisation	◦	◦	Missing: commercial-assignment fields are empty
Time- and speed-constrained routing	◦	◦	Missing: order date and edge travel times
LLM qualitative reasoning	◦	◦	Missing: text corpus

Taken together, the application shows how the pipelines turn a data-availability profile into a method recommendation, and why the same physical system can route to different feasible method sets. Three patterns emerge. First, the pipelines expose data sufficiency, not merely data presence: UC-B routes to the model-based descriptive and prescriptive leaves because it holds the drivers, whereas UC-A routes to the outcome-based descriptive and data-driven prescriptive leaves because it holds the recorded outcome but not the model inputs. Second, every unavailable branch is paired with an explicit route to availability: the text-based branches would open with a document corpus, the Bayesian branch with repeated measurements, the model-based prescriptive branch for UC-A with per-leg distance and node coordinates, and the richer predictive leaves for UC-B with a longer or a directly metered history. Third, the counting rules keep observation count and temporal coverage separate: UC-B's 7500 legs exceed the 30-observation working minimum, so its limitation is not sample size but temporal depth—25 days cannot support seasonal structure—whereas UC-A's 365 days support both seasonal time-series and supervised training (Section 3.5). The section is thus an executability and discrimination check rather than an outcome-based validation: it does not establish that the recommended methods are optimal, only that the data conditions they rely on are present, mutually discriminating and, where absent, clearly identifiable—together with the specific data whose collection would move a use case to a better-supported leaf.

5. Discussion

5.1. Managerial Implications

The structured expert-rule pipelines carry concrete implications for production managers, supply chain planners, and manufacturing analysts.

First, the pipelines expose a precise mapping from data-availability states to feasible method families, making explicit the informational prerequisites of each analytical approach. For practitioners, this identifies the minimum data investment required to access a given class of methods: organisations that have been systematically collecting operational data are structurally better positioned to deploy data-driven prediction methods whose adoption is growing fastest [8].

Second, the expert-rule pipelines reduce the dependency on data science expertise for method selection: the binary predicates at each pipeline node (Is y measurable? Are auxiliary variables X available?) are expressed in operational terms that a trained operations manager can assess, although their evaluation still requires judgement on data quality and on accounting boundaries (Section 3.5).

Third, the incorporation of LLM-based pathways into all three pipelines extends the framework's applicability to data-sparse and text-rich environments. Many production facilities maintain extensive documentation—maintenance logs, supplier correspondence, incident reports—that are underutilised for analytical purposes. The pipelines now offer a structured route to extract value from these unstructured assets through LLM-based text analysis, zero-shot forecasting, and case-history reasoning. This is particularly relevant for sustainability reporting, where unstructured data sources such as supplier audits, environmental incident reports, and community feedback can be analysed through LLM-based pathways to complement structured ESG metrics [28].

Fourth, accurate demand forecasting enabled by the predictive pipeline has direct cost implications: it reduces safety stock, lowers holding costs, and improves production scheduling—outcomes that the literature associates with cost efficiency, although the present study does not quantify them [44]. When extended to sustainability objectives, the same predictive capabilities support energy demand forecasting, waste generation prediction, and reverse logistics planning [30].

5.2. Organisational Roles

Table 7 illustrates the organisational roles involved in executing the pipelines in practice. The Data Owner defines the semantics of the available data, the Data Engineer assesses completeness and accuracy, and the Data Scientist selects and applies the recommended method, with the Process Owner validating the outcome. The IT Department provides and maintains the data infrastructure, while the decision pipelines (as a procedural tool) guide the method-selection process itself. Read row-wise, the table shows that no single role owns the whole selection process: data ownership sits with the business, data engineering with IT, and method choice with the data scientist, so the pipeline acts as a coordination device rather than a substitute for any role. Read column-wise, it shows that the two columns that determine a method family—data boundary (Data Owner) and data completeness (Data Engineer)—are upstream of the method choice (Data Scientist), which is consistent with the data-conditioned logic of the trees. In small organisations without a dedicated data scientist, the Process Owner and Data Owner jointly assume the method-selection column, which is where the usability of the pipelines matters most.

Table 7. Application of a decision pipeline within an enterprise context. ● = role holds this responsibility as primary; ◦ = role does not hold this responsibility directly.

