Jialin Liu, Ke Tang, and Xin Yao
Southern University of Science and Technology, CHINA

Robust Optimization in Uncertain Capacitated Arc Routing Problems: Progresses and Perspectives

Abstract

The capacitated arc routing problem is an important NP-hard problem with numerous real-world applications. The capacitated arc routing problem with uncertainties refers to those instances where there are uncertainties in decision variables, objective functions and/or constraints. The capacitated arc routing problem with uncertainties captures real-world situations much better than a static capacitated arc routing problem because few real-world problems are static and certain. Uncertainties in the capacitated arc routing problem pose new research challenges. Algorithms that work well for a static and certain capacitated arc routing problem may not work on the version with uncertainties. There have been increasing progresses in studying the capacitated arc routing problem with uncertainties during the past two decades. However, the papers on the capacitated arc routing problem with uncertainties have been scattered around in different journals and conferences in artificial intelligence, computer science, and operational research. Different definitions and formulations of capacitated arc routing problem with uncertainties are used by different papers, making comparisons difficult. In order to better understand the state-of-the-art in solving the capacitated arc routing problem with uncertainties, this paper presents a comprehensive review of the problem and its key research issues. Not only has the paper summarized the progresses so far, key research issues are identified, including scalability of the algorithms, performance measures, common benchmarks, etc. Future research directions are also identified at the end of this review.

Digital Object Identifier 10.1109/MCI.2020.3039069
Date of current version: 12 January 2021

I. Introduction

The capacitated arc routing problem (CARP) [1], [2] is a NP-hard combinatorial problem with numerous real-world applications. Examples include the urban waste collection, snow removal and street salting problems [3], [4]. The objective of the CARP is to efficiently allocate a number of vehicles with limited capacities and select the optimal set of routes from a depot to serve a number of tasks while the total demand of tasks served on any route does not exceed the vehicle capacity. Different from the vehicle routing problem (VRP) [5], the tasks in CARP are located on the edges instead of vertices. However, a good understanding of CARP would benefit the study of VRP.

Motivated by the characteristics of different real-world applications, numerous variants of CARPs have been proposed, which differ in terms of vehicle capacity, demand, and service time windows [2], [4], among which most of the

Corresponding author: Xin Yao (xiny@sustech.edu.cn).

work assume that no uncertainty is involved in the CARP instances. However, uncertainty is everywhere in real life and makes the solutions optimized on deterministic CARP instances non-optimal or even infeasible under the uncertain environment. For instance, in the snow removal and street salting problems [3], the amount of snow to be removed from a street and the amount of salt to be put on a street depend on the weather and are non-deterministic. In the urban waste collection problem [4], the amount of waste to be collected on a street depends on the daily consumption of food or goods, which is uncertain.

Stochastic CARP (abbreviated to "SCARP"), first proposed by Fleury *et al.* [6] in 2002, assumes the stochasticity of demands in the urban waste collection problem while all the other variables remain deterministic. Such problems have also been referred to as the CARP with stochastic demands (CARPSD) in some papers [7], [8]. Besides stochastic demands, stochastic costs are commonly observed in real life as well. In addition, an edge could be broken, thus not be present, due to heavy traffic or road maintenance. Uncertain CARP is then defined by [9], considering all the uncertainties mentioned above. Uncertain CARP [9] is abbreviated to "uCARP" in this paper to distinguish it from the general CARP with uncertainties (UCARP), which include all the possible uncertainties in decision variables, objectives functions and/or constraints.

There have been many variants of the UCARP in the past two decades. Most papers on the UCARP focus on the robust evaluation of solutions optimized on a set of deterministic scenarios of the UCARP (Section V-B1). Only a few of them design algorithms for tackling the UCARP directly (Section V-B2). However, all such work assumes the knowledge of variable distributions (Section III-C). In real-world applications, such knowledge is often unavailable; instead, only a finite set of deterministic realizations of a UCARP is accessible [10]. As a result, scenario-based robust optimization in the UCARP without assuming known distributions of variables has been investigated (Section V-C). Very recently, machine learning techniques were applied to design routing heuristics for the UCARP (Section V-D).

In spite of much work on the deterministic CARP, the UCARP has introduced new research challenges that require further study. Many variants of the UCARP have been proposed in the literature, motivated by different real-world applications. It is unclear whether these variants are fundamentally different or whether they pose different research questions. Papers on the UCARP have been published in a diverse set of journals and conferences. To our best knowledge, there has been no comprehensive review on different variants of UCARP and the state-of-the-art solutions to them. It is unclear what the key research challenges are for different UCARP variants and what future directions might be.

S. Wøhlk [2] reviewed the research on the CARP and its variations published till 2007, when the studies on stochastic CARP were very limited [6], [11]–[13]. Fadzli *et al.* [4] surveyed the applications of CARP and its extensions to waste collection problem published till 2011. There is no dedicated review of UCARP. The main objectives of this paper include reviewing different problem definitions and assumptions of the UCARP, the performance measures and approaches used for solving the problem instances, discussing the key research challenges, and pointing out possible future research directions.

The remainder of this paper is organized as follows. Section II clarifies the scope of this review and the methodology used. Section III formalizes the deterministic CARP and reviews different variants of CARP with uncertainties. Section IV introduces the reliability and robustness metrics used for evaluating solutions. We present the techniques for handling uncertainties, solving CARP with uncertainties and designing routing policies in Section V. Section VI discusses in depth the most important research issues related to the CARP with uncertainties. Finally, Section VII concludes the paper and points out some future research directions.

II. Scope of the Review

To review the related work published till January 2020, search has been conducted in the databases of the following main publishers, *IEEE Xplore*, *ACM*, *Springer* and *Elsevier*, using the search term: *("capacitated arc routing problem") AND ("stochastic" OR "uncertain" OR "random").* We then searched again with *Google Scholar* and *Web of Science* using the same search terms listed above.

Among the returned results, the papers are carefully screened as follows. (i) If a paper does not work on CARPs (this happens as the search term may appear in references or in the text body due to a citation to another work), then it's not included in this survey. (ii) If a paper considers deterministic CARPs, then no uncertainty has been involved in the problem variables (presence of task, presence of edge, demand of task, cost of traversing edges, service cost, vehicle capacity, etc.), and it is therefore not included in this survey. It is notable that if the above keywords do not appear explicitly in the title or body of a publication, then the publication is not included in this survey. The study on the *dynamic and deterministic CARP* (e.g., the CARP with period- or time-dependent demands or costs) where the exact value of demand is computable and no uncertainty is involved, is not included in this review. More discussions on the comparison of dynamic and stochastic CARP will be provided in Section III-A2). The second column of Table V lists the articles on CARP with uncertainties that have been reviewed in this paper.

III. Capacitated arc routing problem with uncertainties

In order to facilitate the description and understanding of the problems, we formalize the static and deterministic version of capacitated arc routing problems (CARPs) in Section III-A1). A taxonomy of CARPs, similar to [14], is provided in Section III-A2) to distinguish between dynamics and randomness in the CARP. Then, Section III-B briefly reviews CARPs with uncertainties. The modelling of uncertainties is described in Section III-C. Section III-D presents the benchmarks of problem instances used in the reviewed literature.

A. Capacitated Arc Routing Problem

The basic form of CARP [1], [5] can be described as follows. Let $G=(V,E)$ be an undirected graph where V and E denote the sets of vertices and edges, respectively. The vertex $v_0 \in V$ is the depot. Each edge $e \in E$ has a cost $c(e) > 0$. If there is a task on this edge, a positive demand is associated to e. The task set T is the set of edges with positive demands. A fleet of vehicles with a given capacity $Q > 0$ are allocated to serve all the tasks in T, starting and terminating at the identical depot v_0. The objective of CARP is to efficiently allocate these vehicles and select the optimal set of routes to serve tasks while the total demand of tasks served on any route does not exceed Q.

1) General Formulation of CARP

Diverse extensions of CARP to the above have been proposed depending on the corresponding applications in the real-world, such as the CARP with stochastic time and periodic CARP [2], the CARP with multiple depot [15] and the CARP with multiple vehicle capacities [16]. Their different formulations are outside the scope of this review as we focus on the CARP with uncertainties.

A solution of CARP can be represented by a set of routes $\boldsymbol{x}=\{r_1,\ldots,r_m\}$ served by m vehicles. Each route $r_k=(t_{k,1},\ldots,t_{k,l_k})$ $(k\in\{1,\ldots,m\})$ is a sequence of tasks served, where l_k is the number of tasks served on this route and $t_{k,i}$ refers to the ith task on the kth route. We define the general formulation used in [17] as follows, given $G=(V,E)$, T, Q, cost $c(t_{k,i})$ and demand $d(t_{k,i})$ of $t_{k,i}$,

$$\min C(\boldsymbol{x}) = \sum_{k=1}^{m}\left(\sum_{i=1}^{l_k}\left(\underbrace{c(t_{k,i})}_{\text{serving cost}} + \underbrace{dist(tail(t_{k,i-1}), head(t_{k,i}))}_{\text{deadheading cost}}\right) + \underbrace{dist(tail(t_{k,l_k}), v_0)}_{\text{deadheading cost from last task to depot}}\right), \quad (1)$$

$$\text{s.t.} \quad t_{k,i}\in T,\ \forall k\in\{1,\ldots,m\}, \quad i\in\{1,\ldots,l_k\}, \quad (2)$$

$$head(t_{k,1}) = tail(t_{k,l_k}) = v_0, \quad \forall k\in\{1,\ldots,m\}, \quad (3)$$

$$t_{k,i}\neq t_{k',i'},\ \forall (k,i)\neq(k',i'), \quad (4)$$

$$t_{k,i}\neq inv(t_{k',i'}),\ \forall (k,i)\neq(k',i'), \quad (5)$$

$$\sum_{k=1}^{m} l_k = |T|, \quad (6)$$

$$\sum_{i=1}^{l_k} d(t_{k,i}) \leq Q,\ \forall k\in\{1,\ldots,m\}, \quad (7)$$

where $head(t_{k,i})$ and $tail(t_{k,i})$ denote the two endpoints of the task $t_{k,i}$. When $i=1$, $tail(t_{k,0})$ is defined as the depot v_0. $inv(t_{k,i})$ denotes the reverse direction of $t_{k,i}$. Besides the *serving cost* $c(t)$ of each task t, the *deadheading cost*, defined as the cost of traversing the shortest path between a pair of tasks, is often considered as well. $dist(v_i, v_j)$ denotes the shortest distance from vertex v_i to vertex v_j. Eq. (1) indicates the minimization of $C(\boldsymbol{x})$, the sum of the serving and deadheading costs of the set of routes $\boldsymbol{x}$. The domain of variables is defined in Eq. (2). Eq. (3) indicates that each route should start and end at the depot. Eqs. (4) and (5) ensure that each task will be served in one direction only, i.e., $(k=k')$ and $(i=i')$ won't hold simultaneously. Together with Eq. (6), Eqs. (4) and (5) further ensure that each task will be served exactly once. The constraint of vehicle capacity is implied in Eq. (7).

2) Taxonomy of CARP

Similar to the taxonomy of VRP [14], we propose a taxonomy of CARPs as follows:

1) static and deterministic CARP;
2) dynamic and deterministic CARP;
3) static and stochastic CARP;
4) dynamic and stochastic CARP.

A CARP instance can be *dynamic*, i.e., the variables are unknown a priori and may change over time; otherwise, it is *static*. A *deterministic* CARP assumes no random element involved in the problem, thus the variables are all deterministic. Only static input (e.g., edges, tasks and demands) is considered in the *static and deterministic* CARP, while in the *dynamic and deterministic* version, the input changes over time, such as an increasing travelling cost because of the increasing vehicle load. The *stochastic* CARP or CARP with uncertainties, in general, considers that the input variables are random and their exact values are only known at the time of execution. The use of terms *dynamic* and *stochastic* is sometimes ambiguous. Table I lists some of the main differences between the *static and stochastic* CARP and the *dynamic and stochastic* version.

In this paper, we focus on the CARP with uncertainties, either static or dynamic. Studies on dynamic and deterministic CARP are out of the scope of this paper. From the literature, we have observed that all review studies so far aimed at solving static and stochastic CARP, most of which were based on methods for solving static and deterministic CARP. Few treated a stochastic CARP as a stochastic problem. To focus

TABLE I Comparison between *static and stochastic* CARP and *dynamic and stochastic* CARP.

