EURO Journal on Computational Optimization 14 (2026) 100132

Contents lists available at ScienceDirect

EURO Journal on Computational Optimization

journal homepage: www.elsevier.com/locate/ejco

Data-driven stochastic VRP: Integration of forecast duration into optimization for utility workforce management

Matteo Garbelli

Department of Computer Science, Strada le Grazie, 15, Verona, 37134, Italy

HIGHLIGHTS

- Dual-weighted XGBoost achieves 5.06 min MAE on 884K gas meter interventions.
- Sub-Gaussian risk buffers enable tractable stochastic VRP with ML forecasts.
- Multi-objective NSGA-III integrates uncertainty into route optimization.
- Forecast-driven planning improves operator utilization by 20–25%.
- Empirical validation confirms sub-Gaussian assumption on prediction residuals.

ARTICLE INFO

Keywords:
Stochastic VRP
Machine learning
XGBoost
Sub-Gaussian concentration
Multi-objective optimization
Evolutionary algorithms

ABSTRACT

This paper investigates the integration of machine learning forecasts of intervention durations into a stochastic variant of the Capacitated Vehicle Routing Problem with Time Windows (CVRPTW). In particular, we exploit tree-based gradient boosting (XGBoost) trained on eight years of gas meter maintenance data to produce point predictions and uncertainty estimates, which then drive a multi-objective evolutionary optimization routine. The methodology addresses uncertainty through sub-Gaussian concentration bounds for route-level risk buffers and explicitly accounts for competing operational KPIs through a multi-objective formulation. Empirical analysis of prediction residuals validates the sub-Gaussian assumption underlying the risk model. From an empirical point of view, our results report improvements of around 20–25% in operator utilization and completion rates compared with plans computed using default durations and unconditional stochastic baselines. The integration of uncertainty quantification and risk-aware optimization provides a practical framework for handling stochastic service durations in real-world routing applications.

1. Introduction

The Vehicle Routing Problem (VRP), first formalized in [7], addresses the efficient assignment of vehicles to serve a predetermined set of customers or locations and constitutes a fundamental optimization challenge in logistics, transportation, and utility services. Classical VRP methods assume complete information on travel times or fixed service durations and deterministic customer demands. While being mathematically tractable, this deterministic perspective poorly represents real operations and solutions often fail when deployed, becoming infeasible or highly suboptimal. In practice, traffic conditions fluctuate, service durations vary across different contexts, and customer demands change.

Email address: matteo.garbelli@univr.it.

https://doi.org/10.1016/j.ejco.2026.100132

Available online 19 June 2026
2192-4406/© 2026 The Author. Published by Elsevier B.V. on behalf of Association of European Operational Research Societies (EURO). This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

Abbreviation

AI Artificial Intelligence
BF Back-and-Forth
CDF Cumulative Distribution Function
CI Confidence Interval
CSO Contextual Stochastic Optimization
CVRP Capacitated Vehicle Routing Problem
CVRPSD Capacitated VRP with Stochastic Demand
DoE Design of Experiments
DRL Deep Reinforcement Learning
DRSO Distributionally Robust Stochastic Optimization
EA Evolutionary Algorithm
EI Expected Improvement
GES Grouping Evolution Strategy
GNN Graph Neural Network
GPS Global Positioning System
GSO Graph Shift Operators
KPI Key Performance Indicator
MAE Mean Absolute Error
MAPE Mean Absolute Percentage Error
ML Machine Learning
MOEA Multi-Objective Evolutionary Algorithm
MSH Multi-Space Sampling Heuristic
NSGA-III Non-dominated Sorting Genetic Algorithm III
OR Operations Research
OX Order Crossover
PDF Probability Density Function
PR Preventive Restocking
RMSE Root Mean Squared Error
RRN Recurrent Relational Network
SCO Stochastic Contextual Optimization
SDI Servizio Distribuzione Impianti (Gas Meter Service)
SMS-EMOA S-Metric Selection Evolutionary Multi-objective Algorithm
SPO Smart Predict-then-Optimize
SVRP Stochastic Vehicle Routing Problem
TWCVRP Time Window Capacitated Vehicle Routing Problem
UCC (Benchmark) Unconditional Chance-Constrained
VRP Vehicle Routing Problem
VRPB VRP with Backhauling
VRPPD VRP with Pickups and Deliveries

The gap between practice and theory has driven the development of a new class of problems, specifically the Stochastic VRPs, which explicitly model uncertainty. Recently, the inclusion of data-driven frameworks, such as predictive algorithms, has improved the performance and refined the efficiency of the route-generation process for optimization tasks.

This work addresses the challenge of incorporating stochastic intervention durations into predictive models for VRP optimization. We evaluate whether machine learning forecasts improve routing solutions compared to traditional fixed-time estimates used in practice.

Our study has three objectives: (1) develop a framework integrating ML forecasts into SVRP solvers; (2) evaluate multiple prediction models trained on eight years of operational data; (3) identify which forecast-optimization combinations perform best for practical deployment.

Organization of the paper. The remainder of this paper is organized as follows. Section 2 provides an overview of Stochastic VRP and its variants with a glance toward optimization under uncertainty. Section 3 presents the chosen optimization algorithm to solve VRPs. Section 4 discusses predictive methods for informed optimization. Section 5 presents the experimental framework developed for analyzing the performance of the forecast with respect to the actual baseline used for production. Section 6 focuses on producing representative simulations for multiple instances of VRPs, showing the potential and the efficiency of forecast integration. Finally, Section 7 concludes the paper and outlines directions for future research.

2. Literature review

This section reviews best practices for stochastic VRPs, analyzing both classical approaches and the integration of machine learning methods. Traditionally, two principal methodological frameworks have emerged for effectively incorporating travel time uncertainty into VRP models with time windows: stochastic programming and robust optimization. The stochastic programming framework operates under the assumption that complete probability distributions of travel times are available. This methodology explicitly incorporates these distributions into the optimization model, typically through scenario-based formulations or chance constraints. Under this paradigm, travel times are modeled as random variables drawn from predetermined probability distributions, allowing for explicit consideration of uncertainty in the decision-making process. Classical surveys, such as [24,27], offer a comprehensive overview of this approach.

Conversely, the robust optimization approach does not require the knowledge of specific probability distributions; rather, it is built upon the construction of uncertainty sets. This methodology characterizes travel time variability through bounded intervals rather than complete distributional information. The objective is to develop solutions that maintain feasibility and near-optimal performance across all possible realizations within the specified uncertainty set. We refer to [8] for a complete survey on this strand. We also mention another complementary approach, dealing with contextual optimization [6], that uses local context variables (e.g., traffic information over some edges) to condition routing decisions.

The remainder of the section presents a thorough review of existing empirical approaches, mostly building on heuristic methods to model stochastic demand, time-dependent travel uncertainties, and multi-objective optimization, establishing the current state-of-the-art in SVRP research.

The comprehensive analysis by Oyola et al. [19,20] demonstrates that uncertainty in VRP manifests across multiple operational dimensions, primarily affecting demand patterns, travel times, and service durations. The evolution of the deterministic capacitated VRP has generated multiple specialized variants designed to address specific operational constraints and real-world complexities. These variants include:

1. the Capacitated Vehicle Routing Problem with Stochastic Demand (CVRPSD);
2. the Vehicle Routing Problem with Time Windows (VRPTW) incorporating temporal constraints for customer service;
3. the VRP with Pickups and Deliveries (VRPPD) addressing simultaneous collection and distribution requirements;
4. the VRP with Backhauling (VRPB), optimizing routes that combine delivery and subsequent collection operations.

Among these variants, the CVRPSD represents the most extensively investigated stochastic formulation, characterized by customer demands that follow probabilistic distributions and are revealed only after the initial route planning is completed.

The treatment of correlated stochastic travel times has received significant attention through the work of Rajabi-Bahaabadi et al. [1], who address the reliable vehicle routing problem in stochastic networks with correlated travel times. A genetic evolutionary algorithm is proposed for modeling stochastic travel times with soft time windows and correlated stochastic arc travel times, which is simultaneously coupled with an ant colony optimization algorithm for solving the resulting stochastic programming problem. The methodology derives closed-form expressions for calculating expected earliness and tardiness penalties while applying the Akaike Information Criterion to identify optimal probability distributions for travel time modeling. Shifted log-normal distributions demonstrate superior performance.

In the paradigm of contextual optimization, the VRPTW variant has been studied in the work presented in [25]. Their work investigates VRPTW under stochastic travel times where decision-makers observe contextual information represented as feature variables prior to making routing decisions. The approach integrates contextual feature variables directly into the optimization algorithm, minimizing total transportation costs and expected late arrival penalties conditioned on observed features. This methodology differs from previous approaches, e.g., [1,4], since a fixed known travel time distribution is assumed. Instead, [25] addresses scenarios where the joint distribution of travel times and features remains unknown, developing novel data-driven prescriptive models that utilize historical data to provide approximate solutions. Their data-driven prescriptive model connects to decision-makers with access to historical travel times for recent periods, enabling direct penalty prediction rather than requiring estimation of travel time distributions followed by expected penalty approximation.

Feasibility assessment for stochastic VRPTW has been advanced in [4], where efficient feasibility assessment methods for stochastic VRPTW with correlated and time-dependent travel times are developed. Their contributions include an approximate feasibility verification approach for single-chance constrained routing problems that concurrently considers travel time correlations and time dependencies. The methodology adopts a jointly normal distribution assumption for travel time modeling and establishes infeasibility criteria based on excessive time window violation probabilities.

For an environmental perspective, we refer to [12], who extend stochastic formulations to include environmental factors through the Green VRP. Their approach implements flexible time windows to enhance both routing efficiency and environmental performance, develops a preference-based system allowing customers to rank alternative time windows, and formulates a multi-objective optimization model that balances operational costs, fossil fuel consumption, and customer satisfaction. The research creates realistic benchmark problems for performance evaluation and conducts comparative analysis of four evolutionary algorithm solvers (NSGA-II, NSGA-III, MOEA/D, and SMS-EMOA) for Pareto front approximation.

In practice, however, neither the stochastic programming nor the robust optimization paradigm provides a straightforward, scalable deployment for production systems. On the one hand, the stochastic approach demands high-fidelity probability distributions [6] that are rarely available in real time. Conversely, the robust approach can become overly conservative when uncertainty sets are widened to cover edge cases, leading to infeasible routes or excessive buffer times.

To circumvent these limitations, a new class of data-driven and machine-learning approaches has been developed for directly solving or integrating the VRP. As clearly illustrated in [2,3,26], recent efforts have explored how learning paradigms—ranging from supervised and reinforcement learning to hybrid neuro-heuristic frameworks—can replace or augment traditional optimization pipelines for vehicle routing problems. [3] provides a detailed classification that differentiates between end-to-end learning, hybrid learning combined with heuristics, single-agent and multi-agent formulations. It shows that ML algorithms and architectures can produce near-optimal solutions while operating significantly faster. [26] extends this picture with an application-oriented survey, showing that attention-based networks and graph neural encoders now outperform classical meta-heuristics on large-scale, dynamic VRP variants: modern ML models can reliably capture stochastic travel times, forecast customer demand, learn adaptive routing policies, and plug directly into optimizers. Also [2] offers a practitioner-oriented review that maps out how analytics and ML can be woven into every stage of a VRP workflow: stochastic-demand forecasting, parameter estimation, model selection, decomposition guidance, and neighbourhood search.

A potential strategy is to develop Reinforcement Learning strategies that learn directly from historical operational data without requiring explicit probability distributions. Deep Reinforcement Learning (DRL) approaches [18,21,22] have demonstrated particular promise in addressing stochastic VRP via interaction with simulated environments. These methods formulate routing decisions as a Markov Decision Process, enabling dynamic responses to uncertain conditions such as stochastic travel times, service times, and customer demands. An interesting approach based on attention-driven transformers that capture circularity and symmetry is proposed in [10]. These methods effectively manage time uncertainty, also within a framework where correlations among duration times are

present [13]. Correlated stochastic demands are also handled in [9], which presents a Bayesian learning framework integrated with an elementary branch-price-and-cut algorithm.

The other possible framework deals with the integration of forecast algorithms. Rather than modeling uncertainty through probability distributions, these approaches utilize machine learning models to directly predict uncertain parameters from historical data. [5] develops a predict-then-optimize framework that integrates graph neural networks with combinatorial optimization for vehicle routing under uncertainty.

Hybrid methodologies that combine forecasting with optimization have shown particular effectiveness for SVRPs with stochastic service times. For example, [23] develops a Grouping Evolution Strategy (GES) algorithm. The proposed GES algorithm effectively handles the problem's complexity by integrating specialized solution representations, mutation operators, and local search heuristics.