Task	Provide Data Defining Its Boundary and Conditions	Verify the Completeness and Accuracy of Data	Choose a Solving Method	Make Decision
Owner	Data Owner	Data Engineer	Data Scientist	Process Owner
Business Department	●	◦	●	●
IT Department	◦	●	●	◦
Decision pipelines	◦	●	●	◦

5.3. Relevance for Sustainable Supply Chain Management

The decision pipelines have direct relevance for sustainability-oriented applications because sustainability problems often involve distinctive data-availability conditions that the pipelines are designed to handle. This subsection analyses how each pipeline changes sustainability decision-making in practice.

ESG metrics as measurable outcomes under the GHG Protocol. In many sustainability applications, the outcome variable y represents an ESG metric—e.g., carbon emissions, water consumption, or waste generation—that is measurable from sensors, utility bills, or reporting databases. Under the GHG Protocol and ISO 14064, Scope 1 (direct) and Scope 2 (energy indirect) emissions are typically quantifiable from metering and invoicing data, triggering the measurable branch of the descriptive pipeline. However, Scope 3 emissions—which can represent 70–90% of total emissions [45]—depend on supplier-reported data, spend-based estimation methods, and emission factors from secondary databases, creating heterogeneous data quality across emission categories. This heterogeneity across categories is a data-availability condition the descriptive pipeline must accommodate; where independent measurements of the same quantity exist (for example, two metering sources for Scope 1 and 2), the Bayesian data-fusion branch (Node 2) applies a Kalman filter or hierarchical Bayesian model to produce a fused estimate, whereas spend-based Scope 3 estimates are integrated with directly measured emissions at the inventory level through boundary-aware models rather than fused pointwise [28].

CSRD/ESRS forward-looking disclosure requirements. The Corporate Sustainability Reporting Directive (CSRD) and European Sustainability Reporting Standards (ESRS) require firms to disclose forward-looking sustainability targets and transition plans. This creates demand for predictive analytics on ESG metrics—e.g., forecasting emissions tra-

jectories under different decarbonisation scenarios, or predicting resource consumption under circular economy interventions. When historical ESG data series are available (e.g., from mandatory emissions reporting), the predictive pipeline applies directly. However, for emerging impact categories (biodiversity, water stress, social metrics) where historical data are short or non-existent, the pipeline directs practitioners toward LLM-based zero-shot forecasting via knowledge graphs—a pathway that is particularly relevant for CSRD compliance where forward-looking disclosures must be made even when historical data are limited.

LCA data conditions under ISO 14040/44. Lifecycle assessment (LCA) requires detailed material and energy flow data across the product lifecycle. When such data are available as structured records (e.g., from enterprise resource planning systems), they function as auxiliary variables X in the predictive pipeline, enabling environmental impact forecasting as a function of production parameters. When LCA data exist only in textual form—e.g., environmental product declarations (EPDs), sustainability reports, or supplier environmental audits—the LLM-based extraction pathway in the descriptive pipeline becomes relevant, extracting quantitative indicators from unstructured text [29].

Circularity constraints in prescriptive problems. In circular economy applications, the prescriptive pipeline's constraint set must include circularity constraints mandated by extended producer responsibility (EPR) regulations: minimum recycled content, maximum landfill disposal, and take-back compliance targets. When an analytical model $y = g(D)$ is available—e.g., a material flow model relating disassembly decisions to recovery rates—traditional mathematical programming applies. When no formal model exists, LLM-based agents can reason over textual descriptions of circular supply chain configurations to propose candidate solutions [30,32].

Digital twin-enabled decarbonisation. When a validated digital twin of a supply chain is available, the prescriptive pipeline's analytical-model branch enables scenario analysis for carbon footprint reduction. The digital twin functions as the analytical model $y = g(D)$, where y is the carbon footprint and D represents routing, sourcing, or production decisions. This enables green facility location, energy-efficient routing, and low-carbon procurement without requiring extensive historical data. The pipeline explicitly conditions on whether the digital twin is validated against empirical data (the $\hat{y}$ predicate): unvalidated twins yield no reliable prescription, directing practitioners toward data collection or simulation-based approaches instead [33,34].

5.4. Reading the Pipelines in Sustainability Contexts

This subsection reads the general pipelines of Section 3 through three sustainability-specific conditions—emission-accounting scope, accounting boundary, and assurance requirement. The three conditions do not modify the structure of the trees: they form a reading layer that adjusts how the data-availability predicates of Section 3.5 are evaluated and which of the jointly feasible leaves is acceptable.