	STATIC AND STOCHASTIC CARP	DYNAMIC AND STOCHASTIC CARP
PLANNING	A PRIORI SOLUTION IS COMPUTED BY OFFLINE PLANNING	ONLINE PLANNING
	ONLINE REPAIRING IS APPLIED ACCORDING TO ACTUAL VARIABLE VALUES	RE-OPTIMIZATION WHILE TRAVELLING IN REAL TIME
TASKS/ CUSTOMERS	ALL POTENTIAL CUSTOMERS ARE KNOWN IN ADVANCE, WHILE EACH HAPPENS WITH A PROBABILITY	SOME CUSTOMERS ARE KNOWN TO HAPPEN WITH A PROBABILITY NEW REQUESTS CAN BE MADE DURING EXECUTION
TIMING	SOMETIMES TIME WINDOW IS CONSIDERED	URGENCY OF TASK IS USUALLY CONSIDERED

on uncertainties in the CARP, from now on, the term "static" will be omitted when referring to *static and deterministic* or *static and stochastic CARP*.

B. Variants of CARP with Uncertainties

In the past two decades, different variations of *CARP with uncertainties* were proposed and studied considering the non-deterministic factors in real-world problems.

1) CARP with Stochastic Demands (CARPSD)

Motivated by the urban waste collection problem, Fleury *et al.* [6], [11] proposed the *stochastic CARP* (abbreviated as "SCARP") in which only the demands are stochastic, i.e., $d(\cdot)$ in Eq. (7) is a random variable rather than a constant, and follows normal distributions. Christiansen *et al.* [7] and Laporte *et al.* [8] used the abbreviation "CARPSD" for the *CARP with stochastic demands* (same as SCARP) but assumed Poisson distributed demands in their work.

2) Uncertain CARP (uCARP)

Besides stochastic demands, many other stochastic variables can be present in real-world applications, such as the stochastic costs of tasks, the absence of edges and/or tasks [3], [4]. Factors that can affect the cost of traversing a path include, but are not limited to, the speed of a vehicle, the load of a vehicle, the traffic flow and the weather. Due to heavy traffic or road maintenance, an edge can be considered broken, thus it is not present. The above can be roughly summarized into the following 4 random factors: presence of tasks, demand of tasks, presence of paths and deadheading costs. Mei *et al.* [9] proposed a more general case, uncertain CARP (uCARP), that considers all the above. In [9], Bernoulli and Gamma distributions were assumed for the stochastic variables.

However, in real life, some of the uncertainties, originating from nature or humans, cannot be modelled easily by stationary probabilities. Their distributions are unknown a priori. Taking the urban waste collection problem as an example, family parties or Christmas often result in a higher food consumption and a significant increase in the amount of waste, while zero waste collection demand occurs occasionally during spring holidays. Variables in such scenarios can hardly be modelled by fixed probability distributions. As pointed out by Wang *et al.* [10], in reality, only a set of scenarios[1] is likely to be available. Afterwards, research has been conducted on searching robust solutions for a set of deterministic samples of uCARP without assuming a priori known distributions for uncertainty [10], [17].

The exact values of random variables are known only at the time of reaching a task or edge (road or street); therefore, solutions optimized a priori might be infeasible at the time of execution. Techniques for avoiding constraint violation (summarized in Section V-A1) and repair operators for fixing solutions during execution (Section V-A2) are essential. For instance, if the amount of waste to collect exceeds the available capacity of a vehicle, the vehicle will need to return to the depot, empty its collection, then continue to serve the remained tasks. How to efficiently adapt a solution to the actual scenario while minimizing the cost and risk is an important topic for research.

3) CARP with Fuzzy Demand (CARPFD)

Instead of using probabilistic models, fuzzy numbers have been used to model demands in CARP [18], [19].

4) CARP with Stochastic Time (CARPST)

Different from the above work, Chen *et al.* [20], [21] considered the stochasticity of time in the road network daily maintenance problem. In [20], the problem is formulated as a CARP with stochastic service and travel times (CARP-SSTT) following a normal distribution, while in [21], the problem is formulated as an ARP with stochastic service time without assuming (ARP-SST) any known variable distribution.

5) Abbreviations

To be consistent with the existing publications, hereinafter, the abbreviation "DCARP" is used to refer to the deterministic CARP and "UCARP" is used to refer to the CARP with uncertainties. The abbreviation "uCARP" stands for the uncertain CARP defined by [9] considering 4 random variables. Although "SCARP" is the first abbreviation used for the CARP with stochastic demands [6], the abbreviation "CARPSD" [7] will be used as it is more informative. It is notable that in some literature, the phrase "dynamic CARP" is used as an alternative to UCARP, such as [22]. However, we argue that the use of "dynamic CARP" is not suitable here, as discussed previously in Section III-A2). "CARPFD" and "CARPST" stand for CARP with fuzzy demands and CARP with stochastic times, respectively.

C. Modelling Uncertainties

A number of work on CARP with uncertainties assumed certain distributions for non-deterministic variables. This section presents the assumptions considered in the reviewed work.

1) Cancelled/Unexpected Tasks

The presence of tasks is often modelled as a Bernoulli distribution [9]. Thus, given a set of potential tasks, the *i*th task is present with probability $p_i \in (0,1)$. The case that a task is not present can be considered as one with 0 demand. When executing a solution, if a task is no longer present, the vehicle will skip the task and travel to the next one via the shortest path. The presence of tasks will affect the set of tasks to be served, the domain of solution variables, Eqs. (6) and (2) in the model formulated in Section III-A1).

2) Edge Failure

The case in which an edge is not present is called an *edge failure*, possibly due to a broken path or heavy traffic. When an edge failure occurs, the cost of this edge is set to ∞. If there is a task on this edge, the vehicle will take it as a cancelled

[1] Different terms, such as "replication", "sample" and "scenario", have been used in the literature to refer to a deterministic realization of a random process.

task. The edge failure is also often assumed to follow a Bernoulli distribution [9]. The presence of edges is crucial for calculating the deadheading costs, which are terms of the total cost of a solution, as detailed in Section III-A1).

3) Random Demand

In real-world problems, the edge-demand is often random. For instance, in the urban waste collection problem, the amount of waste on a street is non-deterministic. In the parcel collection problem, an order could change suddenly. A random demand needs to respect the following constraints: it should not be negative or exceed the vehicle capacity Q. If the demand of a task exceeds the available capacity of a vehicle, then the vehicle fails to serve this task and a *route failure* occurs. The capacity constraint Eq. (7) formalized in Section III-A1) is violated. As a consequence, an *extra trip* (i.e., returning to the depot and then going back to the task) is required. In most of the studied stochastic models for demands, their expectations are known. The most studied models are truncated normal, Poisson and Gamma models, while another important category is the estimation of demands by sampling scenarios (e.g., [10]).

a) Uniform Distribution

In the waste collection problems considered in [37]–[39], random demands were generated using uniform distribution.

b) Truncated Normal Distribution

The urban waste collection problem has been formulated as a CARPSD and the demand of each task was assumed to be a random variable following a truncated normal distribution in [6], [11]–[13], [34], [35], ignoring the extremal case such as Christmas. The normal distribution is truncated to avoid exceeding the vehicle capacity Q and negative demand, and its standard deviation $\sigma(v)$ is set to $\alpha \cdot \overline{d(t_i)}$, where $\overline{d(t_i)}$ is the mean of the demand of the ith task t_i and α is a positive control parameter. The demand of the ith task, t_i with this *truncated*, *multiplicative* normal noise, is formalized as $D(t_i) \sim \overline{d(t_i)} + \max\{0, \mathcal{N}(0, \sigma_i^2)\}$, where $\sigma_i = \alpha \cdot \overline{d(t_i)}$ and the control parameter α is usually set to 0.1.

c) Log-Normal Distribution

Log-normal distributed stochastic demands have been used in [36].

d) Poisson Distribution

In [7], [8], the CARPSD is studied assuming that the stochastic demands follow Poisson distributions.

e) Gamma Distribution

Mei *et al.* [9] assumed that the perturbation on demands follows a Gamma distribution with shape parameter k and scale parameter θ to avoid negative noises, denoted as $G(k, \theta)$. As the random presence of tasks is also considered in [9], the demand of the ith task, t_i, is modelled as $D(t_i) \sim G(k_i, \theta_i)$ if $rand < p_i$, otherwise, $D(t_i) = 0$. The probability of the presence of task i is denoted as p_i and is set as 0.9 in [9]. The shape parameter is set as $k_i = 20$ for all the tasks to make the Gamma distribution close to normal distribution and the scale parameter θ_i is set to $d_i / p_i k_i$ so that the expected value of the random demands is equal to its static value [9]. A number of work, summarized in Table II, have used the UCARP instances designed by [9].

f) Fuzzy Demand

Eydi and Javazi [18] studied multi-commodity CARP and represented the demand of every commodity on a serving edge as a triangular fuzzy number [18]. Babaee Tirkolaee *et al.* [19] formulated an urban solid waste management problem as the multi-trip CARPSD under fuzzy demands.

4) Random Deadheading Cost

Due to the speed limit, condition of roads, traffic and load, the deadheading costs are often non-deterministic. This can lead to different values of total cost if an identical solution is simulated more than once. Mei *et al.* [9] modelled the deadheading cost as a Gamma distribution, taking into account the probability of the presence of path between the vertices i and j, $q_{i,j}$. The deadheading cost of the path between the vertices i and j is defined as $dc_{i,j} \sim G(k^c_{i,j}, \theta^c_{i,j})$ if $rand < q_{i,j}$; otherwise $dc_{i,j} = \infty$. In [9], the shape parameter $k^c_{i,j}$ and the probability $q_{i,j}$ are set to 20 and 0.95, respectively. The scale parameter $\theta^c_{i,j}$ is set to $dc_{i,j} / k^d_{i,j}$ [9].

D. Benchmarks of CARP with Uncertainties

The most used technique to generate benchmark functions for CARP with uncertainties is by adding random perturbations to one or more variables of existing DCARP instances to model uncertainties or replace the deterministic parameters by user-defined stochastic distribution presented in Section III-C.

To the best of our knowledge, the benchmark sets of CARPSD and UCARP are usually extended from 3 well-known benchmark sets of static CARP, *gdb* [23], *egl* [40] and *val* [24]. The corresponding UCARP benchmark sets designed by [9] are referred to as *ugdb*, *uegl* and *uval*, respectively. In particular, *ugdb* and *uval* have been popular and are most used in recent years. Tables II summarizes the list of articles that have used *ugdb* and *uval* in their case studies and serves as a list of baselines.

TABLE II References that used *ugdb* and *uval*, two UCARP benchmark sets extended from the well-known static CARP benchmark sets, *gdb* [23] and *val* [24] respectively, designed by [9]. In *ugdb* and *uval*, the demands of tasks and costs of tasks follow Gamma distributions and the presences of task and edges are modelled as boolean variables, as detailed in Section III-C.

DISTRIBUTION	UNCERTAINTIES	
GAMMA	DEMAND OF TASK	COST OF EDGE
BOOLEAN	PRESENCE OF TASK	PRESENCE OF EDGE
REFERENCES	[9], [10], [22], [17], [25], [26], [27] [28], [29], [30], [31], [32], [33]	

In some of the work, such as [6], [37], artificial instances have been used as test cases. Chen *et al.* [20], [21] designed instances based on real data of road network in Shanghai city. However, the artificial instances are often small- or medium-scale, and smaller than the biggest instance in *ugdb* and *uval*. To illustrate the popularity of stochastic variables and their distributions assumed in the self-designed instances, the list of articles that have used self-designed instances are provided in Table III, grouped by their assumptions.

E. Summary of Uncertainty Modelling

The study on CARP with uncertainties has a short history, despite its importance in real life. Different variants have been proposed based on the actual non-deterministic variables in the corresponding real-world applications. Modelling the uncertainties is important both for the problem formulation and the generation of scenarios for testing solutions or solvers, as the uncertain variables affect the feasibility of solutions and the actual total cost. The assumption on models relies on the knowledge of the actual problem and accessible data, e.g., the amount of snow on each street during the past month in the snow removal problem and amount of waste in the urban waste collection problem. When the historical data of uncertain variables are available, the prediction of variables is feasible in some of the cases or at least the bounds for the variables can be determined. Additional forecast information will be beneficial for adjusting the model, a distribution controlled by one or more parameters, which are called stochastic control parameters in this paper. In most work, the demand of tasks is assumed to follow a Gamma or normal distribution (cf. Tables II and III) controlled by given static parameters. However, some of the uncertainties do not follow a Gamma or normal distribution, or even cannot be modelled as known probability distributions. Additionally, rarely occurring events are often ignored while designing the model. Building more realistic variable distributions and scenario-based optimization methods without assuming variable models are two valuable directions for research.

TABLE III References that used self-designed benchmarks, grouped by their assumptions.