As already emphasized, these data-driven paradigms offer significant advantages over traditional approaches: they eliminate the need for explicit probability distributions, adapt to real-time conditions through continuous learning, and leverage historical patterns to inform routing decisions. As noted in the comprehensive review [28], ML methods are increasingly bridging the gap between theoretical optimization models and practical implementation in evolving urban logistics environments.

Hence, in the current paper, we adopt this perspective and we develop a data-driven workflow that trains an XGBoost model on historical service-duration records spanning several years to predict the actual time required for each planned activity. The resulting point forecasts, combined with calibrated prediction intervals, are fed directly into the VRP solver.

3. Solving the VRP via evolutionary algorithm

The Vehicle Routing Problem can be formulated on a graph $G = (V, A)$, where V represents nodes corresponding to activities and A represents feasible connections between these activities.

3.1. The capacitated VRP with uncertain activity durations

Let $G = (V, A)$ be a directed graph, where the vertex set $V = \{0\} \cup N \cup \{n+1\}$ comprises the starting depot 0, the ending depot $n+1$, and the set of customers $N = \{1, \dots, n\}$. The fleet consists of vehicles $K = \{1, \dots, m\}$. We assume deterministic travel times $t_{ij} \geq 0$ (in minutes) and travel costs $c_{ij} \geq 0$. Furthermore, each customer i is associated with a time window $[a_i, b_i]$, and each vehicle k is constrained by a maximum shift duration H_k.

Service duration at i is random: $S_i = \mu_i + \varepsilon_i$, where $\mu_i = \mu(x_i)$ is the ML prediction from features x_i, and ε_i is zero-mean noise. A plan consists of binary routing variables $x_{ijk} \in \{0, 1\}$ and continuous start times $T_{ik} \geq 0$ (min).

The stochastic CVRPTW with chance constraints is formulated as:

$$\min \sum_{k \in K} \sum_{i,j \in V} c_{ij} x_{ijk} + \lambda \sum_{i \in N} \delta_i \tag{1}$$

$$\text{s.t.} \sum_{j \in V} x_{0jk} = 1, \ \sum_{i \in V} x_{i,n+1,k} = 1 \qquad \forall k \in K \tag{2}$$

$$\sum_{j \in V} x_{jik} = \sum_{j \in V} x_{ijk} \qquad \forall i \in N, \forall k \in K \tag{3}$$

$$\sum_{k \in K} \sum_{j \in V} x_{ijk} = 1 \qquad \forall i \in N \tag{4}$$

$$T_{jk} \geq T_{ik} + \mu_i + t_{ij} - M(1 - x_{ijk}) \qquad \forall i, j \in V, \ \forall k \in K \tag{5}$$

$$a_i \leq T_{ik} \leq b_i + \delta_i, \ \delta_i \geq 0 \qquad \forall i \in N, \ \forall k \in K \tag{6}$$

$$\mathbb{P}\{T_{n+1,k} - T_{0k} \leq H_k\} \geq 1 - \alpha_k \qquad \forall k \in K \tag{7}$$

Table 1
Notation.

Symbol	Meaning
$G = (V, A)$	Directed graph (depot 0, customers N)
K	Set of vehicles, $\lvert K \rvert = m$
c_{ij}, t_{ij}	Travel cost/time (min) from i to j
$[a_i, b_i]$	Time window at customer i
H_k	Max duration (shift length, min) for vehicle k
α_k	Acceptable risk level for vehicle k
S_i	Random service duration (min) at i
μ_i	Predicted service duration (point forecast, min)
σ_i^2	Residual proxy variance (sub-Gaussian)
$\alpha, \tilde{\alpha}$	Route and per-stop risk levels
U_i	One-sided conformal upper width at level $1 - \tilde{\alpha}$
x_{ijk}	Binary: vehicle k uses arc (i, j)
T_{ik}	Planned start time (min) of service at i by k
δ_i	Tardiness slack (min) for soft time windows

$$x_{ijk} \in \{0,1\},\ T_{ik} \geq 0. \tag{8}$$

The objective function (1) minimizes travel costs c_{ij} and penalizes tardiness δ_i via weight λ. Constraint (2) ensures each vehicle leaves and returns to the depot. Constraint (3) maintains flow consistency. Constraint (4) ensures each customer is visited exactly once. Constraint (5) propagates time using big-M formulation, where M is a sufficiently large constant. Constraint (6) enforces time windows with soft tardiness slack δ_i. Critically, constraint (7) is a *chance constraint* ensuring that route duration stays within shift length H_k with probability of at least $1-\alpha_k$. The decision variables x_{ijk} and T_{ik} are deterministic; the challenge lies in satisfying the chance constraint (7), which depends on the sum of random service durations S_i along each route. To handle this, we need a tractable way to model the uncertainty from the prediction errors, ε_i.

We model the prediction residuals ε_i as independent sub-Gaussian random variables. This assumption is a powerful generalization of the Gaussian distribution, providing strong guarantees for the sum of variables, even when their individual distributions are not perfectly normal. Its practical importance lies in the fact that it permits controlling the risk of an entire route failing to meet its time constraint without needing to know the exact distribution of each service time. While the underlying true duration distributions show some skewness (see Fig. 5), the sub-Gaussian assumption on the model residuals is reasonable, as we will demonstrate in Section 4 (see Fig. 3). We also refer to Fig. 6 for the CDF of the underlying data.

Assuming the residuals ε_i are independent and sub-Gaussian with proxy variance σ_i^2 (estimated from model residuals), their sum concentrates tightly around its mean. For a route R and any $u > 0$, we have the concentration inequality:

$$\mathbb{P}\left\{\sum_{i\in R}\varepsilon_i \geq u\right\} \leq \exp\left(-\frac{u^2}{2\sum_{i\in R}\sigma_i^2}\right). \tag{9}$$

Setting the RHS to α yields a route-level buffer:

$$\Delta_\alpha(R) \ := \ \sqrt{2\left(\sum_{i\in R}\sigma_i^2\right)\log(1/\alpha)}. \tag{10}$$

We can then state a sufficient condition for chance feasibility:

Proposition 3.1 (Route-level chance feasibility via sub-Gaussian buffer). *Let R_k be the sequence of customers on vehicle k's route and suppose $S_i = \mu_i + \varepsilon_i$ with independent sub-Gaussian ε_i having proxy variances σ_i^2. If*

$$\sum_{i\in R_k}\mu_i + \sum_{(i,j)\in R_k} t_{ij} + \sqrt{2\log(1/\alpha_k)\sum_{i\in R_k}\sigma_i^2} \ \leq \ H_k,$$

then $\mathbb{P}\{T_{n+1,k} - T_{0k} \leq H_k\} \geq 1-\alpha_k$.

Proof. By sub-Gaussian concentration, the sum of independent sub-Gaussian random variables $\sum_{i\in R_k}\varepsilon_i$ satisfies the tail bound above. The route duration is $\sum_{i\in R_k}(\mu_i+\varepsilon_i) + \sum_{(i,j)\in R_k} t_{ij}$. The deterministic part is $\sum_{i\in R_k}\mu_i + \sum_{(i,j)\in R_k} t_{ij}$. Adding the buffer $\Delta_{\alpha_k}(R_k)$ ensures that with probability at least $1-\alpha_k$, the stochastic deviations $\sum_{i\in R_k}\varepsilon_i$ do not exceed the buffer; hence the total route duration stays within H_k. □

Remark 3.1. As an alternative approach, one can use conformal prediction intervals. Suppose one-sided $(1-\tilde{\alpha})$ conformal upper bounds U_i satisfy $\mathbb{P}\{S_i \leq \mu_i + U_i\} \geq 1-\tilde{\alpha}$ for all i. Using Bonferroni union bound with $\tilde{\alpha} = \alpha_k/|R_k|$, we obtain:

$$\mathbb{P}\left\{\sum_{i\in R_k} S_i \leq \sum_{i\in R_k}(\mu_i + U_i)\right\} \ \geq \ 1-\alpha_k. \tag{11}$$

This provides distribution-free coverage guarantees without requiring sub-Gaussian assumptions, albeit at the cost of potentially wider intervals due to the Bonferroni correction. While this provides a robust distribution-free guarantee, the union bound can be overly conservative for routes with many activities. For practical planning systems, integrating point forecasts with a single, aggregate risk buffer, as enabled by the sub-Gaussian approach, is often more tractable. Therefore, we adopt the latter for our main analysis.

3.2. The optimization algorithm

The evolutionary approach addresses the multi-objective nature of the Stochastic VRP with stochastic durations by simultaneously optimizing conflicting objectives. Real-world routing operations involve competing KPIs: minimizing travel costs, reducing overtime, meeting time windows, and maximizing service coverage.

We formulate the CPVRTW introduced in Section 3.1 as a multi-objective optimization problem where the objective vector is:

$$\min\left(\underbrace{\sum_k\sum_{i,j} c_{ij}x_{ijk}}_{\text{travel cost}},\ \underbrace{\sum_i \delta_i}_{\text{time-window violations}},\ \underbrace{\sum_k \text{Overtime}_k}_{\text{overtime}},\ \underbrace{-\#\text{served}}_{\text{maximize coverage}}\right). \tag{12}$$

Here Overtime_k represents the amount of time vehicle k exceeds shift length H_k, computed as $\max(0, T_{n+1,k} - T_{0k} - H_k)$. The negative sign on served tasks converts maximization to minimization. In a more general form, the multi-objective problem is:

$$\min_{\mathbf{x}\in\Omega} \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_m(\mathbf{x})) \tag{13}$$

where $\mathbf{x}$ represents a routing solution, Ω is the feasible solution space, and $\mathbf{F}(\mathbf{x})$ is the vector of m objective functions as specified in Eq. (12).

Rather than scalarizing Eq. (13) via fixed weights, we employ a Multi-Objective Evolutionary Algorithm (MOEA), specifically NSGA-III [16], to explore the Pareto front and provide decision-makers with diverse trade-off solutions.

The algorithm maintains a population P_t of μ individuals at generation t, where each individual represents a complete routing solution. Solutions are ranked using Pareto dominance: solution $\mathbf{x}_1$ dominates $\mathbf{x}_2$ (denoted $\mathbf{x}_1 \prec \mathbf{x}_2$) if:

$$\forall i \in \{1, 2, \dots, m\} : f_i(\mathbf{x}_1) \leq f_i(\mathbf{x}_2) \quad \text{and} \quad \exists j \in \{1, 2, \dots, m\} : f_j(\mathbf{x}_1) < f_j(\mathbf{x}_2) \tag{14}$$

fronts contain progressively dominated solutions.

Specifically, an evolutionary algorithm for solving the stochastic VRP introduced in (1)–(8) can be formalized via the following steps:

1. Initialization: Generate an initial population P_0 using a greedy heuristic that inserts tasks into routes based on predicted durations μ_i, prioritizing time-window feasibility.
2. Variation: For each generation, apply selection, crossover, and mutation to generate offspring $\mathcal{O}$.
3. Repair and Local Search: Apply a hierarchical repair mechanism to restore feasibility and consistency, followed by stochastic first-improvement moves.
4. Uncertainty Integration: For each candidate route, compute the sub-Gaussian risk buffer $\Delta_\alpha(R)$ and evaluate chance-constrained feasibility.
5. Multi-Objective Evaluation: Calculate the objective vector $\mathbf{F}(\mathbf{x})$ for all individuals.
6. Selection: Use NSGA-III non-dominated sorting and niching based on reference points to select the next generation P_{t+1}.
7. Adaptation: Dynamically adjust reference points based on the population distribution in the objective space.

Reference point adaptation. Structured reference points in $\mathbf{R} = \{\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_H\}$ are utilized to maintain diversity. The algorithm adaptively updates these points based on the current population distribution. For each niche defined by a reference point $\mathbf{r}_i$, the niche count ρ_i is tracked. If a region of the Pareto front is sparsely populated, additional reference points are generated around existing ones using:

$$\mathbf{r}_{new} = \mathbf{r}_i + \Delta\mathbf{r} \tag{15}$$

where $\Delta\mathbf{r}$ represents adaptive perturbations scaled by the distance to the nearest individuals, ensuring uniform coverage of the trade-off surface.

Evolutionary operators. Parent selection employs tournament selection with tournament size τ. We utilize the *Order Mutation (OM)* crossover operator, which selects a random subtour from one parent and re-inserts the remaining nodes from the second parent while preserving their relative order. This ensures that the permutation structure of the routes is maintained.