First, emission-accounting scope: the practitioner records whether y refers to Scope 1, Scope 2, or Scope 3 emissions, because the three scopes differ in measurability, boundary, and data provenance. Scope 1 and Scope 2 typically satisfy the measurability predicate directly. Scope 3, which depends on supplier-reported data and spend-based factors, usually does not, and routes instead to the model/derivation branch—emission factors applied to activity data—or, where supplier documentation exists only in textual form, to the text-based branch; it is not routed to the heterogeneous-quality fusion branch, because spend-based estimates and metered values are not repeated measurements of one quantity.

Second, accounting boundary: the reading distinguishes entity-level corporate inventories from product-level life-cycle inventories. The same numeric quantity cannot be

combined across boundaries, and the Kalman-fusion branch of the descriptive tree is valid only for repeated measurements of a single quantity within one boundary—for example, two metering sources for the same Scope 1 flow.

Third, assurance requirement: where third-party verification or CSRD auditability is mandated, the practitioner downgrades opaque or exploratory leaves—in particular the LLM-based fallbacks—even when their data condition is satisfied, and prefers methods that produce traceable, documented estimates.

Applied this way, the three conditions change the evaluation of the predicates and the acceptability of the leaves for a given dataset without changing the trees, which is the increment a sustainability readership should expect from a domain-independent framework. Extending the pipelines themselves with these conditions as explicit nodes, and validating the resulting choices on real sustainable supply chain cases, remains future work (Section 5.5).

5.5. Limitations and Future Research

Several limitations of the present work warrant acknowledgement.

First, the pipelines are expert-designed but not algorithmically derived from data. The distinction between expert-designed and data-driven pipelines is acknowledged as a limitation and a direction for future algorithmic formalisation.

Second, the method prescriptions are necessarily coarse-grained. Finer-grained classifications (e.g., distinguishing stochastic from deterministic optimisation, or deep learning from classical machine learning) may reveal richer patterns but also increase classification ambiguity. The sensitivity of the pipeline prescriptions to classification granularity is an open research question.

Third, the pipelines assume that data-availability conditions are static at the time of method selection. In practice, data availability may evolve as data collection infrastructure matures, potentially shifting the feasible set of methods over time. Dynamic extensions of the pipelines that account for temporal data maturation are a natural direction for future work.

Fourth, the pipelines have been developed in the SCS domain, and the application examples demonstrate executability rather than soundness. A retrospective application to an operational data warehouse (Section 4) shows that the data conditions the pipelines rely on are present and discriminative in real records and that a shared structure routes to different method families depending on data depth and information content; however, an outcome-based validation—an execution record with measured results, practitioner feedback (e.g., a Delphi round), or a comparison against expert method choices—is not yet reported and remains the most immediate direction for future work. Beyond that, the pipelines should be verified in application domains beyond SCS, such as healthcare logistics or humanitarian supply chains.

Fifth, the LLM-based pathways introduced in this paper rely on the availability of domain-specific knowledge graphs $\mathcal{K}$ and unstructured text corpora $\mathcal{T}$. The quality of LLM outputs is contingent on the completeness and accuracy of these knowledge sources, and systematic assessment of LLM reliability in supply chain decision-making remains an open research question [11]. Future work should establish validation protocols for LLM-based method prescriptions.

Sixth, the integration of causal AI with digital twin technologies [12,13] creates additional prescriptive pathways that bypass traditional historical-data requirements. The pipelines should be further extended to account for digital-twin maturity as an explicit conditioning variable, enabling practitioners to select methods based on simulation fidelity rather than solely on empirical observations.

6. Conclusions

This paper introduces three structured expert-rule pipelines—descriptive, predictive, and prescriptive—for analytics method selection in Supply Chain Systems. The pipelines translate observable data-availability conditions into recommended method families through binary decision trees.

The descriptive pipeline guides practitioners from the measurability of a target variable to the appropriate analytical or simulation methods, and extends the reach to unstructured text corpora through LLM-based text analysis when structured data are unavailable. The predictive pipeline conditions on the availability of auxiliary variables to recommend time-series or supervised machine-learning approaches, and incorporates LLM-based zero-shot forecasting via retrieval-augmented generation for data-sparse settings. The prescriptive pipeline maps the presence or absence of validated models and historical data to optimisation or simulation-based methods, and introduces LLM-based qualitative reasoning from case histories as an alternative solver when the objective function cannot be formalised.