DISTRIBUTION	UNCERTAINTIES	
	DEMAND OF TASK	TIME
NORMAL	[6], [11], [13], [12], [34], [35]	[20], [21]
LOG-NORMAL	[36]	
POISSON	[7], [8]	
UNIFORM	[37], [38], [39]	
FUZZY	[18], [19]	

IV. Reliability and Robustness of Solutions

In the deterministic CARP, the commonly used performance measure of a solution $\boldsymbol{x}$ is the total cost $C(\boldsymbol{x})$, defined in Eq. (1). Solvers for DCARP aim at minimizing $C(\boldsymbol{x})$ while satisfying the constraints, as formulated in Section III-A1). In the non-deterministic versions, due to the random variables, the exact value of the total cost of a given solution $\boldsymbol{x}$ is computable only after serving all the available tasks. The lower bound of solution costs are unknown. In the deterministic CARP, the number of trips, $T(\boldsymbol{x})$, is deterministic, while in the uncertain version, it may be higher than the number of vehicles due to route failures. Thus, one or more vehicles may route extra trip(s) due to route failures as described previously in Section III-C3), which implies higher total cost. Therefore, special performance measures should be designed for evaluating the reliability and robustness of solutions optimized for the CARP with uncertainties, and special repairing operators should be designed for handling the violation of constraints. The performance of solvers or repairing operators in robust optimization of CARP with uncertainties can be evaluated by the quality of their recommended solutions. When evaluating a solution, a number of evaluations is often needed in an uncertain environment in order to obtain a good estimation of the solution's quality. The expected performance or average performance over the tested scenarios is widely used to measure the quality of a solution, while the worst-case performance and the variance are often used to measure its robustness.

The performance measures proposed so far in the literature are summarized as follows, some of which have been directly utilized as optimization objectives (Tables IV and V).

A. Performance Measures Using Known Variable Distributions

As presented in Section III-D, the instances of CARP with uncertainties are generated based on a benchmark of DCARP instances, by replacing the deterministic edge-cost with a stochastic model. If the expected values are calculable (e.g., random demand respecting a normal distribution), the expected values and standard deviations of performance indicators can be used to measure the reliability of a solution, such as in [6], [8], [36], [37].

1) Expected Cost & Deterministic Cost

The expectation of variables was set as the deterministic variables of the DCARP used for generating CARPSD in [6], [12]. Given a solution $\boldsymbol{x}$, the deterministic cost of the DCARP, denoted as $C_d(\boldsymbol{x})$, is used to model the expected cost of the corresponding CARPSD or uCARP, denoted as $\mathbb{E}[C(\boldsymbol{x})]$ [6], [9].

2) Expected Number of Trips

The number of trips influences the cost of returning to the depot. Minimization of the expected number of trips [11], $\mathbb{E}[T(\boldsymbol{x})]$, implicitly leads to the minimization of cost.

3) Expected Makespan

The cost of the longest trip in a solution $\boldsymbol{x}$ is called *makespan* [34]; its expectation is denoted as $\mathbb{E}[M(\boldsymbol{x})]$. Fleury *et al.* [34] applied bi-objective optimization considering both the total cost of the trips and the makespan.

4) Variability

The measure *variability* [6], [12] is defined as $\sigma[C(\boldsymbol{x})]/\mathbb{E}[C(\boldsymbol{x})]$ and calculated for evaluating the robustness of any given solution $\boldsymbol{x}$. In [6], [12], it was observed that taking into account the standard deviation of costs in the objective function led to lower variability, thus producing more robust solutions.

Using expectation as objectives or measures is easy to execute and methods for handling DCARPs can be applied directly. However, it is less realistic in real-world applications as the probability distributions of random variables are usually unknown.

B. Performance Measures Over Multiple Simulations

When the exact distributions are unknown or the expectation and variance cannot be computed directly, multiple simulations over different scenarios often have to be carried out in order to estimate expectations and variances.

1) Average Performance

In practice, expectations of variables are sometimes unknown and it is not possible to sample all the scenarios of a CARPSD or uCARP. Statistics collected during $n \in \mathbb{N}^+$ independent simulations of $\boldsymbol{x}$ can be used to estimate some performance measures (e.g., [6], [12]).

a) Estimation of Cost

The empirical average of the total cost $\mathbb{E}_n[C(\boldsymbol{x})]$, as well as the empirical standard deviation $\sigma_n[C(\boldsymbol{x})]$, have been widely used for estimating the expected cost $\mathbb{E}[C(\boldsymbol{x})]$ and standard deviation $\sigma[C(\boldsymbol{x})]$, respectively. Examples include [6], [10], [12], [17].

b) Estimation of Number of Trips

Similar to the above, the empirical average of the number of trips $\mathbb{E}_n[T(\boldsymbol{x})]$, as well as the empirical standard deviation $\sigma_n[T(\boldsymbol{x})]$, have been used for estimating $\mathbb{E}[T(\boldsymbol{x})]$ and $\sigma[T(\boldsymbol{x})]$, respectively. Examples also include [6], [10], [12], [17]

c) Robustness Estimator

The variability can also be estimated by $\sigma_n[C(\boldsymbol{x})]/\mathbb{E}_n[C(\boldsymbol{x})]$ [11].

2) Best-Case Performance

The best-case performance, defined as the cost of the best solution and the number of trips in the best solution found, is used as a performance measure by Fleury *et al.* [6]. Such evaluation is optimistic but it gives a brief idea of how low the cost could possibly be.

3) Worst-Case Performance

A pessimistic measure is the worst-case performance, i.e., simulating a solution on random realizations of a UCARP instance and its worst performance (highest total cost or highest number of trips) used as the performance metric. Then, the solution with the best worst-case performance is recommended. This is the classic Wald's maximin model [41] for decision making. Such a performance measure is also called a robustness measure.

C. Other Robustness Measures

The performance measures presented above are used most often in the literature. This section summarizes some other metrics for measuring robustness of solutions other than the worst-case performance measures, variability [6], [12] and the robustness estimator [11]. These measures

TABLE IV Performance measures of solutions for the CARP with uncertainties. The notations used in this table and detailed description of each measure are explained in Section IV. The ρ in measures (2), (8), (11), (16) and (19) is a user-specified constant for a linear combination of expectation/average and standard deviation.

		SCENARIO-BASED		
TRAGET	WITH DISTRIBUTION ASSUMPTIONS	AVERAGE	BEST-CASE	WORST-CASE
COST	(1) EXPECTED COST: $\mathbb{E}[C(\boldsymbol{x})]$ (2) $\mathbb{E}[C(\boldsymbol{x})] + \rho\sigma[C(\boldsymbol{x})]$ OR $\sigma[C(\boldsymbol{x})]$ (3) VARIABILITY: $\frac{\sigma[C(\boldsymbol{x})]}{\mathbb{E}[C(\boldsymbol{x})]}$ (4) $\mathbb{E}[C(\boldsymbol{x})]$ WITH UPPER BOUNDED PROBABILITY OF AN EXTRA RETURN (5) $\mathbb{E}[C(\boldsymbol{x})]$ WITH UPPER BOUNDED PROBABILITY OF #TRIPS (6) $\mathbb{E}[C(\boldsymbol{x})]$ WITH UPPER BOUNDED VARIANCE OF COST (7) EXPECTED MAKESPAN: $\mathbb{E}[M(\boldsymbol{x})]$ (8) $\mathbb{E}[M(\boldsymbol{x})] + \rho\sigma[M(\boldsymbol{x})]$ OR $\sigma[M(\boldsymbol{x})]$ (9) THRESHOLD-BASED ROBUSTNESS INDICATOR	(10) AVERAGE COST $\mathbb{E}_n[C(\boldsymbol{x})]$ WITHOUT ADDITIONAL CONDITION (11) $\mathbb{E}_n[C(\boldsymbol{x})] + \rho\sigma_n[C(\boldsymbol{x})]$ OR $\sigma_n[C(\boldsymbol{x})]$ (12) ROBUSTNESS ESTIMATOR: $\frac{\sigma_n[C(\boldsymbol{x})]}{\mathbb{E}_n[C(\boldsymbol{x})]}$	(13) $\min_{i\in\{1,\dots,n\}} C^{(i)}(\boldsymbol{x})$	(14) $\max_{i\in\{1,\dots,n\}} C^{(i)}(\boldsymbol{x})$
TRIP	(15) EXPECTED #TRIP: $\mathbb{E}[T(\boldsymbol{x})]$ (16) $\mathbb{E}[T(\boldsymbol{x})] + \rho\sigma[T(\boldsymbol{x})]$ OR $\sigma[T(\boldsymbol{x})]$ (17) PROBABILITY OF TRIP INTERRUPTION	(18) AVERAGE #TRIPS: $\mathbb{E}_n[T(\boldsymbol{x})]$ (19) $\mathbb{E}_n[T(\boldsymbol{x})] + \rho\sigma_n[T(\boldsymbol{x})]$ OR $\sigma_n[T(\boldsymbol{x})]$ (20) PERCENTAGE OF INTERRUPTED TRIPS	(21) $\min_{i\in\{1,\dots,n\}} T^{(i)}(\boldsymbol{x})$	(22) $\max_{i\in\{1,\dots,n\}} T^{(i)}(\boldsymbol{x})$
CONSTRAINTS		(23) RELIABILITY-BASED ROBUSTNESS MEASURE (24) REPAIR-BASED ROBUSTNESS MEASURE		
OTHER MEASURES		(25) RELATIVE PERFORMANCE MEASURES (APPLICABLE TO MOST OF THE ABOVE MEASURES) (26) MULTIOBJECTIVE MEASURES (APPLICABLE TO A SET OF DIFFERENT MEASURES)		

TABLE V Summary of studied problem models and performance measures of solutions for the work on CARP with uncertainties reviewed in this paper. The first column indicates the studied problem model, while the second column lists relevant references. The numbers inside "()" on the second row are the indices of performance measures listed in Table IV. The last column indicates if the time measure has been reported, where "RT" means that the run time is reported and "TT" refers to the training time for routing policies. In the other cells (besides the column and row headers in bold), "O" stands for the case that the corresponding measure (shown as row header) is used as a (term of) objective function in the corresponding paper (shown as column header); "✓" means that it has been used as a measure when discussing the experimental results, but not as a (term of) objective function; "D" means that it is discussed but not used.

		PERFORMANCE MEASURES																										
VARIANT	**REF**	**(1)**	**(2)**	**(3)**	**(4)**	**(5)**	**(6)**	**(7)**	**(8)**	**(9)**	**(10)**	**(11)**	**(12)**	**(13)**	**(14)**	**(15)**	**(16)**	**(17)**	**(18)**	**(19)**	**(20)**	**(21)**	**(22)**	**(23)**	**(24)**	**(25)**	**(26)**	**TIME**
CARPSD	[6]	O		✓							✓	✓				✓			✓	✓	✓	✓	✓			✓		
	[11]	O	O								✓	✓	✓						✓	✓	✓					✓		RT
	[12]	O	O	✓	D	D	D				✓	✓	✓				D		✓	✓								RT
	[13]	O									✓	✓							✓	✓						✓		
	[34]	O	O					O	O																		✓	
	[35]	O	O					O	O																		✓	RT
	[7]													✓														RT
	[8]	O									O										O							RT
	[36]	✓									O							✓								✓		
	[37]	O																										
	[38]	O																								✓		RT
	[39]	O							O																		✓	RT
uCARP	[9]	O								D	✓				D									D	✓			
	[22]	O								D	✓				D									D	✓			
	[10]										O	✓		✓	✓													RT
	[17]										✓	✓			O													RT
	[25]										O				✓													
	[26]										O																	
	[27]										O																	
	[28]										O																	TT
	[29]										O																	
	[30]										O				✓													
	[31]										O	✓																
	[32]										O	✓																
	[33]										O	✓																
CARPFD	[18]	O									✓					O				✓							✓	
	[19]	O																										
CARPST	[20]										O																	RT
	[21]										O																	

are designed to quantify the feasibility of solutions under uncertainties (e.g., the overflow of capacity on a route) or serve as additional objectives [9].

1) Interrupted trips

A trip may be interrupted due to an augmented demand or the absence of an edge. This induces an increase in the number of trips. Fleury *et al.* [11] considered directly the percentage of simulations with interrupted trips among n simulations of a given solution $\boldsymbol{x}$ to estimate the probability of trip interruption, which is also the probability of introducing extra trip(s) into the solution $\boldsymbol{x}$.

2) Threshold-Based Robustness Measure

In real-world applications, optimizing the worst-case performance can be conservative and the expected performance is sometimes not computable as the true model of uncertainties is unknown or difficult to compute. The probability of reaching a given quality threshold, specified by decision makers with their affordability, was introduced as a measure in [9], but not actually used in their study.

3) Reliability-Based Robustness Measure

The feasibility of a solution cannot always be guaranteed due to the possible uncertainties in the constraints. The reliability of a solution can be measured by the probabilities of satisfying constraints. Mei *et al.* [9] discussed a reliability-based robustness measure which requires a lower bound P_0 (i.e., confidence probability), for the probability of satisfying a given constraint g: $\mathbb{P}[g(\boldsymbol{x}) \leq 0] \geq P_0$.