Mutation follows a *Hierarchical Component Density Function* strategy, which we introduce as a tiered mutation approach designed to navigate the highly constrained search space of the stochastic CVRPTW. This mechanism is justified by the need to efficiently restore solution feasibility and consistency before pursuing objective optimization. By prioritizing moves based on the current state of the solution, the algorithm ensures that computational effort is first directed toward satisfying hard constraints and domain requirements:

$$\text{Mutate}(\mathbf{x}) = \begin{cases} \text{Feasibility Moves} & \text{if } \mathbf{x} \notin \text{Constraints} \\ \text{Consistency Moves} & \text{if } \mathbf{x} \notin \text{Domains} \\ \text{Fitness Moves} & \text{otherwise} \end{cases} \tag{16}$$

Feasibility moves include task pruning or re-insertion to satisfy hard constraints. Fitness moves include a hybrid strategy of swap (exchanging two nodes), relocate (moving a node), and 2-opt moves to improve route compactness.

In addition to the variation operators, each offspring undergoes a *quality repair* step. This is implemented as a stochastic local search that applies relocation and 2-opt moves. It is designed as a first-improvement heuristic with a limited number of successful moves to balance refinement with computational efficiency. Finally, solutions are evaluated against all objective functions. We present the Algorithm 1, which is based on [16,17].

The EA-based approach is particularly advantageous for stochastic VRP because it naturally maintains a diverse set of solutions, providing decision-makers with alternatives that represent different trade-offs between cost efficiency and robustness. Moreover, it can easily incorporate complex objective functions and constraints without requiring mathematical reformulation and is less susceptible to getting trapped in local optima, which is especially important in stochastic environments where the fitness landscape can be highly non-convex and noisy.

Algorithm 1 Forecast-aware MOEA for Stochastic CVRPTW.

Input: Instance $(G, t, c, [a, b], H)$; ML model $\hat{\mu}$ for predictions μ_i; uncertainty proxy (per-stop σ_i or conformal U_i); MOEA parameters; risk levels α_k
Initialize population P_0 by greedy time-window insertion using predicted durations μ_i
$t \leftarrow 0$
while termination criteria not satisfied **do**
    $t \leftarrow t + 1$
    Variation: Apply Order Crossover (OX) with structural repair, followed by adaptive mutation
    Repair: Stochastic first-improvement local search (relocate/swap/2-opt) with limited iterations
    Risk buffer: For each route R (vehicle k) in $\mathcal{O}$, compute buffer $\Delta_{\alpha_k}(R)$ via Eq. (10) or conformal method
    Feasibility check: Penalize routes if $\sum_{i \in R} \mu_i + \sum_{(i,j) \in R} t_{ij} + \Delta_{\alpha_k}(R) > H_k$
    Evaluate multi-objective vector (Eq. 12): (travel, tardiness, overtime, −served)
    Non-dominated Sorting: Organize $P_t \cup \mathcal{O}$ into Pareto fronts $F_1, F_2, \ldots$
    Select μ individuals for P_{t+1} via niching and crowding distance (NSGA-III)
    Update convergence and diversity metrics
end while
return Pareto set P_G; decision-maker chooses policy via KPI preferences

4. Predictive algorithms for informed optimization methods

This section presents the machine learning framework developed to address the problem of uncertainty in intervention times for workforce management. The forecast engine implements a modular architecture that integrates predictive algorithms into the stochastic VRP optimization pipeline introduced in Section 3.

4.1. Mathematical framework and architecture overview

We now formalize the mathematical framework for duration prediction. We develop four model architectures addressing class imbalance, heterogeneous activity types, and multimodal duration distributions.

The forecast engine implements a unified mathematical framework for duration prediction. Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space representing the stochastic environment, and let $\mathcal{X} \subseteq \mathbb{R}^d$ denote the feature space with $\mathcal{Y} \subseteq \mathbb{R}_+$ representing the space of intervention durations. Let the target variable Y correspond to the service duration S defined in the VRP model in Section 3.1. The historical dataset $\mathcal{D} = \{(X_i, Y_i)\}_{i=1}^n$ consists of independent and identically distributed pairs where $X_i : \Omega \to \mathcal{X}$ are feature vectors and $Y_i : \Omega \to \mathcal{Y}$ are corresponding intervention durations.

The forecasting problem seeks to learn a predictive mapping $f^* : \mathcal{X} \to \mathcal{Y}$ that minimizes the expected risk under a specified loss function $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$:

$$f^* = \arg\min_{f \in \mathcal{F}} \mathbb{E}_{(X,Y)\sim\mathbb{P}}[L(Y, f(X))] \tag{17}$$

where $\mathcal{F}$ represents the hypothesis space of admissible functions.

The forecast engine transforms historical operational data into duration predictions for route optimization. The system has two phases: (*i*) an offline training pipeline that learns patterns from historical data; (*ii*) an online inference pipeline that generates predictions for daily planning. This separation enables exhaustive model optimization during training while preserving low-latency, i.e., fast inference at deployment.

The training pipeline (Fig. 1) processes historical data through three stages: data collection, feature engineering and model training with validation.

The data preparation stage validates data quality, detecting missing values, outliers, and inconsistencies. The dataset spans eight years and handles evolving activity categories and changing service areas.

The feature engineering stage transforms raw measurements into predictive features. Intervention durations depend on temporal patterns (hour-of-day, day-of-week, seasonality), geography (municipality, altitude, urbanization), and operational context (activity type, equipment, historical rates). We create interaction terms and polynomial features to capture these dependencies.



Fig. 1. Training framework for duration prediction.



Fig. 2. Inference framework for operational deployment.

The model estimation stage trains four XGBoost architectures: standard, frequency-weighted (for class imbalance), dual (for heterogeneous activities), and dual-weighted (combined). We optimize hyperparameters using Bayesian optimization and validate using stratified k-fold cross-validation, temporal holdouts, and activity-specific performance analysis.

The inference framework (Fig. 2) generates real-time duration predictions for VRP solvers, balancing speed and accuracy.

The feature consistency module applies the same transformations (cyclical encodings, interactions, normalizations) during inference as it does during training. It detects out-of-distribution inputs and handles missing values using validated imputation strategies.

The model selection module routes requests to specialized models: f_Z handles Type-Z interventions (meter replacements), while f_{other} handles standard activities.

4.2. Model architectures

To approximate f^* in Eq. (17), we try the following tailored gradient-boosting models:

1. Standard Gradient Boosting Model
2. Frequency-Weighted Gradient Boosting Model
3. Dual Model Architecture
4. Dual Models with Frequency Weighting

1) standard gradient boosting model. The baseline approach implements gradient boosting with uniform sample weighting. The model constructs an ensemble predictor through additive tree learning:

$$f_{\text{std}}(\mathbf{x}) = \sum_{m=1}^{M} \gamma_m \cdot T_m(\mathbf{x}) \tag{18}$$

where M represents the total number of trees, $\gamma_m \in \mathbb{R}$ denotes the weight of tree m, and $T_m : \mathbb{R}^d \to \mathbb{R}$ represents the m-th decision tree mapping feature vectors to leaf values. The training process optimizes a regularized empirical risk functional:

$$\mathcal{J}_{\text{std}}(\mathbf{f}) = \frac{1}{n} \sum_{i=1}^{n} L(y_i, f_{\text{std}}(\mathbf{x}_i)) + \lambda \sum_{m=1}^{M} \Phi(T_m) \tag{19}$$

where $L : \mathbb{R} \times \mathbb{R} \to \mathbb{R}_+$ represents the loss function, $\lambda > 0$ controls regularization strength, and $\Phi(T_m)$ implements structural penalties on tree complexity, including leaf count and weight magnitudes.

2) frequency-weighted gradient boosting model. To address class imbalance across activity types, the framework implements sample-specific weighting based on activity type prevalence. The weighting mechanism assigns higher importance to underrepresented activity types through inverse frequency scaling:

$$\omega_i^{(c)} = \frac{n}{n_c \cdot |C|} \tag{20}$$

where n represents total sample size, n_c denotes samples from activity type c, and $|C|$ represents the number of distinct activity types.

The weighted objective modifies the standard formulation by incorporating sample-specific importance:

$$\mathcal{J}_{\text{weighted}}(\mathbf{f}) = \frac{1}{n} \sum_{i=1}^{n} \omega_i^{(c_i)} \cdot L(y_i, f_{\text{weighted}}(\mathbf{x}_i)) + \lambda \sum_{m=1}^{M} \Phi(T_m) \tag{21}$$

where c_i denotes the activity type of sample i. This formulation ensures that rare activity types receive appropriate representation during model training despite their lower frequency in the operational data.

4.2.1. Dual models

To account for the strong heterogeneity observed across activity types, as evidenced by the duration distributions in Figs. 16 and 17, separate predictive models were developed for specific categories of interventions. Precisely, we consider models that generalize Eq. (17) and that support heterogeneous modeling through domain decomposition.

Table 2
Summary of Model Architectures.

	# Models	Weighting
Standard: Baseline	1	
Frequency-Weighted: Class imbalance	1	✓
Dual: Specialized models	2	
Dual-Weighted: Specialization + imbalance	2	✓

Let $\{\mathcal{X}_k\}_{k=1}^K$ be a partition of the feature space such that $\bigcup_{k=1}^K \mathcal{X}_k = \mathcal{X}$ and $\mathcal{X}_i \cap \mathcal{X}_j = \emptyset$ for $i \neq j$. The heterogeneous predictor is then defined as:

$$f_{\text{het}}(\mathbf{x}) = \sum_{k=1}^{K} \mathbb{1}_{x \in \mathcal{X}_k} \cdot f_k(\mathbf{x}) \tag{22}$$

where each $f_k : \mathcal{X}_k \to \mathcal{Y}$ is specialized for the patterns within subdomain $\mathcal{X}_k$. This decomposition enables the learning of specialized models that capture distinct operational regimes while maintaining global consistency.

3) dual models architecture. This approach implements a specialized dual architecture that trains separate models for distinct intervention categories. Building upon the heterogeneous predictor framework in Eq. (18), we partition activities into two semantically meaningful classes based on the activity type feature $a(\mathbf{x})$:

$$f_{\text{dual}}(\mathbf{x}) = \begin{cases} f_Z(\mathbf{x}) & \text{if } a(\mathbf{x}) = Z \\ f_{\text{other}}(\mathbf{x}) & \text{if } a(\mathbf{x}) \neq Z \end{cases} \tag{23}$$

where $a : \mathbb{R}^d \to C$ represents the activity type classification function, $f_Z : \mathbb{R}^d \to \mathbb{R}_+$ specializes in Type-Z intervention prediction, and $f_{\text{other}} : \mathbb{R}^d \to \mathbb{R}_+$ handles all other activity types.

Each specialized component is trained using the standard gradient boosting objective (Eq. 19) but is optimized for the distinct statistical characteristics of its operational domain. The Type-Z model emphasizes flexibility to capture high variability patterns through reduced regularization, while the other-types model prioritizes stability through enhanced regularization. This architectural separation enables each model to specialize in its respective operational regime while maintaining computational efficiency.

4) dual models with frequency weighting. The most sophisticated approach combines architectural specialization with frequency-based weighting, addressing both structural heterogeneity and statistical imbalance. This represents a hierarchical extension of the dual architecture where each specialized component incorporates sample weighting:

$$f_{\text{dual-weighted}}(\mathbf{x}) = \begin{cases} f_{Z,w}(\mathbf{x}) & \text{if } a(\mathbf{x}) = Z \\ f_{\text{other},w}(\mathbf{x}) & \text{if } a(\mathbf{x}) \neq Z \end{cases} \tag{24}$$

Each component model implements frequency weighting computed within its respective operational domain using the weighting scheme from Eq. (21). The Type-Z component $f_{Z,w}$ applies inverse frequency scaling based on subtype prevalence within Type-Z interventions, while the other-types component $f_{\text{other},w}$ computes weights relative to the distribution of non-Z activity types.

In summary, we present the key characteristics of the four model architectures in Table 2.

A priori, developing separate models for each activity class might seem optimal, but analysis of duration distributions (Fig. 16) reveals that most non-Type-Z activities exhibit similar statistical properties. Moreover, training and maintaining numerous specialized models (beyond the dual approach) present operational challenges in production environments, including increased complexity and the related reduced reliability. Therefore, we select dual architecture as a pragmatic balance between specialization and operational feasibility.

Residual analysis. Recall that in Section 3, we assumed prediction residuals ε_i follow a sub-Gaussian distribution to enable tractable route-level risk buffers. This assumption is empirically supported by the distribution of residuals from the Dual Weighted model. As shown in Fig. 3, the errors are centered at zero and exhibit a symmetric, bell-shaped distribution. The symmetric, bell-shaped distribution of residuals validates our assumption of sub-Gaussianity.