Each pipeline is designed to be usable by operations managers with limited statistical training, reducing the dependency on specialised data science expertise for method selection; the predicates, however, still require expert judgement, and the resulting prescriptions should be validated before deployment. The application examples demonstrate how the framework translates concrete data states into method recommendations across production and logistics contexts, including scenarios where traditional data-intensive methods are infeasible and LLM-based pathways provide a viable alternative.

The contributions of this work are threefold. First, the three decision pipelines constitute an expert-rule-based selection tool conditioned on observable data-availability predicates. Second, the framework bridges the gap between problem taxonomy and practical method deployment, offering a structured basis for method selection in production economics. Third, the integration of generative AI and LLM-based method families into the pipelines extends their applicability to text-rich, data-sparse environments that are common in industrial practice.

The framework has direct relevance for sustainable supply chain management. By mapping data conditions to feasible analytical methods, the pipelines enable practitioners to identify the appropriate tools for circular economy analytics, ESG reporting, and digital twin-enabled decarbonisation. As sustainability reporting requirements intensify and circular economy initiatives expand, the ability to systematically select analytical methods based on data availability becomes a strategic capability for organisations seeking to balance economic performance with environmental and social responsibility.

Future work should incorporate digital-twin maturity as an additional branching condition in the prescriptive pipeline, establish validation protocols for LLM-based method prescriptions, extend the pipelines to finer-grained method classifications, validate the framework in sustainability-specific application domains such as circular supply chains and humanitarian logistics, and explore multi-objective extensions that simultaneously address economic, environmental, and social dimensions of supply chain performance.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The dataset analysed in this study and described in Section 4 is not publicly available. The data structures, field semantics, and field-by-field incidence of the two use cases relevant for the pipeline conditions are reported in Table 4.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Kache, F.; Seuring, S. Challenges and opportunities of digital information at the intersection of Big Data Analytics and supply chain management. *Int. J. Oper. Prod. Manag.* **2017**, *37*, 10–36. [CrossRef]
2. Leng, J.; Sha, W.; Wang, B.; Zheng, P.; Zhuang, C.; Liu, Q.; Wuest, T.; Mourtzis, D.; Wang, L. Industry 5.0: Prospect and retrospect. *J. Manuf. Syst.* **2022**, *65*, 279–295. [CrossRef]
3. Wang, G.; Gunasekaran, A.; Ngai, E.W.; Papadopoulos, T. Big data analytics in logistics and supply chain management: Certain investigations for research and applications. *Int. J. Prod. Econ.* **2016**, *176*, 98–110. [CrossRef]
4. Mentzer, J.T. Rigor Versus Relevance: Why wuold we choose only one? *J. Supply Chain. Manag.* **2008**, *44*, 72–77. [CrossRef]
5. Jahani, H.; Jain, R.; Ivanov, D. Data science and big data analytics: A systematic review of methodologies used in the supply chain and logistics research. *Ann. Oper. Res.* **2023**, *359*, 1297–1354. [CrossRef]
6. Lustig, I.; Dietrich, B.; Johnson, C.; Dziekan, C. The analytics journey. *Analytics Magazine*, 11–13 December 2010. [CrossRef] [PubMed]
7. Souza, G.C. Supply chain analytics. *Bus. Horiz.* **2014**, *57*, 595–605. [CrossRef]
8. Dubey, R.; Gunasekaran, A.; Childe, S.J.; Bryde, D.J.; Giannakis, M.; Foropon, C.; Roubaud, D.; Hazen, B.T. Big data analytics and artificial intelligence pathway to operational performance under the effects of entrepreneurial orientation and environmental dynamism: A study of manufacturing organisations. *Int. J. Prod. Econ.* **2020**, *226*, 107599. [CrossRef]