4) Repair-Based Robustness Measure

The repair-based robustness measure defines repair operators to change an infeasible solution to a feasible one [9]. The repair-based robustness measure is more favourable than the reliability-based one because, in UCARP, the constraints are hard constraints and an infeasible solution should be repaired to be feasible [9].

D. Measures Under Conditions on Confidence Probability

In addition to the minimization of average cost and the number of trips, more sophisticated measures have been used as optimization objectives [12].

1) With an Upper Bounded Probability of an Extra Return to the Depot

Minimization of $\mathbb{E}[C(\boldsymbol{x})]$ under the condition $p_i \leq \epsilon$ for any i with a fixed $\epsilon > 0$ [12], where p_i is the probability of returning to the depot during the ith route.

2) Upper Bounded Probability of Number of Trips

Minimization of the empirical average cost of solution $\boldsymbol{x}$ over n simulations, under the condition $\mathbb{P}[T > t] \leq \epsilon$ with $\epsilon > 0$, where T is the actual number of trips and t is a given upper bound [12].

3) With Upper Bounded Variance of Cost

Minimization of the empirical average cost of solution $\boldsymbol{x}$ over n simulations, under the condition $\sigma[C(\boldsymbol{x})] \leq \epsilon$ with a fixed $\epsilon > 0$ [12].

E. Relative Performance Measures

In the deterministic case, if the lower bound of cost (LB_C) is known, it is straightforward to evaluate a solution by computing the distance of its cost to LB_C and to evaluate an algorithm by measuring how fast it converges to LB_C. However, in the uncertain case, there is no fixed optimal solution due to the random aspect. Therefore, the relative performance measure is also used for evaluating a given algorithm A, realized by comparing the solutions computed by A to some baseline solutions [6], [11], [13], [36], [38]. For instance, Fleury *et al.* [6] calculated the percentage of simulations with a higher number of trips than a baseline solution computed by a hybrid genetic algorithm. The studies of [11], [13] compared the empirical average cost of solutions computed by their proposed stochastic memetic algorithm (detailed later in Section V-B2a) against the ones computed by a deterministic memetic algorithm on a set of CARPSD instances.

F. Multiobjective Measures

All the performance measures presented in Sections IV-A–IV-D can be set as objectives for solvers. Sometimes, more than one metric should be optimized depending on the actual needs of the real-world applications. Multiple objectives can be optimized simultaneously using some multiobjective optimization (MOO) approaches or be considered as a single objective using a linear combination of them. In [11], [12], both the average cost and its variance are combined by a linear combination of these two objectives, $\mathbb{E}[C(\boldsymbol{x})] + \rho\sigma[C(\boldsymbol{x})]$ with a constant ρ. Bi-objective optimization approaches have been applied to CARP with uncertainties in [18], [34], [35], [39].

G. Computation Time

The run time performance is important for applications in reality. Most of the published work did not report the computation time; only a few of them did.

The run time for finding the optimal solution of DCARP instances, transformed from CARPSD assuming known variable expectations, was reported in [8], [11], [12], [35]. In [37]–[39], the run time performance of solving the robust models of CARPSD was evaluated using different solvers. When the variable distributions of UCARP are unknown, the run time is defined as the execution duration till certain stopping criteria are met, such as when a given number of generations elapses or when a robust solution dominates for a given number of iterations [10], [17]. Chen *et al.* [20] studied the average run time of solving some CARP-SSTT instances optimally, while the run time of finding ϵ-optimal solutions is studied in [7]. Instead of the run time of (approximately) solving UCARP instances, Maclachlan *et al.* [28] reported the computational time for training routing policies.

Due to different UCARP variants, the computation time is often very difficult to compare directly among different studies.

H. Discussion

The performance measures of solutions for CARP with uncertainties can be categorized into different groups, considering the objectives, constraints, scenarios and the dependence on variable distribution assumptions, as summarized in Table IV. The measures used in each publication reviewed in this survey are given in Table V.

Measures for evaluating solutions are usually defined or selected by human decision makers. A solution to UCARP with low risk of extremely high cost can be too conservative and leads to low reward or high cost. Sometimes, solutions with high expected or average performance are preferred, while solutions with low risk are favourable in the cases that failure leads to high cost. Balancing the trade-off between risk and cost is not trivial. Given a set of solutions (decisions), the human decision makers need to determine the *optimal* solutions using pre-defined decision-making criteria based on the practical consideration.

In Table IV, measures (1)-(9) and (15)-(17) are not very practical as they are calculated based on known variable distributions, which are usually not accessible in real-world applications. Instead, measures (10)-(12) and (18)-(20) over a set of given scenarios can be calculated and then used. (14) and (22) indicate how bad a situation the evaluated solution could lead to, while (13) and (21) indicate how good the situation could be. They could be regarded as *pseudo* lower or upper bounds. It is possible in a solution that a large number of vehicles or trips are involved with a low travel cost, which will lead to a high cost of vehicle usage. In such cases, the measures related to the number of trips, (15)-(22), are crucial. Measures (23) and (24) help to handle stochastic constraints. As the actual lower bound of cost is unknown in the CARP with uncertainties, the relative performance measure (25) assists the understanding of solution quality. The multi-objective measure (26) generally refers to any measure which considers more than one measure presented above.

V. Solution Approaches

After reviewing uncertainty models and performance measures, this section categorizes different solution approaches to robust optimization for the CARP with uncertainties. CARPs can be transformed into VRPs. However, the number of vertices of the resulting VRP triples the number of tasks of the original CARP [42], which implies an increase in the computation time. Therefore, the methods for solving (stochastic) VRPs [43]–[45] are often not suitable to be applied to solving (stochastic) CARPs. Section V-A presents the uncertainty handling techniques designed specially for avoiding the trip interruptions (Section V-A1) and repairing failed solutions (Section V-A2). Published work around robust optimization in CARP with uncertainties will be briefly divided into two categories: evaluating the robustness of solutions optimized for the deterministic CARP transformed from the UCARP, and approaches for directly searching for robust solutions for UCARP. As approaches of these two categories usually overlap, we propose a new taxonomy as shown in Figure 3 (Sections V-B and V-C) to facilitate our understanding of different approaches. Finally, recent advances in learning routing policies are presented in Section V-D.

A. Handling Uncertainties

Uncertainties in CARP usually lead to re-planning of routes. There are four major sources of uncertainties: a cancelled task, an absent edge, a task on an absent edge and a violation of vehicle capacity. The first one is easy to understand. The second and third ones may be due to a temporal maintenance or an accident on a road (with a task), which is difficult to forecast at the time of planning. The last one is due to the perturbation of demands. A vehicle may exhaust its capacity before completing a trip, then a *trip interruption*, sometimes called *route failure*, occurs. A prior and posterior techniques (Figure 1) have been proposed to reduce the probability of route failure or increase the ability to react optimally in such situations.

1) A Prior Techniques

To reduce the probability of route failure due to a violation of vehicle capacity, two a prior techniques by editing the problem constraint [6] and objective function [12] have been proposed, respectively.

a) Slack Approach—Conservative Constraint

To avoid solutions in which the total load is close to the vehicle capacity Q, Fleury *et al.* [6] reduced the vehicle capacity by $k\%$ before the deterministic optimization process. This technique is named as a *slack*

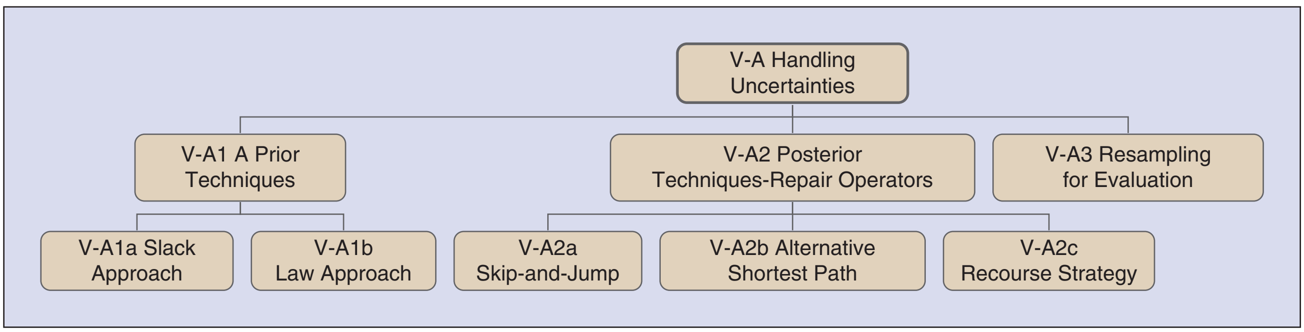

FIGURE 1 Techniques for handling uncertainties.

approach in [6], while the regular constraint on capacity without reduction, as in classic DCARP, is called the *tight approach*. In [6], a pseudo capacity 90% Q is used during optimization, while the actual capacity Q is used when evaluating the solutions. The slack approach is capable of avoiding some violation of capacity constraint due to stochastic demands, however it may increase the number of trips, which implies an increase in the solution cost. Moreover, the control parameter k is difficult to set. A similar approach was used by [36] to generate DCARP instances.

b) Law Approach—Bi-Objective Function

Different from revising the vehicle capacity as in the slack approach, the *law approach* [12] varies the objective functions to handle uncertainties. Instead of minimizing the cost of solutions, the *law approach* considers the minimization of a weighted sum of the cost and its standard deviation.

2) Posterior Techniques—Repairing Operators

Several operators have been designed for repairing infeasible solutions during execution [9].

a) Skip-and-Jump

In case of a cancelled task or when its actual demand is 0 at the time of serving, as well as when a task is on an absent edge, the vehicle skips the task and jumps to the next one.

b) Alternative Shortest Path

If an edge without task is absent, namely no longer being present on the map, this absent edge is replaced by the shortest path between its two vertices.

c) Recourse Strategy—Return and Continue

In case of a violation of vehicle capacity before serving a task t, the vehicle must return to the depot via the shortest path, refill goods or unload goods, go back to the task t and serve it, then continue its planned trip (e.g., [7], [11]). At least one supplementary trip will be induced depending on the number of times of capacity violation of a planned trip occurred, which implies an increase in the total cost. Note that this repairing technique is applied under the assumptions that (i) a task's actual demand is known before it is served and (ii) assigning another vehicle is impossible or inefficient. Techniques like this are called *recourse strategies* [11]. For instance, in the waste collection problem, a vehicle may be fully filled in the middle of a road (edge), thus cost has already been caused. For a better understanding of recourse strategies, we translate the recourse strategy used by [9] into Algorithm 1, which also facilitates its reproduction.

3) Resampling for Evaluation

Techniques for handling uncertainties are important in optimization under uncertainties, among which sampling is the most discussed for reducing the probability of mis-ranking two solutions due to uncertain test scenarios [46], [47]. In [10], the solutions for uCARP were evaluated on an identical set of scenarios for a fair comparison. Wang *et al.* [17] resampled uCARP instances for evaluating each solution and used its maximal cost obtained on the sampled instances (thus worst-case cost) for ordering the solutions. Chen *et al.* [20], [21] used Monte Carlo simulations to evaluate the robustness of solutions.

These techniques for handling uncertainties are often integrated into the robust optimization approaches for approximately solving CARPs with uncertainties. Sections V-B and V-C will present the approaches when assuming known or unknown distributions of variables, respectively.

Algorithm 1 The recourse strategy used in [9], translated into pseudo-code by us.

```
Require: Planed route of tasks r = (v0, t1, t2, ..., tl)
Require: Current location v
Require: Current available capacity Q'
Locate the next task t
if Next task t exists then
    if Q' ≥ E[d(t)] then
        Traverse to head(t) through the original path
    else
        Return to the depot and empty, Q' = Q
        Traverse to head(t) through the shortest path
    end if
    while tail(t) not reached do        ▷ t is not completed
        Loop 1
            Serve t progressively and update Q'
            if Arrive at tail(t) then      ▷ t is completed
                break while
            end if
            if Q' = 0 then      ▷ Capacity is exhausted on the way of service
                Return to the depot and empty, Q' = Q
                                ▷ Neglect the remaining demand
                Traverse to head(t) through the shortest path
                from current location
                break Loop 1
            end if
        End Loop
    end while
else
    Return to the depot
end if
```

B. Robust Optimization With Variable Distribution Assumptions

This section presents the work on robust optimization in CARP with uncertainties when known variable distributions are assumed.