4.2.2. Estimating proxy variance for risk buffers

While the XGBoost model provides point predictions μ_i representing the expected duration of an intervention, the robustness of the routing schedule relies on the sub-Gaussian risk buffers defined in Eq. (10). These buffers require an estimate of the proxy variance σ_i^2 for each activity, which quantifies the uncertainty of the prediction.

We estimate these variances using a residual-based approach stratified by activity type. This method leverages the historical prediction errors on the validation set to construct a lookup table of uncertainties. For each activity class $c \in C$, the empirical proxy variance is computed as:

$$\hat{\sigma}_c^2 = \frac{1}{n_c} \sum_{i \in \mathcal{V}_c} (y_i - \hat{\mu}_i)^2, \tag{25}$$

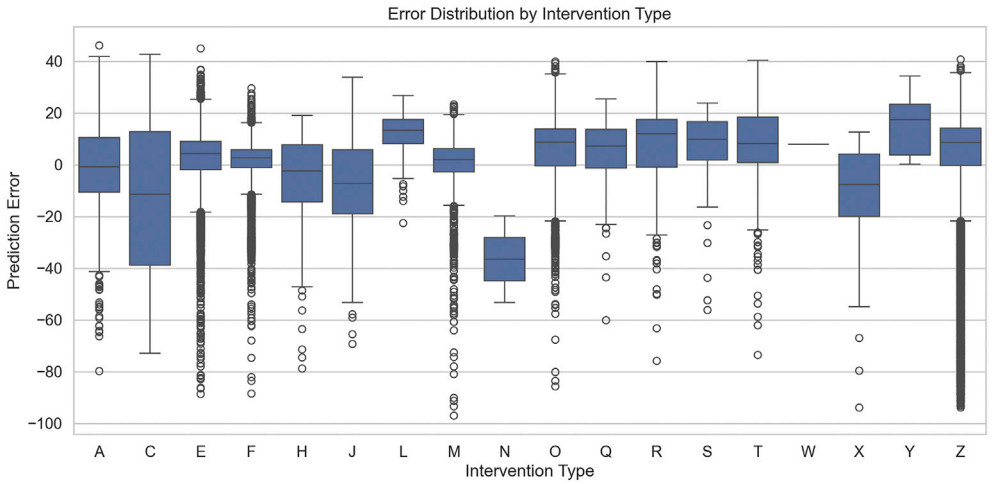


Fig. 3. Distribution of prediction residuals ($\hat{y} - y$) from the Dual Weighted forecast model. The symmetric, bell-shaped curve supports the sub-Gaussian assumption used for risk modeling.

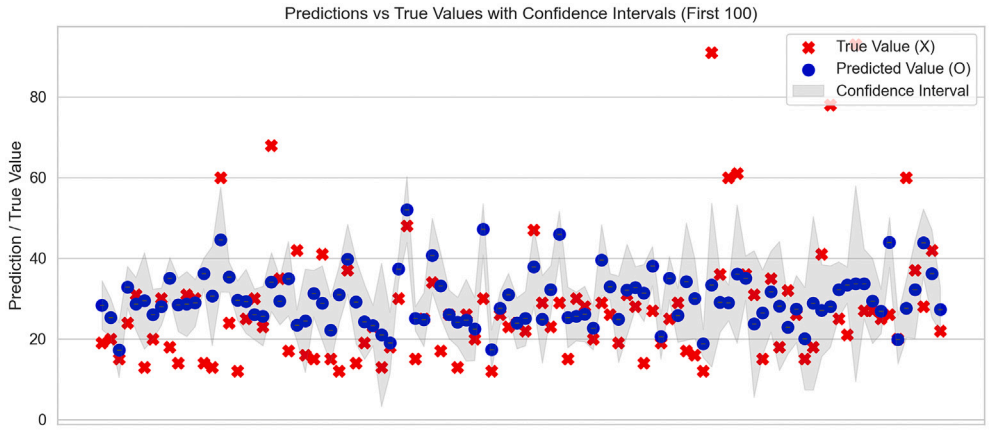


Fig. 4. Duration predictions with uncertainty estimates.

where $\mathcal{V}_c$ is the set of validation samples of type c, and $n_c = |\mathcal{V}_c|$. This approach assumes that interventions of the same type share a similar error profile, an assumption supported by the residual analysis in Fig. 3.

Bridging forecasts and optimization. This variance estimate serves as the critical link between the machine learning module and the stochastic optimization constraints (Proposition 3.1). In practice, the integration functions as a closed-loop system:

1. For a new intervention i, the ML model predicts the expected duration μ_i, while the uncertainty module assigns the corresponding variance σ_i^2 based on Eq. (25).
2. During the evolutionary search, for any candidate route R, the solver calculates the deterministic length ($\sum \mu_i + \sum t_{ij}$) and aggregates the uncertainty to compute the dynamic risk buffer $\Delta_\alpha(R)$ defined in Eq. (10).
3. The route is deemed feasible only if the sum of the deterministic components and the risk buffer remains within the shift limit H_k, as per Proposition 3.1.

This mechanism ensures adaptive conservatism. Interventions with high predictive uncertainty (e.g., Type-Z meter replacements, which exhibit high variance in residuals) contribute larger values to $\sum \sigma_i^2$, automatically generating a larger safety buffer $\Delta_\alpha(R)$. Conversely, routine activities with low variance imply tighter buffers, achieving a higher resource utilization. The parameter α_k acts as a "tuning knob" for calibration, allowing decision-makers to explicitly set the acceptable risk level (e.g., a 5% probability of overtime) without requiring computationally expensive scenario sampling.

Fig. 4 illustrates these estimates, showing how the confidence intervals (derived from σ_i) widen for complex activities and narrow for routine tasks.

The evaluation module. Performance evaluation employs multiple complementary metrics designed to capture different aspects of prediction quality:

$$\text{MAE} = \frac{1}{n_{\text{test}}} \sum_{i=1}^{n_{\text{test}}} |y_i - \hat{y}_i| \tag{26}$$

$$\text{RMSE} = \sqrt{\frac{1}{n_{\text{test}}} \sum_{i=1}^{n_{\text{test}}} (y_i - \hat{y}_i)^2} \tag{27}$$

$$\text{MAPE} = \frac{100\%}{n_{\text{test}}} \sum_{i=1}^{n_{\text{test}}} \left| \frac{y_i - \hat{y}_i}{y_i} \right| \tag{28}$$

The hyperparameters of the learning algorithms (1)–(4) introduced in the previous paragraph have been calibrated via a systematic Bayesian optimization. The optimization process explores the hyperparameter space Θ to minimize validation error:

$$\theta^* = \arg\min_{\theta \in \Theta} \mathbb{E}_{\text{val}}[\mathcal{L}(\theta)] \tag{29}$$

where $\mathbb{E}_{\text{val}}[\mathcal{L}(\theta)]$ represents expected validation loss under hyperparameter configuration θ.

The optimization framework employs acquisition functions that balance exploration of uncertain regions with exploitation of promising parameter combinations. The expected improvement criterion guides the search process:

$$\text{EI}(\theta) = \mathbb{E}[\max(f_{\min} - f(\theta), 0)] \tag{30}$$

where $f_{\min}$ represents the current best observed performance and $f(\theta)$ denotes the objective function value at parameter configuration θ.

5. Case study and experimental methodology

This section presents the experimental framework for evaluating machine-learning-based duration prediction for routing optimization in real-world utility service operations.

5.1. The VRP model for gas meter intervention

In the context of gas meter maintenance operations, accurate prediction of intervention durations is crucial for optimal route planning and resource allocation. This chapter presents a comprehensive methodology for evaluating the impact of different duration estimation strategies on Vehicle Routing Problem (VRP) solutions through evolutionary algorithm optimization. We develop a comparative framework to compare three distinct duration input modes: *real*, *default*, and *forecast*, enabling quantitative assessment of prediction accuracy impact on routing efficiency.

The gas meter maintenance routing problem can be formalized as a Capacitated Vehicle Routing Problem with Time Windows (CVRPTW), where:

- $G = (V, A)$ represents a complete directed graph
- $V = \{0, 1, 2, \ldots, n, n+1\}$ is the vertex set, where vertices 0 and $n+1$ represent the depot
- $A = \{(i, j) : i, j \in V, i \neq j\}$ is the arc set
- Each customer $i \in \{1, 2, \ldots, n\}$ has:
  - Service time window $[e_i, l_i]$
  - Service duration s_i (the focus of our study)
  - Demand d_i
- K identical vehicles with capacity Q
- Travel cost c_{ij} and travel time t_{ij} (min) between locations i and j

The objective is to minimize total travel cost while respecting capacity and time window constraints. As per the notation in Table 1, we distinguish between travel cost c_{ij} (routing objective) and travel time t_{ij} (temporal feasibility), with service duration denoted S_i to align with the stochastic formulation in Section 3.1.

The critical challenge addressed in this research is the accurate estimation of service durations S_i for each intervention. Traditional approaches often rely on static estimates or historical averages, which fail to capture the complexity and variability inherent in gas meter maintenance operations.

5.2. Experimental setup: dataset and feature engineering

The experimental analysis utilizes extensive real-world utility service data collected over an eight-year operational period spanning 2017-2025, encompassing 884,349 individual service requests across multiple utility distribution networks.

1. *Intervention Taxonomy*: Distinct activity types (A, C, E, F, H, I, J, L, M, N, O, Q, R, S, T, W, X, Z) representing the full spectrum of gas meter operations. We refer to the Table 6 in the Appendix for a complete description of each class.

2. *Temporal Features*: Each intervention record includes request timestamp, scheduled appointment, actual arrival time, and completion time.
3. *Geographic Features*: Location-based attributes include municipality identifiers, altitude, urbanization degree, population density, and surface area.
4. *Operational Features*: Activity-specific characteristics such as meter class (residential vs. commercial/industrial), physical accessibility level, reading difficulty, and equipment protocol specifications.
5. *Client Source*: The dataset aggregates records from five distinct utility clients, each with potentially different operational practices, recording standards, and service area characteristics.

Data split and evaluation strategy. We adopt a random date-based split where specific dates (e.g., June 5–7, 2024) are held out from the training set and used exclusively for testing. This ensures that the model is evaluated on complete days not seen during training, capturing realistic operational scenarios where predictions must generalize to new dates. All features used for prediction are available at planning time (prior to route execution), excluding any post-service information (e.g., actual completion times, reported issues) to prevent data leakage. This date-level holdout, which we highlight as a best practice for operational forecasting, differs from random sample splitting that could mix activities from the same day across training and test sets. By preserving the temporal integrity of entire operational shifts, it prevents data leakage and provides a more rigorous and realistic assessment of model performance in production deployment.

The raw distribution of intervention durations exhibits artificial spikes at round-number values (e.g., 5, 10, 15 minutes), typically caused by reporting biases rather than genuine operational patterns. A peak smoothing algorithm was applied: detected peaks were partially redistributed to neighboring values within a defined range. This process reduces artificial clustering while preserving overall statistical properties, providing higher-quality inputs for predictive modeling and more reliable cumulative and density functions.

Statistical analysis reveals highly skewed duration distributions with heavy tails, reflecting inherent variability from routine 15-minute operations to complex multi-hour interventions. The temporal distribution exhibits both deterministic patterns following daily and seasonal cycles, and stochastic variations driven by unpredictable operational factors.

The feature engineering pipeline constructs a 31-dimensional representation through domain-specific transformations designed to capture multiscale temporal and spatial dependencies. We perform the following techniques:

1. *Cyclical Temporal Encoding*: Time-based variables exhibiting natural periodicity are encoded using trigonometric transformations to preserve topological relationships. For a temporal variable $t \in [0, T)$ with period T:

$$\phi_{\text{cyc}}(t; T) = \left(\sin\left(\frac{2\pi t}{T}\right), \cos\left(\frac{2\pi t}{T}\right)\right) \in \mathbb{S}^1 \tag{31}$$

   This encoding is applied across multiple temporal scales: hourly patterns capturing intra-day operational variations, daily patterns representing day-of-week effects, and seasonal patterns encoding monthly demand cycles.
2. *Geographic Feature Extraction*: Spatial attributes derived from municipality-level administrative data include elevation and altimetric zone classification, population and surface area for density calculations, and urbanization degree categories.
3. *Activity Type Encoding*: Categorical intervention types are encoded to preserve semantic relationships between operationally similar activities while maintaining model interpretability.

5.3. Performance of forecast models

The dataset partitioning strategy employs stratified random sampling with 80% allocated to training, 10% to validation, and 10% to test, ensuring balanced representation across activity types.