9. Stentoft, J.; Rajkumar, C. Balancing theoretical and practical relevance in supply chain management research. *Int. J. Phys. Distrib. Logist. Manag.* **2018**, *48*, 504–523. [CrossRef]
10. Bahroun, Z.; Saihi, A.; As'ad, R.; Tanash, M. A systematic analysis of generative artificial intelligence for supply chain transformation. *Supply Chain. Anal.* **2026**, *13*, 100188. [CrossRef]
11. Wang, H.; Jiang, J.; Hong, L.J.; Jiang, G. LLMs for Supply Chain Management. *arXiv* **2025**, arXiv:2505.18597. [CrossRef]
12. Gottimukkala, S.R.; Bhuram, S.K. Prescriptive Analytics for Next-Gen Supply Chains: Integrating Causal AI with Digital Twin Technologies. In Proceedings of the 2025 IEEE International Conference on Pattern Recognition, Machine Vision and Artificial Intelligence (PRMVAI), Loudi, China, 20–22 June 2025; pp. 1–6.
13. Soykan, B.; Rabadi, G. Integrating AI and ML in Supply Chain Digital Twins: Bridging Potential and Foundational Research Gaps. In *Optimizing Supply Chains Through Digital Twins: Navigating the Modern Landscape*; Rabadi, G., Soykan, B., Eds.; Springer Nature: Cham, Switzerland, 2025; pp. 243–265. [CrossRef]
14. Smyth, C.; Dennehy, D.; Wamba, S.F.; Scott, M.; Harfouche, A. Artificial intelligence and prescriptive analytics for supply chain resilience: A systematic literature review and research agenda. *Int. J. Prod. Res.* **2024**, *62*, 8537–8561. [CrossRef]
15. Ivanov, D. What network and performance indicators can tell us about supply chain and sourcing resilience (and what they cannot). *J. Purch. Supply Manag.* **2025**, *31*, 101014. [CrossRef]
16. *ISO 14064-1:2018*; Greenhouse Gases—Part 1: Specification with Guidance at the Organization Level for Quantification and Reporting of Greenhouse Gas Emissions and Removals. International Organization for Standardization: Geneva, Switzerland, 2018.
17. *ISO 14040:2006*; Environmental Management—Life Cycle Assessment—Principles and Framework. International Organization for Standardization: Geneva, Switzerland, 2006.
18. Anthony, R. *Planning and Control Systems: A Framework for Analysis*; Harvard Business School Publications, Division of Research, Graduate School of Business Administration, Harvard University: Cambridge, MA, USA, 1965.
19. Schmidt, G.; Wilhelm, W.E. Strategic, tactical and operational decisions in multi-national logistics networks: A review and discussion of modelling issues. *Int. J. Prod. Res.* **2000**, *38*, 1501–1523. [CrossRef]
20. Delen, D. *Predictive Analytics: Data Mining, Machine Learning and Data Science for Practitioners*, 2nd ed.; FT Press: New York, NY, USA, 2020.
21. Lambert, D.M.; Enz, M.G. We Must Find the Courage to Change. *J. Bus. Logist.* **2015**, *36*, 9–17. [CrossRef]
22. Bertsimas, D.; Kallus, N. From Predictive to Prescriptive Analytics. *arXiv* **2018**, arXiv:1402.5481. [CrossRef]
23. Elmachtoub, A.N.; Grigas, P. Smart "predict, then optimize". *Manag. Sci.* **2022**, *68*, 9–26. [CrossRef]
24. Provost, F.; Fawcett, T. Data Science and its Relationship to Big Data and Data-Driven Decision Making. *Big Data* **2013**, *1*, 51–59. [CrossRef] [PubMed]
25. Chen, H.; Chiang, R.H.L.; Storey, V.C. Business Intelligence and Analytics: From Big Data to Big Impact. *Manag. Inf. Syst. Q.* **2012**, *36*, 1165–1188. [CrossRef]
26. Trkman, P.; McCormack, K.; de Oliveira, M.P.V.; Ladeira, M.B. The impact of business analytics on supply chain performance. *Decis. Support Syst.* **2010**, *49*, 318–327. [CrossRef]
27. Hofmann, E. Big data and supply chain decisions: The impact of volume, variety and velocity properties on the bullwhip effect. *Int. J. Prod. Res.* **2017**, *55*, 5108–5126. [CrossRef]
28. Mustafa, F.; Smolarski, J.; Elamer, A. The Convergence of Artificial Intelligence and Sustainability Reporting: A Systematic Review of Applications, Challenges and Future Directions. *Bus. Strategy Environ.* **2025**, *34*, 9761–9784. [CrossRef]