1) Deterministic Optimization, Stochastic Evaluation—Solving transformed DCARP

A number of existing work did not design algorithms for solving CARP with uncertainties directly, but addressed UCARP as transformed DCARP and solved it using DCARP solvers, then evaluated the solutions on realizations (samples) of the corresponding UCARP. They focused more on the evaluation of solutions for selecting optimal heuristics for the UCARP. A two-phase framework, *deterministic optimization* phase and *stochastic evaluation* phase (DOSE), has been widely used in the robust optimization of UCARP. We illustrate this two-phase framework, DOSE, in Figure 2. During the optimization phase, algorithms for solving DCARP are applied directly to the static version of the corresponding UCARP, where they solve the UCARP instances by utilizing the expectation of the random variables. In some work, the evaluation phase is also called the replication phase because each solution is performed on a number of replications of UCARP. Each replication of a UCARP is a DCARP instance as the random variables are replaced by their deterministic realizations, usually via Monte Carlo simulations. This technique is also called "resampling" in noisy optimization aiming at reducing the probability of mis-ranking solutions [48], [49]. This framework assumes that the model of the random variables, or at least, the expectation of the random variables, perfectly reflects the true one in real life. In other words, it assumes that the expectations of the random variables are known.

- **Deterministic optimization process:** During this stage, the given UCARP with *stochastic models for demands* of known expectations is transformed to a DCARP instance of which the demands are deterministic and equal to the expectations of the stochastic demands. Then, the resulting DCARP instance can be solved by existing *heuristics for the DCARP*.
- **Stochastic evaluation process:** Then, the heuristics are evaluated and selected based on some predefined *performance metrics*, computed using the simulation results of their optimized, deterministic solutions on a set of sampled instances of UCARP. Some *repairing techniques* may be applied during simulation if the actual demand of a task to serve next exceeds the vehicle's available capacity. Section IV gives a comprehensive review of different measures that have been used in the literature.

Core components involved in the above framework are: (i) the stochastic models for demands; (ii) the heuristics for the DCARP, (iii) the performance metrics for evaluating heuristics and (iv) the repairing techniques. Complete lists of previously studied (i), (iii) and (iv) have been introduced in Section III-C, Sections IV and V-A2), respectively. Although all the heuristics for the DCARP can be used in this two-stage framework, we will not cover all of them but only the ones used by the publications with a focus on UCARP. The corresponding demand model and performance metric(s) will also be discussed.

The algorithms designed for DCARP that have been used in this framework are summarized as follows.

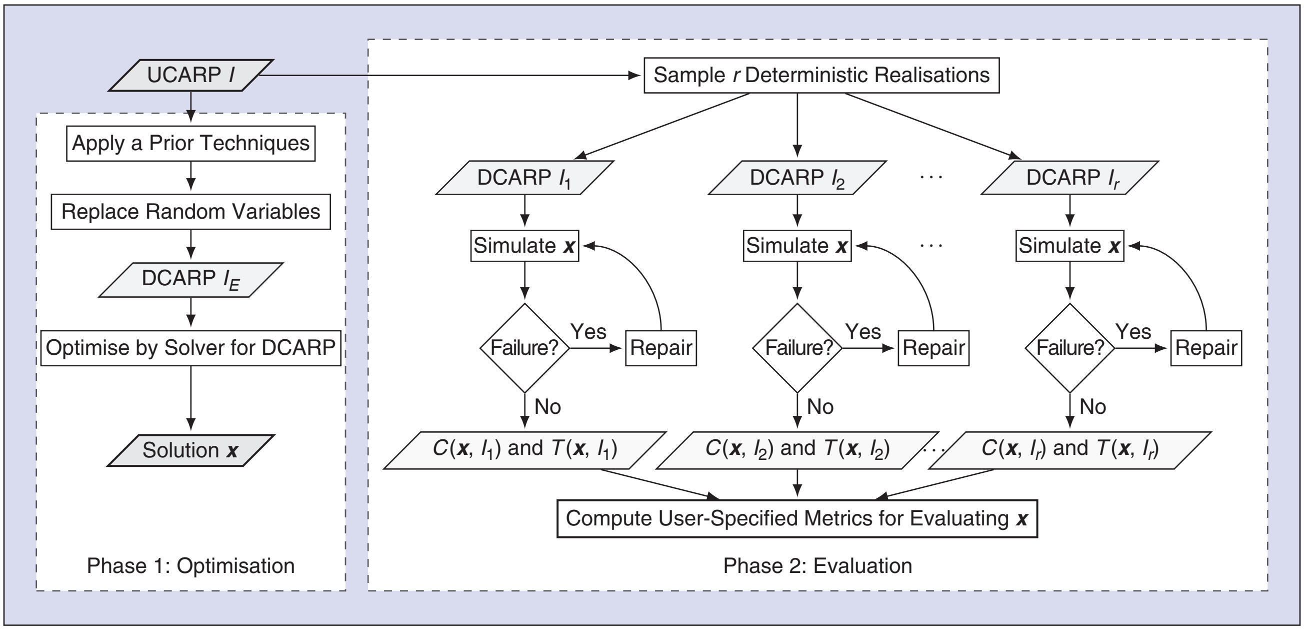

FIGURE 2 Two-phase framework: deterministic optimization–stochastic evaluation (DOSE). $DCARP_E$ denotes the DCARP instance generated by replacing the random variables of the given UCARP by their expectations. $I_1, \ldots, I_r$ denote the r deterministic realizations (samples) of UCARP. $C(\boldsymbol{x}, I_i)$ and $T(\boldsymbol{x}, I_i)$ denote the resulted cost and number of trips of simulating a solution $\boldsymbol{x}$ on a deterministic realization I_i.

a) Meta-Heuristics

Meta-heuristics have been widely used for handling DCARP [2], some of which have been used in the DOSE framework. CARPSD with truncated normal random demands was considered in [11]–[13] and this two-stage framework was used. In [12], [13], a hybrid genetic algorithm (HGA) is used as the DCARP solver. Besides the average performance measures presented in Section IV as optimization objectives, the study of [12] also considered the average cost and its variance simultaneously by a linear combination of these two objectives, $\mathbb{E}_n[C(\boldsymbol{x})] + \rho\sigma_n[C(\boldsymbol{x})]$, where ρ is a constant. The empirical average of solution $\boldsymbol{x}$ over n simulations, $\mathbb{E}_n[C(\boldsymbol{x})]$, is used to estimate the expectation of the cost of $\boldsymbol{x}$. In [11], a deterministic memetic algorithm (MA) [50] is used as the DCARP solver. This heuristic is referred to as deterministic MA (DMA). In its comparison and selection steps, the solutions are evaluated on the associate DCARP only. We translate the procedure into a more general *deterministic generate-and-test process* as in Algorithm 2. In the "generate" step, an algorithm for DCARP generates one or more solutions, and then the generated solution is evaluated and tested if it is better than the current recommendation. Fleury *et al.* [11] evaluated a DMA (denoted as MA1) and another DMA with slack approach (denoted as MA2) on CARPSD with the minimization of deterministic cost as objective, and concluded that the use of slack approach increased the deterministic cost of the associate DCARP but provided robust solutions to CARPSD. MA was further extended to stochastic MA (SMA) in [11] for solving CARPSD instead of DCARP, detailed later in Section V-B2a. During the stochastic evaluation, Fleury *et al.* [11] assumed that (i) a trip can be interrupted for at most once and (ii) if it occurs, with a high probability the point of failure is just before the last task. [12] minimized both the average cost and variance by a linear combination of these two objectives.

Mei *et al.* [9] also used the DOSE framework. During the deterministic optimization phase, an MA with extended neighbourhood search (MAENS) [51] and RTS*, which is a repair-based tabu search (RTS) with an adjusted stopping criterion [52], were used to solve the *gdb* instances, based on which their uCARP instances are generated. A core feature of [9] is that, instead of recording only the best-so-far solution, all the best feasible solutions updated during the search process (line 18 of Algorithm 3) are recorded. During the stochastic evaluation phase, each solution is simulated on 30 realizations of the corresponding uCARP and the lowest empirical average cost of the recorded solutions is used as the robustness indicator [9]. Mei *et al.* [9] observed that (i) for a DCARP instance, there could be more than one globally optimal solution of different robustness levels; (ii) the optimal solution for uCARP, in terms of the empirical average cost, may not be the optimal solution for DCARP, and concluded that solving uCARP instances by applying DCARP solvers to its associate DCARP instance will make finding highly robust solutions difficult.

The arc routing problems with stochastic demands (ARPSDs) are considered by [36] and the procedure shown in Algorithm 4 is used. Randomized savings heuristic was applied to the arc routing problem (RandSHARP) for solving the DCARP instances sampled using the slack approach (π_s), and the reliability and robustness of the obtained solution were estimated via Monte Carlo simulations (π_e) to decide whether to continue or stop searching [36].

b) Bi-Objectivity

The above work only considered single objectives, either minimizing the total cost or minimizing the total number of trips. Lacomme *et al.* [53] incorporated a local search to a NSGA-II [54] for solving DCARP with the minimization of the cost and the duration of the longest trip of a solution as its two objectives. Then, a bi-objective NSGA-II with local search was used as solver during the deterministic optimization phase for solving CARPSD [34], [35]. In [18], the total cost and the total number of vehicles were minimized simultaneously.

c) Other Algorithms

In [7], DCARP was formulated as a set partitioning problem and solved using a branch-and-price algorithm.

Algorithm 2 **Deterministic generate-and-test process, generalized from [11]. At its comparison and selection steps, the solutions are evaluated on the associate DCARP only.**

Require: I: A UCARP with stochastic models of variables of known expectations
Require: S_D: A DCARP solver, e.g., a memetic algorithm
$I_E \leftarrow$ Transform I to its associate DCARP by replacing the stochastic variables to their expectations
$\boldsymbol{x} \leftarrow S_D$ generates a solution for I_E
$y \leftarrow C(\boldsymbol{x}, I_E)$ ▹ *Evaluate* $\boldsymbol{x}$ *once on* I_E*, other measures* ▹ *in Table IV can be used*
while Stopping criteria not met **do**
  $\boldsymbol{x}' \leftarrow S_D$ generates a solution for I_E from $\boldsymbol{x}$ by perturbation
  $y' \leftarrow C(\boldsymbol{x}', I_E)$ ▹ *Evaluate* $\boldsymbol{x}'$ *once on* I_E
  if $y' < y$ **then** ▹ *Test and replace*
    $\boldsymbol{x} \leftarrow \boldsymbol{x}'$
  end if
end while
return $\boldsymbol{x}$

2) Stochastic Optimization, Stochastic Evaluation—Solving Sampled DCARP

a) Adaptation of Meta-Heuristics

MA is adapted to CARPSD by using the empirical average objective value computed over a number of simulations (as in the *Evaluation phase* of Figure 2) for comparing and selecting solutions, instead of using the deterministic value (Algorithm 2) [11], [12]. The resulting algorithm is called stochastic MA (SMA) due to the stochastic evaluation of solutions (Algorithm 3), which is exactly the same as the stochastic evaluation phase of the optimization-evaluation framework; thus, repairing operations may occur [11]. SMA with different objectives have been considered; one aims at minimizing the expected total cost (denoted as MA3) while the other aims at minimizing the expected number of trips (denoted as MA4) [11]. Fleury *et al.* [11] compared two SMAs, a DMA with slack approach and a DMA without slack approach on a set of CARPSDs to the LB_Cs of corresponding DCARPs. The SMAs were more robust than the DMAs on the considered CARPSDs.

In [6], [12], only the average performance was used for evaluating solutions, while Mei *et al.* [9] concluded that such statistics were not enough to indicate highly robust solutions.

b) Stochastic Path Scanning with Adaptive Large Neighbourhood Search

The study of [8] considered CARPSD with Poisson distributed demands. In [8], a mathematical analysis of the probability of route failure and the expected cost was given under the assumptions that a trip could be interrupted for at most once and the demand was uniformly distributed on the corresponding edge. A stochastic path scanning method was proposed to construct solutions considering the above analysis as additional constraints and an adaptive large neighbourhood search (ALNS) heuristic with a removal-insertion operation was designed to iteratively improve the solutions [8]. At each iteration of ALNS, a removal heuristic destroys the current solution by removing served edges and then an insertion heuristic reinserts the removed edges differently [8]. It is notable that the used removal (or insertion) heuristic is selected from a set of removal (or insertion) heuristics using a roulette-wheel selection with weights updated adaptively according to the quality of newly constructed solutions by the removal-insertion operation [8]. The stochastic path scanning method with ALNS is also applied to CARP with stochastic service and travel times (CARP-SSTT) [20].