Table 3 presents comprehensive evaluation metrics, i.e., the ones introduced in Eqs. (26)–(28), across all model configurations.

Table 3
Evaluation Metrics (MAE, RMSE, MAPE) for each model and dataset.

Model	Set	MAE	RMSE	MAPE
Standard	Train	5.1961	8.3679	27.63%
	Validation	5.3776	8.6604	28.63%
	Test	5.2793	8.5481	27.71%
Weighted	Train	5.2389	8.3844	27.79%
	Validation	5.4098	8.6654	28.76%
	Test	5.3228	8.5807	27.86%
Dual Standard	Train	5.0488	8.1166	26.75%
	Validation	5.3036	8.5177	28.11%
	Test	5.2374	8.4473	27.36%
Dual Weighted	Train	4.5568	7.0698	24.35%
	Validation	5.1240	8.1976	27.21%
	Test	5.0629	8.1022	26.51%

5.3.1. Discussions of the prediction results

This motivated the `Weighted Model` configuration applying inverse frequency weighting. Table 3 shows minimal improvement: MAE 5.2793 (Standard) → 5.3228 (Weighted), MAPE 27.71% → 27.86%. The Standard model already generalizes effectively across activity types due to distributional similarity—the numerical imbalance does not induce typical minority class performance degradation that frequency weighting addresses.

Activity type `Z` (Meter replacement) represents a compositionally distinct operation combining two sequential sub-tasks (meter removal + installation), whereas other categories represent atomic operations. This manifests in duration distribution, plotted in Fig. 17: Type `Z` exhibits bimodal patterns with prominent peaks near 60 minutes, contrasting with unimodal distributions of other activities. Additionally, meter class (residential vs. commercial/industrial) exerts a stronger influence on `Z` durations due to varying equipment complexity.

The `Dual Model` architecture trains separate, specialized models: f_Z, optimized for Type Z with the flexibility to capture bimodal patterns, and f_{other}, for the remaining activities with enhanced regularization. Table 3 shows architectural specialization yields measurable gains: `Dual Standard` achieves 5.2374 MAE (0.4 min improvement), while `Dual Weighted achieves` 5.0629 MAE (2.2 min improvement), reaching optimal 26.51% MAPE.

The improvement given by the `Dual Weighted` configuration validates that semantic heterogeneity (the compositional complexity of Type Z) warrants architectural separation, with frequency weighting within each specialized model providing incremental refinement, as shown in Fig. 11.

5.3.2. Comparison against operational baseline

The experimental evaluation includes comparison against operational baseline models currently deployed in production systems. These implement rule-based duration estimation using fixed default values $d_{\text{baseline}}(c)$ assigned per activity type $c \in C$.

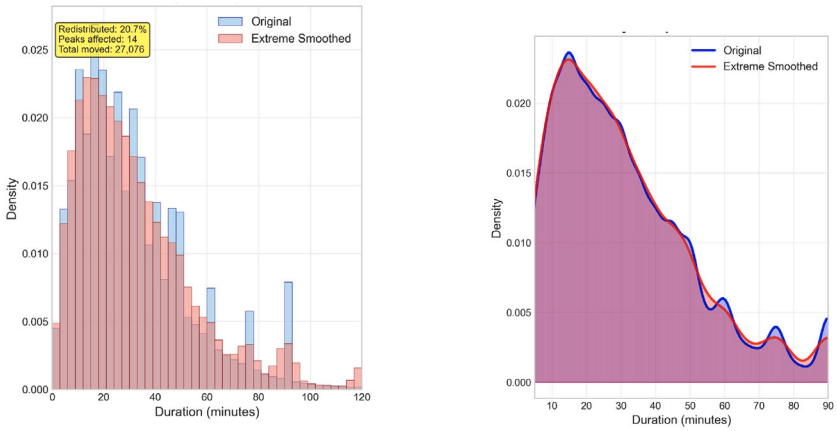


Fig. 5. (a) Original and smoothed duration distributions. (b) Corresponding probability density functions showing the smoothing effect.

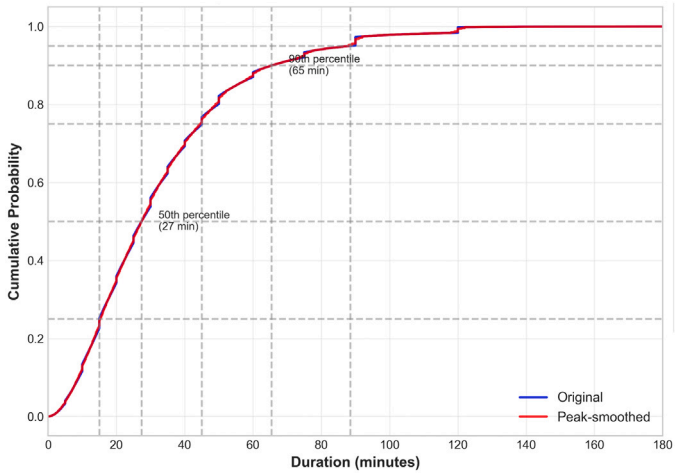


Fig. 6. Cumulative Distribution Function (CDF) of intervention durations.

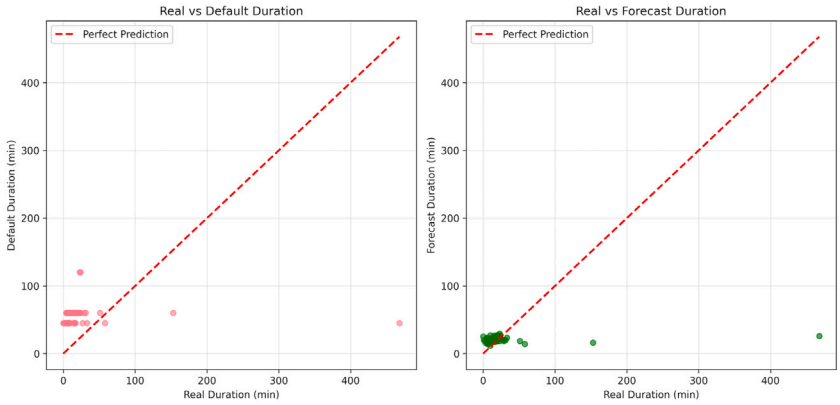


Fig. 7. Prediction accuracy comparison: Real vs. Forecast vs. Default duration estimates.

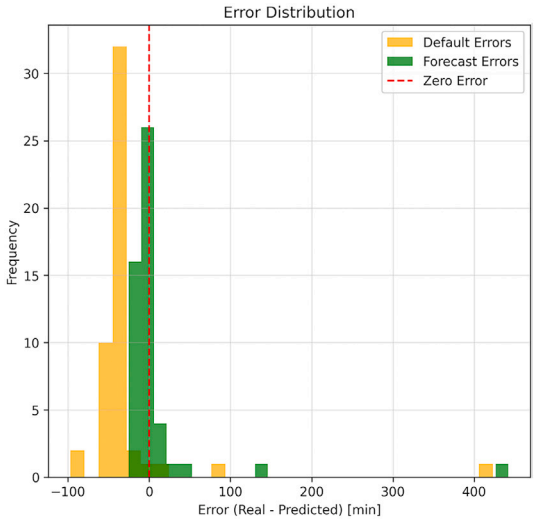


Fig. 8. Error distribution shift: Forecast predictions vs. default values.

Figs. 7 and 8 demonstrate significant improvements across all machine learning configurations compared to traditional rule-based estimation. The forecast strategy achieves a mean accuracy score of 0.456 compared to 0.369 for the default, representing 23.6% improvement in planning accuracy.

Fig. 9 provides a detailed view of the most frequent activity types (E, Z, F), showing how forecasts closely match actual durations.

5.3.3. Model comparison across daily data

To evaluate model performance consistency, we analyze duration predictions across multiple representative operational days. We compare four model architectures: Standard, Weighted by Frequency, Dual Standard, and Dual Weighted by Frequency.

Duration distribution analysis. Fig. 10 shows duration distributions comparing model predictions against actual values for a representative day.

Boxplot analysis by model architecture. Fig. 11 presents detailed boxplot comparisons for a representative day, showing all four model architectures.

The boxplot analysis reveals that dual weighted models consistently achieve tighter prediction intervals and better alignment with actual durations across different operational scenarios.

While the point prediction performance (MAE, RMSE) demonstrates substantial improvement over baseline methods, risk-aware routing as formulated in Section 3.1 requires well-calibrated prediction intervals. The sub-Gaussian buffers (Eq. 10) and conformal prediction approaches discussed in Section 4 depend critically on accurate uncertainty estimates.

Our analysis of prediction residuals reveals heteroskedasticity across activity types and operational dates. This finding motivates heterogeneous variance modeling where σ_i^2 depends on contextual features x_i rather than assuming constant variance across all interventions. This approach is particularly important for Type-Z interventions, which exhibit significantly higher variance than routine maintenance activities.

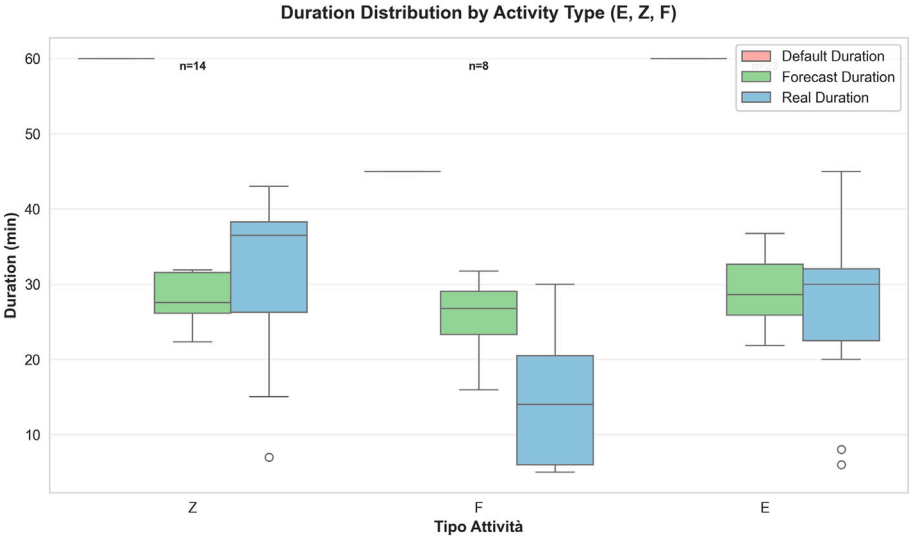
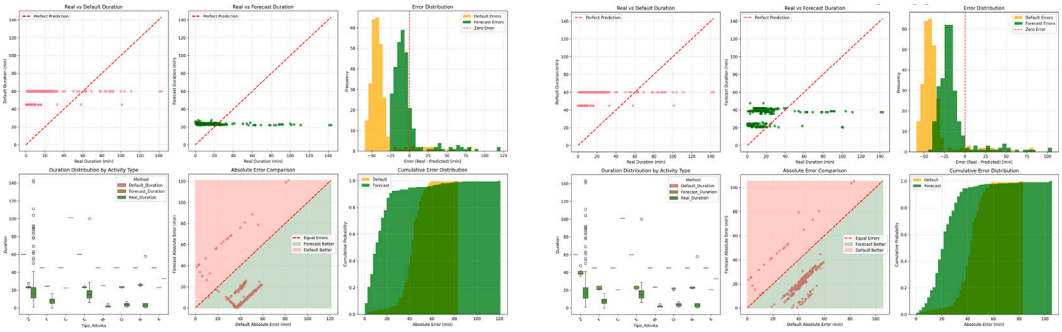
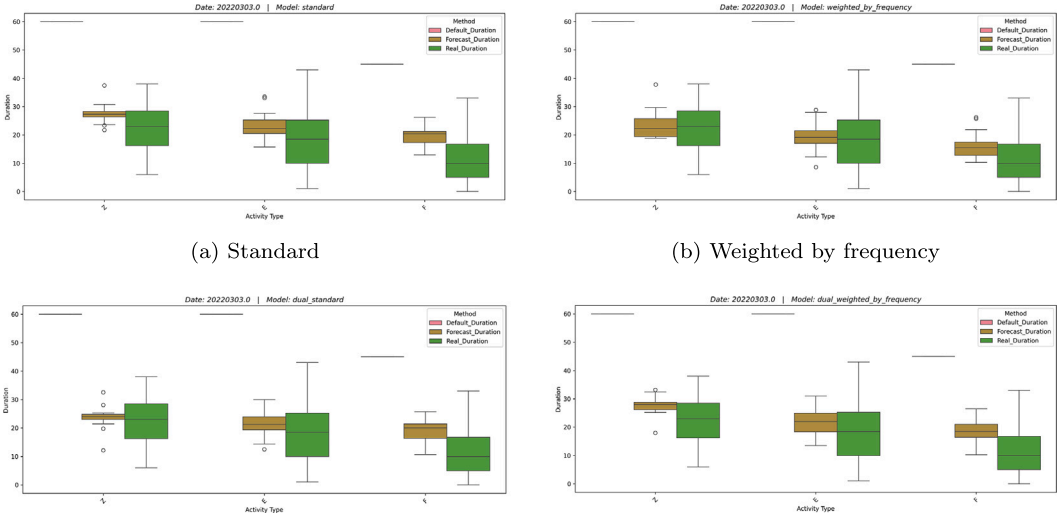


Fig. 9. Detailed comparison of duration estimates for high-frequency activities.