29. Strazzullo, S.; Acampora, L.; Cricelli, L.; Ianniello, S.; Nenni, M.; Singh, N. Leveraging Artificial Intelligence for ESG Reporting: A Case Study in the European Fashion Industry. *Bus. Strategy Environ.* **2026**, *35*, 4521–4537. [CrossRef]
30. Kokkinou, A.; Mitas, O. Driving the circular economy: How artificial intelligence could empower transportation and logistics companies to lead the transition. *Clean. Logist. Supply Chain* **2026**, *18*, 100300. [CrossRef]
31. Farshadfar, Z.; Mucha, T.; Tanskanen, K. Leveraging Machine Learning for Advancing Circular Supply Chains: A Systematic Literature Review. *Logistics* **2024**, *8*, 108. [CrossRef]
32. Kumar, L.; Khedlekar, U.K. An analytics-driven circular supply chain framework integrating quality, warranty, and human efficiency. *Supply Chain. Anal.* **2025**, *11*, 100140. [CrossRef]
33. Phonthanukitithaworn, C.; Ketkaew, C. Managing green supply chain performance through digital twin technology—Insights from SMEs and large firms. *J. Open Innov. Technol. Mark. Complex.* **2026**, *12*, 100762. [CrossRef]
34. Arij, L.; Areej, S. Digital Twins for Decarbonized Supply Chain: A Conceptual Framework. *IFAC-PapersOnLine* **2025**, *59*, 1624–1629. [CrossRef]
35. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
36. Rice, J.R. The Algorithm Selection Problem. In *Advances in Computers*; Elsevier: Amsterdam, The Netherlands, 1976; Volume 15, pp. 65–118. [CrossRef]
37. Kerschke, P.; Hoos, H.H.; Neumann, F.; Trautmann, H. Automated Algorithm Selection: Survey and Perspectives. *Evol. Comput.* **2019**, *27*, 3–45. [CrossRef] [PubMed]
38. Petropoulos, F.; Apiletti, D.; Assimakopoulos, V.; Babai, M.Z.; Barrow, D.K.; Ben Taieb, S.; Bergmeir, C.; Bessa, R.J.; Bijak, J.; Boylan, J.E.; et al. Forecasting: Theory and practice. *Int. J. Forecast.* **2022**, *38*, 705–871. [CrossRef]
39. Jannelli, V.; Schoepf, S.; Bickel, M.; Netland, T.; Brintrup, A. Agentic LLMs in the Supply Chain: Towards Autonomous Multi-Agent Consensus-Seeking. *arXiv* **2024**, arXiv:2411.10184. [CrossRef]
40. Akbari-Aghghaleh, Z.; Mozdgir, A.; Seyedi, I.; Messina, E. Designing a perishable closed-loop poultry supply chain: Metaheuristic approaches and model evaluation. *Environ. Dev. Sustain.* **2025**, 1–43. [CrossRef]
41. Madani, B.; Saihi, A.; Abdelfatah, A. A Systematic Review of Sustainable Supply Chain Network Design: Optimization Approaches and Research Trends. *Sustainability* **2024**, *16*, 3226. [CrossRef]
42. Jayarathna, C.P.; Agdas, D.; Dawes, L.; Yigitcanlar, T. Multi-Objective Optimization for Sustainable Supply Chain and Logistics: A Review. *Sustainability* **2021**, *13*, 3617. [CrossRef]
43. Gholian-Jouybari, F.; Hajiaghaei-Keshteli, M.; Smith, N.R.; Calvo, E.Z.R.; Mejía-Argueta, C.; Mosallanezhad, B. An in-depth metaheuristic approach to design a sustainable closed-loop agri-food supply chain network. *Appl. Soft Comput.* **2024**, *150*, 111017. [CrossRef]
44. Ivanov, D. Intelligent digital twin (iDT) for supply chain stress-testing, resilience, and viability. *Int. J. Prod. Econ.* **2023**, *263*, 108938. [CrossRef]
45. Carbon Trust. Scope 3 Emissions: What Are They and Why Do They Matter? 2023. Available online: https://www.carbontrust.com/our-work-and-impact/guides-reports-and-tools/scope-3-emissions-what-are-they-and-why-do-they-matter (accessed on 16 September 2026).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.