Algorithm 3 **Stochastic generate-and-test process assuming known expectations of random variables, summarized by us.**

Require: I: A UCARP with stochastic models of variables of known expectations
Require: r: Resampling number
$I_E \leftarrow$ Transform I to its associate DCARP by replacing the stochastic variables to their expectations
$\boldsymbol{x} \leftarrow$ Generate a solution for I_E
$y \leftarrow 0$
for $i \in \{1,\ldots,r\}$ **do**
  $I_i \leftarrow$ Sample a deterministic realization of I
  $y \leftarrow y + C(\boldsymbol{x}, I_i)$ ▹ *Evaluate* $\boldsymbol{x}$ *once on* I_i *and* ▹ *cumulate the cost*
end for
$y \leftarrow y/r$ ▹ *Average cost of* $\boldsymbol{x}$ *over r simulations,* ▹ *other measures in Table IV can be used*
while Stopping criteria not met **do**
  $\boldsymbol{x}' \leftarrow$ Generate a solution for I_E from $\boldsymbol{x}$ by perturbation
  $y' \leftarrow 0$
  for $i \in \{1,\ldots,r\}$ **do**
    $I_i \leftarrow$ Sample a deterministic realization of I
    $y' \leftarrow y' + C(\boldsymbol{x}', I_i)$ ▹ *Evaluate* $\boldsymbol{x}'$ *once on* I_i *and* ▹ *cumulate the cost*
  end for
  $y' \leftarrow y'/r$ ▹ *Average cost of* $\boldsymbol{x}'$ *over r simulations*
  if $y' < y$ **then** ▹ *Test and replace*
    $\boldsymbol{x} \leftarrow \boldsymbol{x}'$
  end if
end while
return $\boldsymbol{x}$

3) Robust Optimization Modelling

a) Stochastic Programming Model with Recourse

CARP-SSTT following normal distributions was formulated as stochastic programming models with recourse (SPM-R), minimizing the serving cost and the expected recourse cost [20].

b) Chance Constrained Programming Model

CARP-SSTT following normal distributions was also formulated as a chance-constrained programming model (CCPM) [20]. A branch-and-cut algorithm was used for solving the CCPM of small size, aimed at minimizing the total serving cost with an upper bounded probability of extra trips for each vehicle [20]. The solutions found by the branch-and-cut algorithm were used as references for evaluating the quality of solutions found by the ALNS algorithm [20]. Eydi and Javazi [18] studied multi-commodity CARP with fuzzy demands (MOMC-CARPFD) and formulated it as a fuzzy chance constrained program (fuzzy CCP) by representing the demand of every commodity on a serving edge as a triangular fuzzy number. The proposed model was solved by a multi-objective GA [18].

c) Solving the Robust Counterpart
Babaee Tirkolaee *et al.* [37] proposed a robust optimization model for CARPSD based on Bertsimas and Sim's method [55] assuming known deviation of demands and solved it approximately using a hybrid simulated annealing algorithm. The work of [37] was further extended by taking into account the working time of drivers [38]. More recently, Babaee Tirkolaee *et al.* [39] formulated a bi-objective multi-trip periodic CARP (PCARP) with stochastic demands and used an invasive weed optimization algorithm to solve it approximately.

C. Scenario-Based Robust Optimization

In the work mentioned above, at each iteration of search, a new solution is generated for the transformed or sampled DCARP, in which the variables are set to the expectations of random variables of the corresponding UCARP or sampled values following some distribution assumptions, respectively. Wang *et al.* [10] pointed out that in real-world applications, the random variables rarely follow a specific well-formed distribution; instead, only a finite set of random realizations of a uCARP (i.e., DCARP instances) is accessible. The problem is transformed into the search of robust solutions to a given set of n DCARP instances, $\mathcal{A} = \{I_1^d, I_2^d, \ldots, I_n^d\}$, also called scenarios in decision making problems.

1) Adaptation of Meta-Heuristics

Wang *et al.* [10] adapted MAENS to the scenario-based uCARP by adding an *instance selection mechanism* and designing a new fitness function. At the beginning of search, a population is initialized for each realization, thus $|\mathcal{A}|$ populations are initialized. Then, at each iteration of search of the improved MAENS [10], the *instance selection mechanism* operates as follows: (i) an realization, I_p, is selected from $\mathcal{A}$ following a probability distribution π_p; and (ii) two distinct solutions are selected from the population of I_p as the parents for reproduction. π_p is updated periodically using the current best normalized evaluations searched for each realization. The normalized evaluation of a solution s on the realization I_p is defined as the $E_N(s, I_p) = (E(s, I_p) - C^*(I_p)) / C^*(I_p)$, where $C^*(I_p)$ is the cost of all tasks in I_p and $E(s, I_p)$ is the evaluation of solution s on I_p:

$$E(s, I_p) = TC(s, I_p) + \alpha \cdot TV(s, I_p), \quad (8)$$

where α is a user-specified parameter and is adapted during the searching process. $TV(s, I_p)$ is the total capacity violation of s and $TC(s, I_p)$ is the total cost of s simulating on the instance I_p. The new fitness function [10] is defined as

$$fitness(s) = \sum_{I_p \in \mathcal{A}} \pi_p \cdot E_n(s, I_p), \quad (9)$$

where π_p is the probability of selecting the realization $I_p \in \mathcal{A}$ by the instance selection mechanism. It is notable that in [10], though the weighted average performance over $\mathcal{A}$ is used as the fitness value, the robustness of solutions is measured using the unweighted average. Wang *et al.* [10] compared two instances of the improved MAENS using Eqs. (8) and (9) as objective function, respectively, on *ugdb* and *uval* benchmarks, and observed trade-off between robustness

Algorithm 4 Stochastic generate-and-test process without assuming known expectations of random parameters.

Require: I: A UCARP
Require: r: Resampling number
Require: π_g: Strategy for sampling a DCARP used to generate solutions
Require: π_t: Strategy for sampling a DCARP used to evaluate and test solutions
$I_s \leftarrow$ Sample an independent deterministic realization of I using π_g
$\boldsymbol{x} \leftarrow$ Generate a solution for I_s
$y \leftarrow 0$
for $i \in \{1, \ldots, r\}$ **do**
  $I_i \leftarrow$ Sample an independent deterministic realization of I using π_t
  $y \leftarrow y + C(\boldsymbol{x}, I_i)$ ▷ *Evaluate* $\boldsymbol{x}'$ *once on* I_i *and* ▷ *cumulate the cost*
end for
$y \leftarrow y/r$ ▷ *Average cost of* $\boldsymbol{x}$ *over r simulations,* ▷ *other measures in Table IV can be used*
while Stopping criteria not met **do**
  $I_s \leftarrow$ Sample an independent deterministic realization of $\boldsymbol{I}$ using π_g
  $\boldsymbol{x}' \leftarrow$ Generate a solution for I_s from $\boldsymbol{x}$ by perturbation
  $y' \leftarrow 0$
  for $i \in \{1, \ldots, r\}$ **do**
    $I_i \leftarrow$ Sample an independent deterministic realization of I using π_t
    $y' \leftarrow y' + C(\boldsymbol{x}', I_i)$ ▷ *Evaluate* $\boldsymbol{x}'$ *once on* I_i *and* ▷ *cumulate the cost*
  end for
  $y' \leftarrow y'/r$ ▷ *Average cost of* $\boldsymbol{x}'$ *over r simulations*
  if $y' < y$ **then** ▷ *Test and replace*
    $\boldsymbol{x} \leftarrow \boldsymbol{x}'$
  end if
end while
return $\boldsymbol{x}$

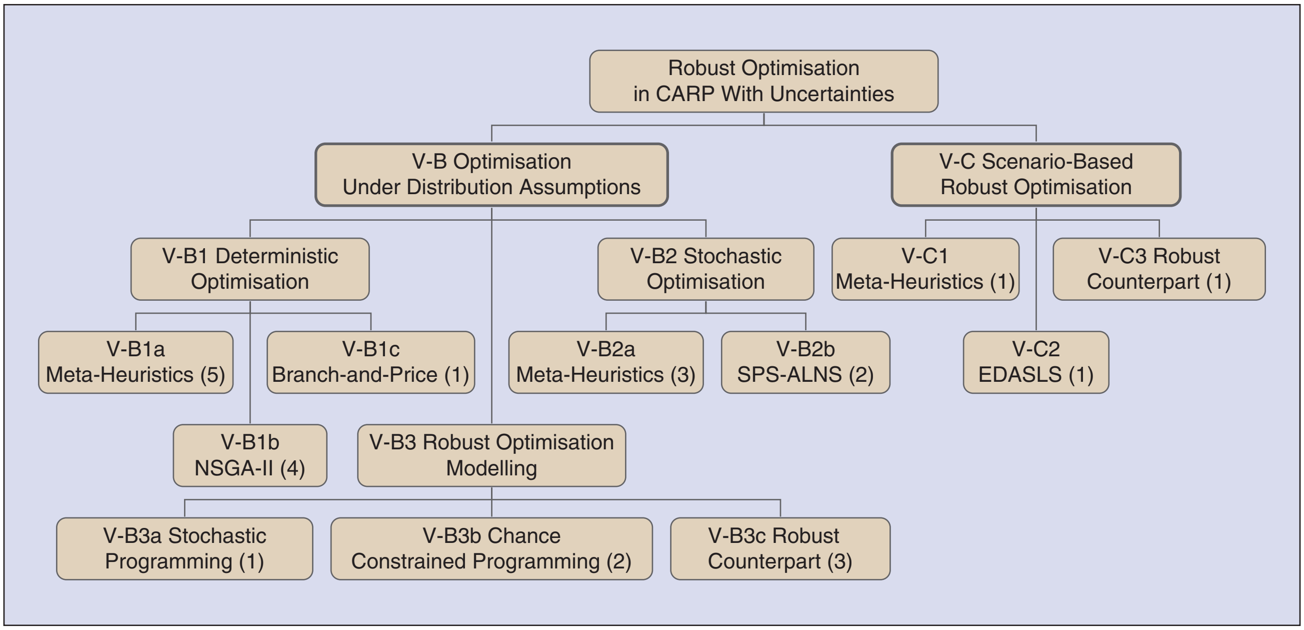

FIGURE 3 Taxonomy of robust optimization approaches for CARP with uncertainties. The number in brackets indicates the number of papers in that category.

and time consumption. The MAENS using (9) achieved more robust solutions but was more time-consuming due to the evaluation on all instances, while the MAENS using (8) was computationally faster but less robust to the perturbation of variables.

2) Estimation of Distribution Algorithm with Stochastic Local Search

Similar to [10], Wang *et al.* [17] focused on searching for robust solutions over a set of DCARP instances and proposed an Estimation of Distribution Algorithm with Stochastic Local Search (EDASLS), based on the Edge Histogram Based Sampling Algorithm (EHBSA) and a novel Stochastic Local Search (SLS) procedure which is designed to handle the uncertainties in uCARP. In [17], the worst-case performance was used when ranking solutions during search.

3) Solving the Robust Counterpart

Chen *et al.* [21] formulated the road network monitoring service problem as a CARP with stochastic service time (CARP-SST) without the assumptions of known variable distributions made in [20]. A robust optimization model was built by replacing the constraint on the stochastic service time variable by a number of constraints based on a set of scenarios with deterministic service time [21]. The proposed model aimed to optimize the worst case value over the given set of scenarios [21]. It was solved by a branch-and-cut method and the resulted solutions were evaluated via Monte Carlo simulations [21].

D. Learning Routing Policies

Very recently, several machine learning methods have been applied to learning routing policies for CARP with uncertainties (Figure 4).

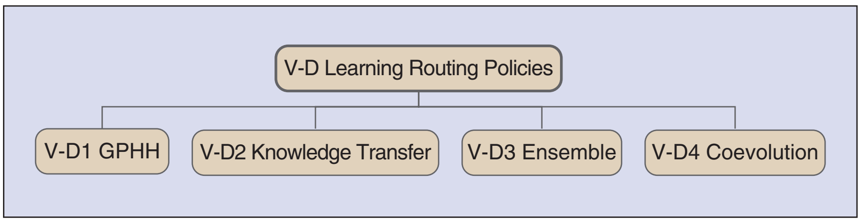

FIGURE 4 Methods for learning routing policies.

1) Genetic Programming Hyper-Heuristic

In [25], [26], the routing policy for multi-vehicle UCARP was modelled as a Lisp tree and evolved using genetic programming hyper-heuristics (GPHH). The routing policy builds the routes in parallel and it is evaluated by a discrete event simulation system, which consists of a system state and a priority queue of 3 events (refill, serve and refill-and-serve events). Mei and Zhang [26] found that the proposed GPHH generated better routing policies than the manually designed ones, but clarified that when the number of vehicles is changed, the policies need to be retrained.