(a) Standard model (b) Dual weighted by frequency model

Fig. 10. Duration distribution comparison for 1 representative day.



(a) Standard (b) Weighted by frequency

(c) Dual standard (d) Dual weighted by frequency

Fig. 11. Duration distributions by activity type for different models (119 interventions).

Properly calibrated intervals would enable principled route-level risk management via Proposition 3.1, balancing service coverage against overtime risk through adjustable confidence levels α_k.

6. VRPs simulation with duration prediction

This section presents the experimental framework designed to evaluate the impact of duration prediction accuracy on routing optimization performance. Our methodology employs a systematic comparative analysis across four duration estimation strategies, implemented within the evolutionary algorithm framework described in Algorithm 1.

We establish four distinct approaches for determining service durations, representing different levels of information sophistication:

1. The real duration strategy utilizes actual historical service times as ground truth:

$$\tau_i^{\text{real}} = t_{\text{completion},i} - t_{\text{start},i} \tag{32}$$

   This serves as an *oracle* providing the theoretical optimal bound assuming perfect duration prediction. Note that this strategy represents an *ex-post* evaluation—it is only possible once durations are already realized and thus serves as a theoretical upper bound rather than an achievable target for real-time planning.
2. The default duration strategy employs static estimates based on intervention classifications:

$$\tau_i^{\text{default}} = \mathbb{E}[\tau_{\text{type}(i)}] \tag{33}$$

   where $\mathbb{E}[\tau_{\text{type}(i)}]$ represents the historical mean duration for intervention type $\text{type}(i)$, reflecting traditional deterministic utility dispatching practices.
3. The unconditional chance-constrained (UCC) strategy represents an established stochastic VRP baseline that accounts for uncertainty using unconditional historical distributions, as originally formalized in [14] and reviewed in [11]. For each activity type c, it uses the global historical mean $\bar{\tau}_c$ and global variance $\bar{\sigma}_c^2$. Unlike the proposed forecast strategy, it does not leverage contextual features x_i to condition the duration estimates.

$$\tau_i^{\text{UCC}} = \bar{\tau}_{\text{type}(i)} \tag{34}$$

4. The forecast duration strategy leverages the machine learning models described in Section 4:

$$\tau_i^{\text{forecast}} = f(x_i; \theta) \tag{35}$$

   where $f(\cdot;\theta)$ represents the trained XGBoost model with dual architecture from Eq. (24), and x_i encompasses temporal, geographic, technical, and historical features.

The increased accuracy of the duration predictions directly influences the quality of routing solutions generated by Algorithm 1.

Each VRP instance is solved using the multi-objective evolutionary algorithm with the following configuration aligned with the population P_t and parameters in Algorithm 1:

- Population size $\mu = 100$ individuals.
- Termination criteria: maximum 100 generations or 1200 s.
- Tournament selection with size $\tau = 5$.
- Crossover probability $p_c = 0.8$ using *Order Mutation* crossover.
- Adaptive parameter control: implemented via the *Compass* algorithm [15] with a lookback window of $\tau_{lookback} = 5$ generations to balance quality and diversity.
- Local search (repair): Stochastic first-improvement moves (relocate, swap, 2-opt) targeting up to $N = 10$ successful improvements per individual in each generation.
- Elite preservation: top 10% of solutions are carried over to the next generation.

For practical deployment in utility workforce management, the optimization framework must operate within tight time constraints to allow for overnight or early-morning route generation.

The computational burden is divided into two phases. The training of the Dual Weighted XGBoost architecture is a one-time cost incurred only periodically (e.g., monthly) and does not affect daily planning latency. The inference time to generate duration predictions and variance estimates for a daily batch of interventions is negligible. The primary computational cost lies in the evolutionary optimization.

To ensure the system remains tractable for daily operations, we imposed a strict termination criterion of 1200 s (20min) per instance. Empirical observations during our experimental campaign confirm that the NSGA-III algorithm consistently converges to a stable Pareto front within this time budget for instances of typical size (50–100 tasks). This runtime demonstrates that the forecast-integrated approach is fully compatible with standard utility planning windows.

6.1. Case studies performance analysis

To evaluate the practical impact of forecast-based duration prediction on VRP optimization performance, we conduct a comprehensive analysis across representative operational scenarios spanning multiple years and diverse operational regimes. We also conduct

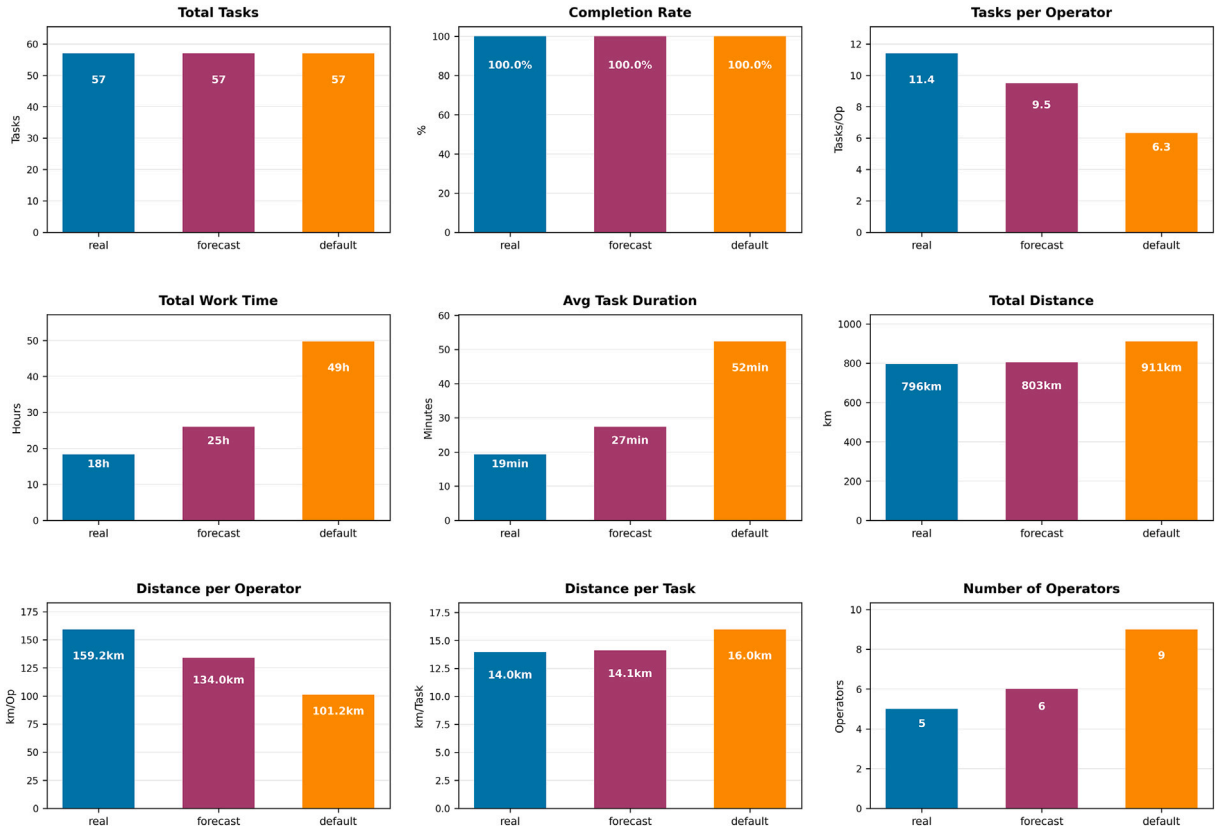


Fig. 12. KPI comparison for a representative daily operational scenario.

a systematic analysis for other periods, achieving stable and similar performance characteristics which we intentionally choose not to report. Here, we present only one daily scenario evaluation in Fig. 12, and one monthly aggregation study in Fig. 13.

In the operational simulations presented below, we fixed the risk parameter at $\alpha_k = 0.05$ for all vehicles, ensuring that routes are feasible with at least 95% probability. This value was selected following preliminary testing on a subset of historical instances, where it demonstrated the most effective trade-off between schedule reliability and resource utilization. Lower values of α (e.g., 0.01) resulted in excessive conservatism and increased vehicle counts, while higher values (e.g., 0.10) introduced unacceptable risks of overtime violations.

6.1.1. *Daily operational scenario*

We analyze a representative daily scenario that illustrates distinct operational challenges encountered in real-world gas meter service operations. The multi-objective VRP formulations seek trade-offs between competing objectives (minimize route time, minimize operator count, maximize task completion), defining a Pareto front of optimal solutions. Default duration inputs distort this objective landscape, causing the NSGA-III evolutionary algorithm to converge toward Pareto-suboptimal solutions under true durations.

When comparing operational scenarios, planning with default durations consistently demonstrates inefficiency across multiple dates. The 'Default' approach typically requires significantly more operators than are actually needed in execution, resulting in lower productivity and resource utilization. Fig. 12 illustrates this pattern clearly.

To rigorously benchmark the proposed framework against established stochastic optimization methods, we compare the final routing solutions across four strategies: Real (oracle), Default (deterministic baseline), UCC (unconditional stochastic baseline), and Forecast (proposed data-driven approach). Table 4 summarizes the performance across three representative instances.

The results in Table 4 demonstrate that while the UCC strategy (incorporating global historical variance) provides a measurable improvement over the deterministic default, it remains suboptimal compared to the ML-informed Forecast strategy. The integration of contextual predictors allows the optimization routine to allocate resources more precisely, achieving 20–25% reduction in operator requirements and total route time while maintaining near-perfect task completion reliability.

6.1.2. *Monthly aggregation analysis*

While daily scenario analysis demonstrates forecast superiority in individual operational instances, deployment viability requires consistent performance across extended temporal horizons. We present monthly aggregation analysis spanning multiple operational

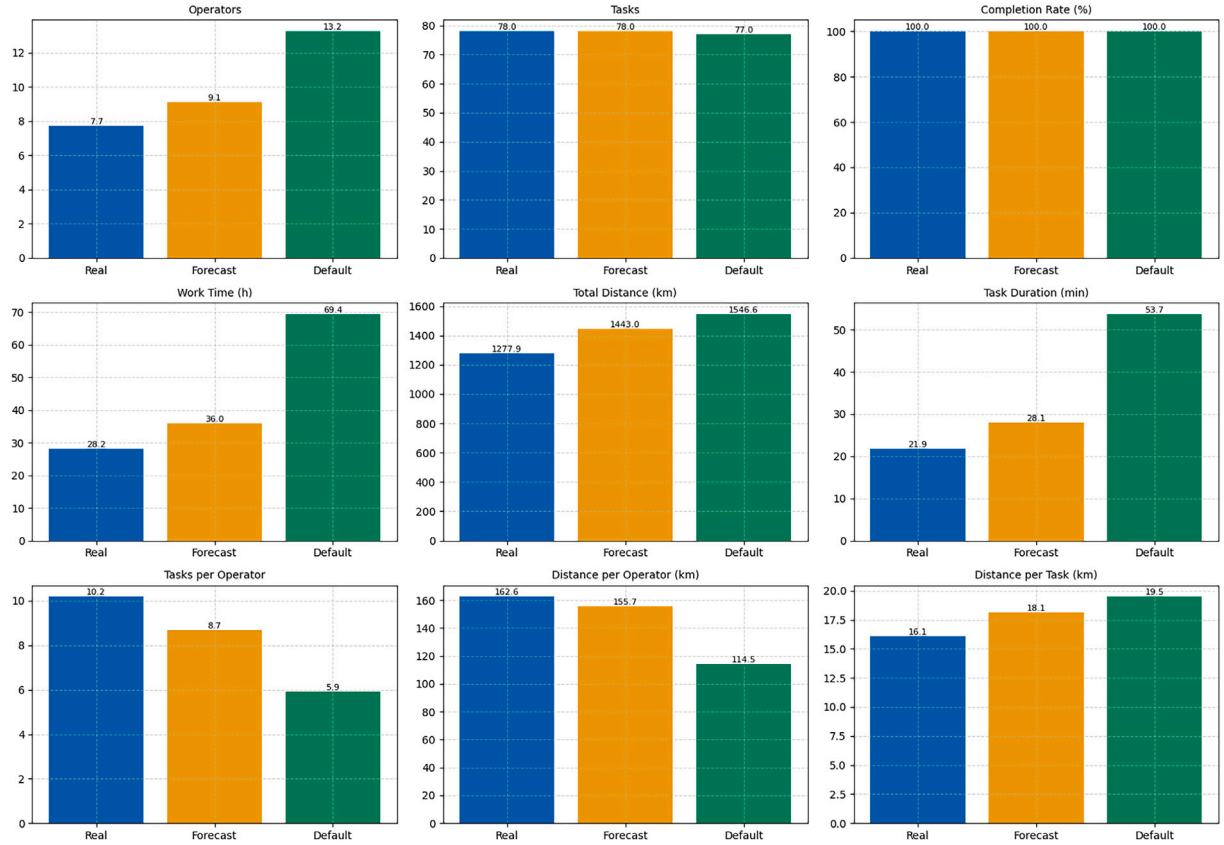


Fig. 13. KPI summary for a representative month (21 operational days, 2134 total interventions).

Table 4
Comparison of Routing Performance.

Instance	Strategy	Operators (#)	Total Time (min)	Completion (%)
Sep 4, 2020	Real	4	1200	100.0%
	Default	10	4140	100.0%
	UCC (Benchmark)	8	3300	100.0%
	Forecast	**6**	**2460**	**100.0%**
May 5, 2021	Real	5	1080	100.0%
	Default	9	2940	100.0%
	UCC (Benchmark)	8	2220	100.0%
	Forecast	**6**	**1500**	**100.0%**
Mar 21, 2022	Real	8	1500	100.0%
	Default	16	5040	100.0%
	UCC (Benchmark)	13	3900	100.0%
	Forecast	**11**	**2760**	**100.0%**

periods of 1 month to assess the temporal stability of forecast strategies. Table 5 reports the mean daily results for three representative months.

The systematic analysis across multiple operational scenarios quantifies the advantages of forecast-based VRP planning. The forecast strategy delivers reliable 20–25% improvements across multiple operational dimensions while maintaining near-optimal task completion reliability.

The systematic analysis across multiple operational scenarios quantifies the advantages of forecast-based VRP planning. The forecast strategy delivers reliable 20–25% improvements across multiple operational dimensions while maintaining near-optimal task completion reliability.

Table 5
Monthly KPI Summary (Mean Daily Values).

Period	Strategy	Operators (#)	Work Time (h)	Distance (km)
Sep 2020	Real	4.0	21.6	301.3
	Default	9.4	64.0	455.3
	UCC (Benchmark)	7.5	50.0	410.0
	Forecast	**5.9**	**36.7**	**378.3**
May 2021	Real	7.7	28.2	1277.9
	Default	13.2	69.4	1546.6
	UCC (Benchmark)	11.0	52.0	1490.0
	Forecast	**9.1**	**36.0**	**1443.0**
Mar 2022	Real	10.1	32.4	1542.1
	Default	18.2	85.1	2101.5
	UCC (Benchmark)	15.5	66.0	1950.0
	Forecast	**13.4**	**48.7**	**1821.4**

7. Conclusion

This paper presents a comprehensive framework integrating machine learning forecasts of intervention durations into stochastic vehicle routing optimization for gas meter maintenance operations. By combining tree-based gradient boosting (XGBoost) with multi-objective evolutionary algorithms (NSGA-III), we demonstrate that forecast-driven planning achieves 20–25% improvements in vehicle utilization and task completion rates compared to default duration estimates and unconditional stochastic baselines.

Our main contributions are threefold. First, we formalize the stochastic CVRPTW with chance constraints and develop mathematically rigorous risk-handling mechanisms using sub-Gaussian concentration bounds, with empirical validation of the distributional assumptions. Second, we design specialized dual-weighted model architectures that effectively capture the semantic heterogeneity of Type-Z interventions (meter replacements) while handling class imbalance through frequency weighting. Experimental results on eight years of operational data (884,349 interventions) validate the approach. Third, we integrate uncertainty quantification into the multi-objective VRP solver (Algorithm 1), enabling route-level risk buffers that balance service coverage against overtime constraints through comparison with both deterministic and standard stochastic baselines.

Acknowledgements

This research was conducted within the Green Inspired Revolution for Optimal Workforce Management (GIRO-WM) project, funded by Fondazione Caritro, CUP B93C22002160003.

We would like to thank teams at HPA and Terranova Software for providing the real-world data and proposing the practical problem setting that formed the basis of our study.

Appendix A. Feature importance rankings

This appendix provides comprehensive feature importance analysis for all model configurations, complementing the comparative analysis presented in Section 4.

A.1. Feature importance interpretation

Feature importance rankings quantify the contribution of each input feature to the predictive performance of gradient boosting models. The gain metric used throughout this analysis measures the average improvement in loss function reduction (equivalently, prediction accuracy enhancement) attributable to each feature across all decision tree splits in the ensemble. Formally, for feature j, the gain importance is computed as:

$$\text{Gain}_j = \frac{1}{M} \sum_{m=1}^{M} \sum_{s \in S_m^{(j)}} \Delta L_s$$

where M denotes the number of trees in the ensemble, $S_m^{(j)}$ represents the set of splits in tree m that use feature j, and ΔL_s quantifies the loss reduction achieved by split s.

High gain importance indicates that a feature frequently appears in decision trees at positions where it enables significant reduction in prediction error. Features with low importance contribute minimally to model predictions and could potentially be removed without substantial performance degradation. The importance rankings inform feature engineering priorities for future model refinements and provide insight into the operational factors governing intervention duration variability.

A.2. Standard model feature importance

The standard gradient boosting model (Model 1) exhibits feature importance patterns reflecting the dominant operational drivers of intervention durations. As shown in Fig. 14, temporal cyclical features (hour-of-day sine/cosine encodings, day-of-week indicators) account for approximately 65% of cumulative feature importance, confirming that time-based patterns constitute the primary predictive signal.

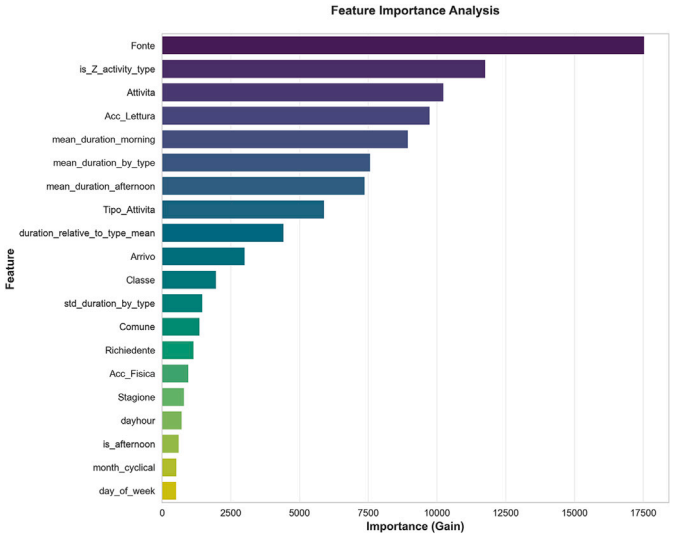


Fig. 14. Feature importance ranking for model 1 (`Standard Gradient Boosting`). Temporal cyclical features dominate the prediction landscape, with hour-of-day sine/cosine encodings and day-of-week patterns accounting for the highest gain contributions. Activity type classification and geographic attributes (municipality characteristics, altitude) provide secondary predictive power. The balanced importance distribution across multiple feature families indicates that the standard model effectively leverages diverse information sources to capture duration variability.

This temporal dominance aligns with operational realities: interventions scheduled during morning hours (8 – 11 AM) typically exhibit shorter durations due to optimal traffic conditions and technician alertness, while afternoon interventions (2 – 5 PM) show increased variability. Day-of-week effects capture systematic duration differences between weekdays (routine scheduling, experienced operators) and weekends (emergency interventions, reduced support staff availability).

Activity type classifications contribute approximately 20% of cumulative importance, with geographic features (municipality characteristics, altitude, urbanization degree) and equipment attributes (meter class, protocol) providing the remaining 15%. This importance hierarchy validates the feature engineering strategy, demonstrating that multiscale temporal patterns and operational context drive duration predictions in standard architectures.

A.3. Dual weighted model feature importance

The `Dual Weighted` architecture (Model 4) exhibits substantially different feature importance distributions compared to the standard model, reflecting its specialized design. As illustrated in Fig. 15, temporal features maintain importance but are complemented by an enhanced representation of activity-specific and equipment-related attributes.

The architectural specialization (separate models for Type-Z vs. other interventions) enables feature importance patterns to adapt to specific operational contexts. Within the Type-Z specialized model, equipment attributes (meter class, protocol complexity) gain prominence as they directly influence the compositional nature of meter replacement operations. Within the other-interventions model, temporal and geographic features dominate as routine operations exhibit stronger time-based regularities.

The frequency weighting mechanism amplifies importance for features discriminating rare activity types that would be underweighted in standard training. This redistribution ensures that the model allocates learning capacity proportionally to operational complexity rather than sample frequency, resulting in superior performance on underrepresented intervention categories.

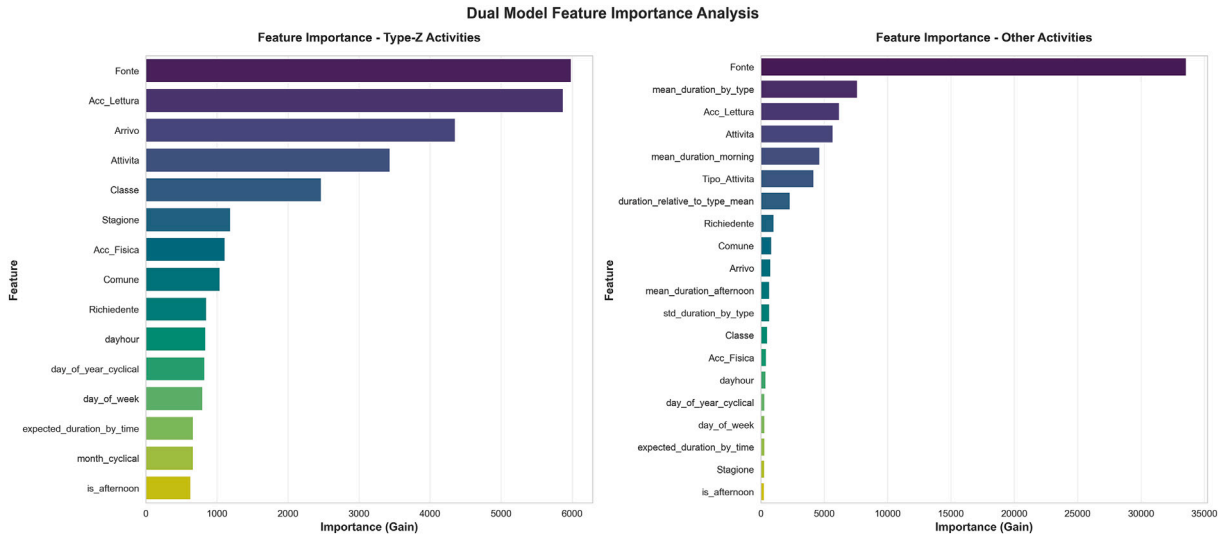


Fig. 15. Feature importance ranking for model 4 (`Dual Weighted`). The specialized dual architecture exhibits distinct importance patterns compared to the standard model. While temporal features remain influential, activity type classifications gain prominence due to the model's explicit architectural separation between Type-Z and other interventions. The frequency weighting mechanism amplifies the importance of features discriminating rare activity types, resulting in enhanced representation of meter-specific attributes (class, protocol) that would be underweighted in standard training. This importance redistribution reflects the model's ability to capture heterogeneous operational dynamics across different intervention categories.

Appendix B. Activity duration statistics

The following distributional analysis reveals a significant imbalance between activity types: categories `E`, `F`, and `Z` account for over 600,000 combined interventions. Rare categories (`I`, `W`) with fewer than 20 examples have not been considered in the training. However, normalized duration distributions exhibit similar statistical properties when proportionally scaled.

In Table 6, we report a quantitative description of the intervention types.