MacLachlan *et al.* [27] proposed two new techniques to improve the GPHH: (i) a no-early-refill filter and (ii) a flood fill (FF) feature to better handle route failures and reduce the extra cost. Backhauls to the depot result in extra cost. If the depot is on the expected shortest path to a task to serve, the vehicle will automatically refill at the depot and then serve this task. This leads to small routes and increase in the total cost due to the backhauls to the depot. The designed no-early-refill filter excludes the tasks in which the depot is on the expected shortest path to the task. Another possible cause of backhauls to the depot is route failures. It was assumed in [6], [27] that route failures usually occur toward the end of routes. The closer the failure is to the depot, the shorter the backhaul is, and the smaller the extra cost is. To determine the tasks closed to the depot and select the next task to serve, the FF factor is defined. For each unserved task *t*, *FF*(*t*) is defined as the number of these shortest paths that *t* is a member of. Tasks with smaller FF values are preferred. No feature of the problem has been used and for each newly sampled scenario, the routing policy needs to be re-trained.

Very recently, Wang *et al.* [31], [32] noticed that the routing policies evolved by GPHH are hard to be interpreted. To evolve less complex and more interpretable routing policies, two approaches have been investigated: (i) a two-stage GPHH [31], which takes the performance as an optimization objective during the first stage, then both the performance and the tree size are optimized with a multi-objective GP proposed in [31], was designed; (ii) three ensemble methods based on GP, namely BaggingGP, BoostingGP and cooperative coevolution GP (CCGP), were proposed and compared to the simple GPHH [26], and the experimental study on the tested UCARP instances showed the potentials of CCGP on evolving more interpretable routing policies.

Almost all the work reviewed in this paper rely on the assumption that it is not possible to assign another vehicle when a route failure occurs. Thus, when a vehicle fails to serve its assigned task *t*, it should return to the depot, release its capacity and then return to the next task to serve; no other, even nearby, vehicle is capable of serving the task *t* instead. This assumption is not always true in reality. To the best of our knowledge, MacLachlan *et al.* [28] were the first to split deliveries in UCARP and proposed an enhanced GPHH with vehicle collaboration (GPHH-C). The vehicle collaboration was proved to be effective compared to the GPHH without collaboration [25] on the *ugdb*, *uval*, *uegl* benchmarks, and also to EDASLS [17] on most of the tested instances.

2) Knowledge Transfer

More recently, Ardeh *et al.* [29] assumed that the routing policies of similar scenarios share similar (sub-)trees, which could be seen as knowledge transfers among multiple routing policies.

3) Ensemble Methods

Wang *et al.* [33] proposed two novel ensemble genetic programming approaches, namely diverse bagging genetic programming (DivBaggingGP) and diverse niching genetic programming (DivNichGP), to evolve policies. The former evolves policies sequentially while the latter evolves policies in a parallel manner. An ensemble of simpler and more interpretable routing policies were evolved in [33].

4) Solution-Policy Co-Evolver

The above work focused mostly on either the evaluation of the robustness of solutions or repairing techniques (sometimes called recourse policy) which lead to a low extra cost due to additional trips (backhauls to depots). Liu *et al.* [30] proposed a new proactive-reactive approach, called solution-policy co-evolver. In [30], a solution is represented as a baseline task sequence and a recourse policy, which are evolved simultaneously in a cooperative coevolution manner by an estimation of distribution algorithm (EDA) and genetic programming (GP), respectively. This approach was only tested on the single-vehicle case [30] but could be extended to the multi-vehicle case.

E. Summary of Solution Approaches

Figures 1, 3 and 4 summarize the techniques for handling uncertainties, the robust optimization approaches for CARP with uncertainties, and the applications of machine learning methods to routing policies.

In the early attempt to solve the CARP with uncertainties, different assumptions of variable distribution have been made (Section V-A). Algorithms for solving DCARP are applied directly to the static version or deterministic realizations of a UCARP (Section V-B1) aiming at minimizing the expected cost, then the obtained solutions are evaluated on unseen samples using diverse performance metrics (Tables IV and V).

Several algorithms are adapted for solving CARP with uncertainties (Section V-B2). These approaches mostly differ in (i) the sampling methods of deterministic realizations when generating intermediate solutions during the search and (ii) using deterministic evaluation (i.e., sample UCARP once and then evaluate on the sampled DCARP) or stochastic evaluation (i.e., sample UCARP multiple times and then average the performance on the sampled scenarios) during the search. Some other work was interested in designing robust optimization models (Section V-B3).

Scenario-based robust optimization approaches were investigated without assuming variable distributions (Section V-C). As the exact values of parameters are known only at the time of execution in practice, the solutions obtained may become infeasible. For example, the vehicle capacity may be exceeded due to unexpectedly high demands of tasks. Consequently, some work focused on changing the problem formulation to reduce the probability of route failure (e.g., [6], [12]), and designing or learning effective and efficient recourse strategies (also called recourse policies or repairing operators), such as in [30]. The routing or recourse policies are often evaluated by the average performance in unseen problem instances.

Techniques for handling uncertainties, including a prior techniques, posterior techniques and resampling, can be integrated into different approaches for solving variants of CARP with uncertainties (Figure 3).

VI Discussion and Challenges

A. Difficulties in Comparing Approaches

Different approaches for handling CARP with uncertainties are rarely compared to each other in the literature, but mostly compared to some state-of-the-art approaches for solving DCARP or to some techniques proposed by the same authors, probably due to the following reasons.

1) Lack of Common Benchmarks

The proposed approaches were mostly designed and tested on different problem instances: different uncertainties, different models for uncertainties, a priori known or unknown models. Those problem instances were designed by adapting differently from well-known DCARP benchmarks or self-designed instances. As shown in Tables II, III and V, diverse variants of CARP with uncertainties have been studied, while for each variant, different assumptions on variable distributions have been made and different distributions have been used for sampling the variables. Only a few work used the benchmarks *uval*, *uegl*, *ugdb*[2]. Most work designed their own instances for testing.

2) Usage of Different Assumptions

a) A priori known or unknown models

As shown in Figure 3, most of the approaches assume certain a priori knowledge of variable distributions while some of them do not and are scenario-based (e.g., [10], [17]).

b) Assumptions on Vehicles

Besides the differences in assumptions on uncertainties, different assumptions have also been made on vehicles, which implies different designs of recourse strategies, as detailed in Section V-A2c. For instance, some work assumed that a vehicle can have at most one extra trip, while some of them do not. Only [28] considered collaboration between vehicles. In all the other work, it is assumed that when a route failure occurs, no other vehicle is able to help, and then a recourse is mandatory.

3) Usage of Different Performance Measures

Table V shows a large number of different performance measures that have been designed with particular foci and used in the optimization and evaluation processes. Although most of the work reported the average cost over a number of simulations, it is impossible to compare different approaches due to the reasons listed above. Moreover, even if a common benchmark, for example *uval*, is used, using different random seeds for sampling scenarios will introduce noise when comparing approaches. For a fair comparison, the solutions recommended by different approaches should be evaluated on an identical set of scenarios.

As a conclusion, a common benchmark for studying CARP with uncertainties is needed.

B. Scalability

Most current work focused on small-scale or medium-scale problem instances. The instances in the UCARP benchmark sets, *ugdb*, *uegl* and *uval* [9], are small compared to the problems in reality, not to mention other self-designed instances of smaller size (cf. Table III). For example, the *uval* instances contain no more than 50 vertices and 97 edges. Meanwhile, in the studies of DCARP, several sets of static CARP instances created based on real-world transport networks (e.g., Flanders district of Belgium [56], [57], Beijing and Hefei of China with up to 3584 tasks [58], [59]) have been used. Moreover, only [18], [34], [35], [53] focused on multi-objective DCARP and considered instances of small size only in their case studies. Adapting recent approaches for handling multi-objective large-scale DCARP, such as MA based on route distance grouping [56], to multi-objective large-scale UCARP is worth investigating.

C. Computation Time

As discussed previously in Section IV-G and shown in Table V, most of the reviewed work did not report the computation time. The stopping criteria were normally designed as a maximum number of iterations or when a predefined solution for a transformed DCARP was found. However, the execution time is crucial in real-world applications and it is not realistic to obtain a transformed DCARP due to complex uncertainties. In reality, we often define a maximum execution time as the budget and report the best solution found within this budget.

VII. Conclusion and Future Directions

Our review in this paper has shown that there has been a surprising broad range of issues that have been addressed by published papers on the CARP with uncertainties. There are many places where uncertainties can occur in the CARP. Various techniques and algorithms have been adapted or developed specifically for handling such uncertainties when finding a near optimal solution to the CARP. However, in spite of the breadth in research, the depth is largely lacking. There are still many open research questions that remain to be answered. This section first draws some conclusions and then points out possible future research directions.

A. Conclusion

During the past decades, CARP and its variations have been studied widely due to its large number of real-world applications. However, most of the work assumed deterministic problem instances, which is far from the reality. Till 2002, Fleury *et al.* [6] started to investigate the robustness of solutions to the CARP with stochastic demands. To the best of our knowledge, Mei *et al.* [9] was the first to propose uncertain CARPs with different random variables, including random demands, costs, presence of edges and tasks. Since then, more and more research studies have been conducted on the robust optimization of CARPs with uncertainties. This paper focuses on the robust optimization of the CARP with uncertainties and reviews the related work by discussing the modelling of uncertainties, robustness evaluation of solutions, uncertainty handling techniques and robust optimization approaches, as well as the learning of routing policies.

The core components for solving UCARP are divided into three main steps: data prediction, problem solving and decision making. Published work around the CARP with uncertainties has very different foci: problem modelling, solvers, metrics for evaluating solutions, creation of instances, generation of testing scenarios, etc. These research topics rely

[2]The Java benchmark generator for sampling UCARP instances, based on static instances, using the same variable assumptions as in [9], can be found in the GitHub project: https://github.com/meiyi1986/gpucarp.

highly on the targeted real-world applications and are often not independent of each other. The uncertainty modelling, uncertainty handling, stochastic optimization and robust decision making are all important research topics closely related to the CARP with uncertainties.

B. Future Directions

There are three main areas of future research related to the CARP with uncertainties.

1) Construction of Benchmark

It is necessary to construct a common benchmark for studying the CARP with uncertainties. Ideally, the benchmark should (i) include a set of different uncertainties and a set of candidate distributions that can be used for modelling each uncertainty; (ii) offer the possibility of adding new uncertainties and models; and finally, (iii) be constructed based on well-known DCARP benchmarks (such as *val*, *egl* and *gdb* or larger instances [58]) for an easier comparison with the deterministic case. The benchmark should be scalable so that we can test the scalability of any proposed algorithm.

2) Investigation into New Approaches

a) Simplified Assumptions on Vehicles

Current recourse strategies (except [28]) assume that assigning another vehicle is impossible or inefficient, hence only the actual capacity-violated vehicle is replanned. In reality, it is possible to allocate an additional vehicle as an alternative or ask a nearby vehicle to help.

b) Robustness Measures

Expected performance and average performance are often used in current work. However, the expectation of variables and even the distributions are mostly unknown in practice. Robustness measures based on a weighted average performance computed using Nash equilibria is worth investigating.

c) More Recommendations for Decision Makers

Besides the very few approaches using bi-objective optimization, most of the approaches recommend one solution only at the end of their execution. It is worth applying multi- or many-objective optimization with user specified preferences, or using probabilistic decision-making models for recommending solutions to generate a small set of good solutions from which human decision makers can further select a solution to execute with their expertise.

d) Hybridization with Classical Methods

Features in the UCARP instances are so far not well exploited. It would be interesting to exploit such features in new approaches and to combine problem-independent meta-heuristics, problem-specific heuristics and classical mathematical programming methods when designing new solution approaches.

e) Trade-Off Between Exploration and Exploitation

Due to the computational cost of simulating a solution on a scenario, given a fixed time budget, balancing the trade-off between the number of times a solution is simulated and the number of iterations that an algorithm executes is important for ensuring the computational efficiency in finding a near optimum. Empirical studies showed that increasing the simulation number along with the iteration number can speed up the convergence, under the assumption that the difference between solutions becomes smaller when searching is close to the optimum [60], [61]. However, in the current work, a constant value of simulations was used for reevaluating solutions. Determining the optimal simulation number for stochastic evaluation is crucial and a key topic for future research.

3) Real-World Applications

Most work in the CARP with uncertainties were experimented on artificial problems only. Few real-world applications have been implemented except for [20], [21], in which real data of road networks in Shanghai city were used. Note that not only real data is needed, and the actual road conditions and traffic rules in reality, which probably determine the design of recourse strategies, should also be considered, as examples given in [7] show: Should a vehicle always return from a node or is it allowable to take a U turn at any point along an edge; if the latter case, how to calculate the cost of its travelled time/distance on this edge; besides the cost associated to edges, is there any additional cost of returning to the depot (e.g., cost of a U turn, cost of refilling/unloading goods)? In the future, it would be interesting to apply robust optimization approaches and design of routing polices to real-world applications of the CARP with uncertainties.