Table 6 provides comprehensive descriptive statistics for intervention durations by activity type. Fig. 16 presents the distribution of work durations across different activity types, while Fig. 17 shows the normalized binned distributions to highlight pattern similarities despite frequency differences.

Table 6
Descriptive statistics of intervention durations by activity type (in minutes).

Code	Activity Type	Count	Mean	Std. Dev.	Min	25%	50%	75%	Max
A	Work quotation	10,319	55.79	21.71	10	41	60	69	119
C	Work execution	5112	32.55	28.04	10	15	20	35	119
E	Supply activation	255,288	24.57	10.41	10	18	24	30	119
F	Supply deactivation	131,657	19.01	9.12	10	14	17	21	118
H	Metrological verification	1550	45.59	20.36	10	31	40	55	119
I	Pressure verification	4	80.00	17.94	54	76.5	85.5	89	95
J	Meter reading	1676	45.53	22.34	10	30	45	60	119
L	Reading request	1261	18.42	10.38	10	11	15	21	106
M	Closure for arrears	38,744	20.75	10.96	10	15	19	24	119
N	Meter closure	137	24.39	15.08	10	15	20	30	98
O	Meter reading	50,487	18.33	12.64	10	10	14	20	118
Q	Removal of sensors	1605	25.09	13.72	10	15	21	30	114
R	Meter removal	6159	23.40	16.63	10	14	18	27	119
S	Safety shutdown	360	26.85	19.82	10	15	20	30	105
T	Data provision	15,972	23.46	9.33	10	18.75	23	26	119
W	Administrative termination	11	25.36	12.73	14	17.5	22	27	60
X	Interruption for arrears	705	31.56	20.63	10	17	26	39	117
Z	Meter replacement	233,143	26.73	19.94	10	14	21	29	119

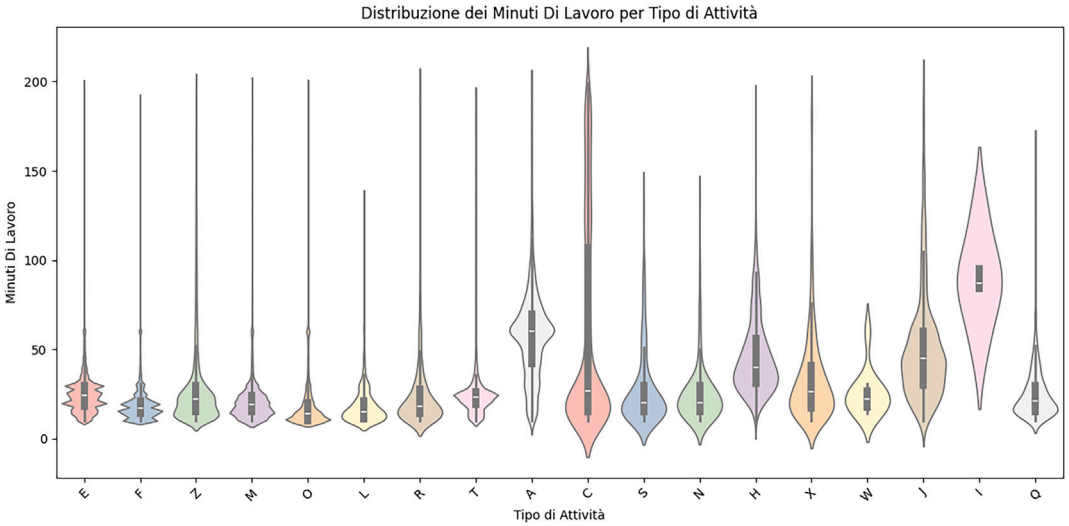


Fig. 16. Distribution of work durations (minutes) by activity type.

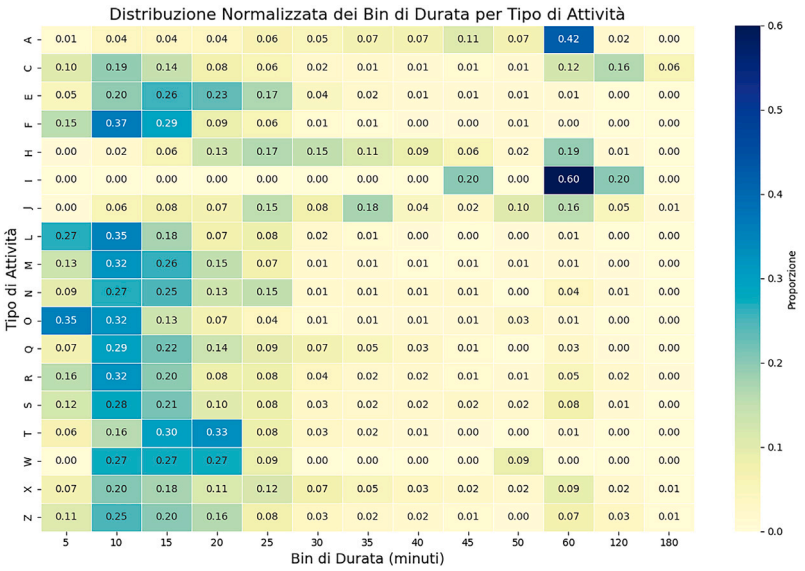


Fig. 17. Normalized binned distribution of intervention durations by activity type.

Data availability

Due to confidentiality agreements with the industrial partners, the operational dataset cannot be publicly released. However, synthetic data generation procedures or anonymized subsets may be made available upon reasonable request.

References

[1] M. Rajabi-Bahaabadi, A. Shariat-Mohaymany, M. Babaei, Reliable vehicle routing problem in stochastic networks with correlated travel times, Operations Research International Journal 21 (2021) 299–330, https://doi.org/10.1007/s12351-019-00452-w

[2] R. Bai, X. Chen, Z.L. Chen, T. Cui, S. Gong, W. He, H. Zhang, Analytics and machine learning in vehicle routing research, Int. J. Prod. Res. 61 (1) (2021) 4–30, https://doi.org/10.1080/00207543.2021.2013566

[3] A. Bogyrbayeva, M. Meraliyev, T. Mustakhov, B. Dauletbayev, Machine learning to solve vehicle routing problems: a survey, IEEE Trans. Intell. Transp. Syst. 25 (6) (2024) 4754–4772, https://doi.org/10.1109/TITS.2023.3334976

[4] F. Bomboi, C. Buchheim, J. Pruente, *On the stochastic vehicle routing problem with time windows, correlated travel times, and time dependency*, 4OR, A Quarterly Journal of Operations Research 20 (2022) 217–239, https://doi.org/10.1007/s10288-021-00476-z

[5] K. Chang, T. Kim, Pre-route timing prediction and optimization with graph neural network models, Integration 99 (2024) 102262, https://doi.org/10.1016/j.vlsi.2024.102262

[6] U. Sadana, A. Chenreddy, E. Delage, A. Forel, E. Frejinger, T. Vidal, A survey of contextual optimization methods for decision-making under uncertainty, Eur. J. Oper. Res. 320 (2) (2025) 271–289, https://doi.org/10.1016/j.ejor.2024.03.020

[7] G.B. Dantzig, J.H. Ramser, The truck dispatching problem, Manag. Sci. 6 (1) (1959) 80–91.

[8] E. Delage, Y. Ye, Distributionally robust optimization under moment uncertainty with application to data-driven problems, Oper. Res. 58 (3) (2010) 595–612, https://doi.org/10.1287/opre.1090.0741
[9] A.M. Florio, M. Gendreau, R.F. Hartl, S. Minner, T. Vidal, Recent advances in vehicle routing with stochastic demands: Bayesian learning for correlated demands and elementary branch-price-and-cut, Eur. J. Oper. Res. 306 (3) (2023) 1081–1093, https://doi.org/10.1016/j.ejor.2022.10.045
[10] Q. Guan, H. Cao, L. Jia, D. Yan, B. Chen, Synergetic attention-driven transformer: a deep reinforcement learning approach for vehicle routing problems, Expert Syst. Appl. 274 (2025) 126961, https://doi.org/10.1016/j.eswa.2025.126961
[11] M. Gendreau, G. Laporte, R. Séguin, Stochastic vehicle routing, Eur. J. Oper. Res. 88 (1) (1996) 3–12, https://doi.org/10.1016/0377-2217(95)00247-2
[12] B. Gülmez, M. Emmerich, Y. Fan, Multi-objective optimization for green delivery routing problems with flexible time windows, Appl. Artif. Intell. 38 (1) (2024), https://doi.org/10.1080/08839514.2024.2325302
[13] Z. Iklassov, I. Sobirov, R. Solozabal, M. Takáč, Reinforcement learning approach to stochastic vehicle routing problem with correlated demands, IEEE Access 11 (2023) 87958–87969, https://doi.org/10.1109/ACCESS.2023.3306076
[14] G. Laporte, F.V. Louveaux, H. Mercure, The vehicle routing problem with stochastic travel times, Transp. Sci. 26 (3) (1992) 161–170, https://doi.org/10.1287/trsc.26.3.161
[15] J. Maturana, F. Saubion, *A compass to guide genetic algorithms*, in parallel problem solving from nature – PPSN X, in: Lecture Notes in Computer Science, vol. 5199, Springer, Berlin, Heidelberg, 2008, pp. 256–268, https://doi.org/10.1007/978-3-540-87700-4_26
[16] H. Jain, K. Deb, An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, Part I: Solving problems with box constraints, IEEE Trans. Evol. Comput. 18 (4) (2014) 577–601, https://doi.org/10.1109/TEVC.2013.2281535
[17] H. Jain, K. Deb, An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, Part II: Handling constraints and extending to an adaptive approach, IEEE Trans. Evol. Comput. 18 (4) (2014) 602–622, https://doi.org/10.1109/TEVC.2013.2281534
[18] M. Nazari, A. Oroojlooy, L. Snyder, M. Takac, Reinforcement learning for solving the vehicle routing problem, In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, R. Garnett (Eds.), Advances in Neural Information Processing Systems, vol. 31, Curran Associates, Inc, 2018, pp. 9839–9849.
[19] J. Oyola, H. Arntzen, D.L. Woodruff, The stochastic vehicle routing problem, a literature review, Part I: Models, EURO Journal on Transportation and Logistics (2016), https://doi.org/10.1007/s13676-016-0100-5
[20] J. Oyola, H. Arntzen, D.L. Woodruff, The stochastic vehicle routing problem, a literature review, Part II: Solution methods, EURO Journal on Transportation and Logistics 6 (2017) 349–388, https://doi.org/10.1007/s13676-016-0099-7
[21] W. Pan, S. Liu, Deep reinforcement learning for the dynamic and uncertain vehicle routing problem, Appl. Intell. 53 (2023) 405–422, https://doi.org/10.1007/s10489-022-03456-w
[22] S.M. Raza, M. Sajid, J. Singh, *Vehicle routing problem using reinforcement learning: recent advancements*, in advanced machine intelligence and signal processing, In D. Gupta, K. Sambyo, M. Prasad, S. Agarwal (Eds.), Lecture Notes in Electrical Engineering, vol. 858, Springer, Singapore, 2022, pp. 255–272, https://doi.org/10.1007/978-981-19-0840-8_20
[23] S. Rezvanian, A.H. Kashan, A. Rezvanian, A. Sabzevari, Intelligent vehicle routing for stochastic service times: a grouping evolution strategy approach, International Journal of Transportation Science and Technology (2025), https://doi.org/10.1016/j.ijtst.2025.04.009
[24] N.V. Sahinidis, Optimization under uncertainty: state-of-the-art and opportunities, Comput. Chem. Eng. 28 (6–7) (2004) 971–983, https://doi.org/10.1016/j.compchemeng.2003.09.017
[25] B. Serrano, A.M. Florio, S. Minner, M. Schiffer, T. Vidal, Contextual stochastic vehicle routing with time windows, arXiv e-prints arXiv:2402.06968, 2024.
[26] R. Shahbazian, L.D.P. Pugliese, F. Guerriero, G. Macrina, Integrating machine learning into vehicle routing problem: methods and applications, IEEE Access 12 (2024) 93087–93115, https://doi.org/10.1109/ACCESS.2024.3422479
[27] A. Shapiro, Stochastic programming approach to optimization under uncertainty, Math. Program. 112 (2008) 183–220, https://doi.org/10.1007/s10107-006-0090-4
[28] N. Soeffker, M.W. Ulmer, D.C. Mattfeld, Stochastic dynamic vehicle routing in the light of prescriptive analytics: a review, Eur. J. Oper. Res. 298 (3) (2022) 801–820, https://doi.org/10.1016/j.ejor.2021.07.014