Real-world problems provide a fertile ground for future research. There are many new problem formulations that we should consider in the future, for example, multiple depots [15], multiple vehicle capacities [16], and stochastic and uncertain decision variables that are, to our best knowledge, not well exploited.

Acknowledgment

This work was supported by the National Key R&D Program of China (Grant No. 2017YFC0804003), the National Natural Science Foundation of China (Grant No. 61906083), the Guangdong Provincial Key Laboratory (Grant No. 2020B121201001), the Program for Guangdong Introducing Innovative and Enterpreneurial Teams (Grant No. 2017ZT07X386), the Shenzhen Science and Technology Program (Grant No. KQTD2016112514355531), the Science and Technology Innovation Committee Foundation of Shenzhen (Grant No. JCYJ20190809121403553) and the Program for University Key Laboratory of Guangdong Province (Grant No. 2017KSYS008). The authors would like to thank all editors and reviewers for their valuable feedback and comments and Dr. Yi Mei for providing the source code of their benchmark generators [9].

References

[1] B. L. Golden and R. T. Wong, "Capacitated arc routing problems," *Networks*, vol. 11, no. 3, pp. 305–315, 1981. doi: *10.1002/net.3230110308.*
[2] S. Wøhlk, "A decade of capacitated arc routing," in *The Vehicle Routing Problem: Latest Advances and New Challenges*. New York: Springer-Verlag, 2008, pp. 29–48.
[3] H. Handa, L. Chapman, and X. Yao, "Robust salting route optimization using evolutionary algorithms," in

Evolutionary Computation in Dynamic and Uncertain Environments. New York: Springer-Verlag, 2007, pp. 497–517.
[4] M. Fadzli, N. Najwa, and M. Luis, "Capacitated arc routing problem and its extensions in waste collection," in *Proc. Int. Conf. Math., Eng. Ind. Appl. (ICoMEIA)*, AIP Publishing, 2015, vol. 1660, no. 1, p. 050001.
[5] G. B. Dantzig and J. H. Ramser, "The truck dispatching problem," *Manag. Sci*., vol. 6, no. 1, pp. 80–91, 1959. doi: *10.1287/mnsc.6.1.80*.
[6] G. Fleury, C. Prins, P. Lacomme, C. Prins, and W. R. Chérif, "Robustness evaluation of solutions for the capacitated arc routing problem," in *Proc. Conf. AI Simulation Planning High Auton. Syst*., 2002, pp. 290–295.
[7] C. H. Christiansen, J. Lysgaard, and S. Wøhlk, "A branch-and-price algorithm for the capacitated arc routing problem with stochastic demands," *Oper. Res. Lett*., vol. 37, no. 6, pp. 392–398, Nov. 2009. doi: *10.1016/j.orl.2009.05.008*.
[8] G. Laporte, R. Musmanno, and F. Vocaturo, "An adaptive large neighbourhood search heuristic for the capacitated arc-routing problem with stochastic demands," *Transp. Sci*., vol. 44, no. 1, pp. 125–135, 2010. doi: *10.1287/trsc.1090.0290*.
[9] Y. Mei, K. Tang, and X. Yao, "Capacitated arc routing problem in uncertain environments," in *Proc. IEEE Congr. Evol. Comput*., 2010, pp. 1–8.
[10] J. Wang, K. Tang, and X. Yao, "A memetic algorithm for uncertain capacitated arc routing problems," in *Proc. 2013 IEEE Workshop Memetic Comput. (MC)*, pp. 72–79. doi: *10.1109/MC.2013.6608210*.
[11] G. Fleury, P. Lacomme, and C. Prins, "Evolutionary algorithms for stochastic arc routing problems," in *Proc.* Workshops Appl. Evol. Comput., New York: Springer-Verlag, 2004, pp. 501–512.
[12] G. Fleury, P. Lacomme, and C. Prins, "Stochastic capacitated arc routing problem," Research Rep. LIMOS RR-05-12, 2005. [Online]. Available: http://www.isima.fr/limos/
[13] G. Fleury, P. Lacomme, C. Prins, and W. Ramdane-Cherif, "Improving robustness of solutions to arc routing problems," *J. Oper. Res. Soc*., vol. 56, no. 5, pp. 526–538, 2005. doi: *10.1057/palgrave.jors.2601822*.
[14] V. Pillac, M. Gendreau, C. Guéret, and A. L. Medaglia, "A review of dynamic vehicle routing problems," *Eur. J. Oper. Res*., vol. 225, no. 1, pp. 1–11, 2013. doi: *10.1016/j.ejor.2012.08.015*.
[15] L. Xing, P. Rohlfshagen, Y. Chen, and X. Yao, "An evolutionary approach to the multidepot capacitated arc routing problem," *IEEE Trans. Evol. Comput*., vol. 14, no. 3, pp. 356–374, 2009. doi: *10.1109/TEVC.2009.2033578*.
[16] H. Handa, L. Chapman, and X. Yao, "Robust route optimization for gritting/salting trucks: A cercia experience," *IEEE Comput. Intell. Mag*., vol. 1, no. 1, pp. 6–9, 2006. doi: *10.1109/MCI.2006.1597056*.
[17] J. Wang, K. Tang, J. A. Lozano, and X. Yao, "Estimation of the distribution algorithm with a stochastic local search for uncertain capacitated arc routing problems," *IEEE Trans. Evol. Comput*., vol. 20, no. 1, pp. 96–109, 2015. doi: *10.1109/TEVC.2015.2428616*.
[18] A. Eydi and L. Javazi, "Model and solution approach for multi objective-multi commodity capacitated arc routing problem with fuzzy demand," *J. Ind. Syst. Eng*., vol. 5, no. 4, pp. 208–229, 2012.
[19] E. Babaee Tirkolaee, I. Mahdavi, M. M. Seyyed Esfahani, and G.-W. Weber, "A hybrid augmented ant colony optimization for the multi-trip capacitated arc routing problem under fuzzy demands for urban solid waste management," *Waste Manag. Res*., p. 0734242X19865782. doi: *10.1177/0734242X19865782*.
[20] L. Chen, M. H. Hà, A. Langevin, and M. Gendreau, "Optimizing road network daily maintenance operations with stochastic service and travel times," *Transp. Res. E, Logist, Transp. Rev*., vol. 64, pp. 88–102, 2014. doi: *10.1016/j.tre.2014.02.002*.
[21] L. Chen, M. Gendreau, M. H. Hà, and A. Langevin, "A robust optimization approach for the road network daily maintenance routing problem with uncertain service time," *Transp. Res. E, Logist. Transp. Rev*., vol. 85, pp. 40–51, 2016. doi: *10.1016/j.tre.2015.11.006*.
[22] Y. Mei, K. Tang, and X. Yao, "Evolutionary computation for dynamic capacitated arc routing problem," in *Evolutionary Computation for Dynamic Optimization Problems*. New York: Springer-Verlag, 2013, pp. 377–401.
[23] J. S. DeArmon, "A comparison of heuristics for the capacitated Chinese postman problem," Master's thesis, Univ. Maryland, College Park, MD, 1981.
[24] E. Benavent, V. Campos, A. Corberán, and E. Mota, "The capacitated arc routing problem: Lower bounds," *Networks*, vol. 22, no. 7, pp. 669–690, 1992. doi: *10.1002/net.3230220706*.
[25] Y. Liu, Y. Mei, M. Zhang, and Z. Zhang, "Automated heuristic design using genetic programming hyper-heuristic for uncertain capacitated arc routing problem," in *Proc. Genetic Evol. Comput. Conf. Companion*, ACM, 2017, pp. 290–297. doi: *10.1145/3071178.3071185*.
[26] Y. Mei and M. Zhang, "Genetic programming hyper-heuristic for multi-vehicle uncertain capacitated arc routing problem," in *Proc. Genetic Evol. Comput. Conf. Companion*, ACM, 2018, pp. 141–142. doi: *10.1145/3205651.3205661*.
[27] J. MacLachlan, Y. Mei, J. Branke, and M. Zhang, "An improved genetic programming hyper-heuristic for the uncertain capacitated arc routing problem," in *Proc. Australas.Joint Conf. Artif. Intell*., New York: Springer-Verlag, 2018, pp. 432–444.
[28] J. MacLachlan, Y. Mei, J. Branke, and M. Zhang, "Genetic programming hyper-heuristic with vehicle collaboration for uncertain capacitated arc routing problem," *Evol. Comput*., pp. 1–31, 2019. doi: *10.1162/evco_a_00267*.
[29] M. A. Ardeh, Y. Mei, and M. Zhang, "Genetic programming hyper-heuristic with knowledge transfer for uncertain capacitated arc routing problem," in *Proc. Genetic Evol. Comput. Conf. Companion*, ACM, 2019, pp. 334–335. doi: *10.1145/3319619.3321988*.
[30] Y. Liu, Y. Mei, M. Zhang, and Z. Zhang, "A predictive-reactive approach with genetic programming and cooperative co-evolution for uncertain capacitated arc routing problem," *Evol. Comput*., vol. 28, no. 2, pp. 289–316, 2020. doi: *10.1162/evco_a_00256*.
[31] S. Wang, Y. Mei, J. Park, and M. Zhang, "A two-stage genetic programming hyper-heuristic for uncertain capacitated arc routing problem," in Proc. *2019 IEEE Symp. Series Comput. Intell.,* pp. 1606–1613.
[32] Wang, Y. Mei, J. Park, and M. Zhang, "Evolving ensembles of routing policies using genetic programming for uncertain capacitated arc routing problem," in Proc. *2019 IEEE Symp. Comput. Intell*., pp. 1628–1635.
[33] S. Wang, Y. Mei, and M. Zhang, "Novel ensemble genetic programming hyper-heuristics for uncertain capacitated arc routing problem," in *Proc. Genetic Evol. Comput. Conf.,* ACM, 2019, pp. 1093–1101. doi: *10.1145/3321707.3321797*.
[34] G. Fleury, P. Lacomme, C. Prins, and M. Sevaux, "A memetic algorithm for a bi-objective and stochastic CARP," in *Proc. 6th Metaheuristics Int. Conf. (MIC 2005)*, 2005, pp. 22–26.
[35] G. Fleury, P. Lacomme, C. Prins, and M. Sevaux, "A bi-objective stochastic approach for stochastic CARP," Tech. Rep., Oct. 2008. [Online]. Available: https://hal.archives-ouvertes.fr/hal-00678494
[36] S. González, D. Riera, A. A. Juan, M. G. Elizondo, and P. Fonseca, "Sim-randsharp: A hybrid algorithm for solving the arc routing problem with stochastic demands," in *Proc. Winter Simulation Conf., ser. WSC '12*, 2012, pp. 275:1–275:11. [Online]. Available: http://dl.acm.org/citation.cfm?id=2429759.2430130
[37] E. Babaee Tirkolaee, M. Alinaghian, M. Bakhshi Sasi, and M. Seyyed Esfahani, "Solving a robust capacitated arc routing problem using a hybrid simulated annealing algorithm: a waste collection application," *J. Ind. Eng. Manag. Stud*., vol. 3, no. 1, pp. 61–76, 2016.
[38] E. Babaee Tirkolaee, I. Mahdavi, and M. M. S. Esfahani, "A robust periodic capacitated arc routing problem for urban waste collection considering drivers and crew's working time," *Waste Manag*., vol. 76, pp. 138–146, 2018. doi: *10.1016/j.wasman.2018.03.015*.
[39] E. Babaee Tirkolaee, A. Goli, M. Pahlevan, and R. Malekalipour Kordestanizadeh, "A robust bi-objective multi-trip periodic capacitated arc routing problem for urban waste collection using a multi-objective invasive weed optimization," *Waste Manag. Res*., vol. 37, no. 11, pp. 1089–1101, 2019. doi: 10.1177/0734242X19865340.
[40] R. W. Eglese, "Routeing winter gritting vehicles," *Discrete Appl. Math*., vol. 48, no. 3, pp. 231–244, 1994. doi: *10.1016/0166-218X(92)00003-5*.
[41] A. Wald, "Contributions to the theory of statistical estimation and testing hypotheses," *Ann. Math. Statist*., vol. 10, no. 4, pp. 299–326, 1939. doi: *10.1214/aoms/1177732144*.
[42] W.-L. Pearn, A. Assad, and B. L. Golden, "Transforming arc routing into node routing problems," *Comput. Oper. Res*., vol. 14, no. 4, pp. 285–288, 1987. doi: *10.1016/0305-0548(87)90065-7*.
[43] C. Bastian and A. H. R. Kan, "The stochastic vehicle routing problem revisited," *Eur. J. Oper. Res*., vol. 56, no. 3, pp. 407–412, 1992.
[44] M. Dror, "Modeling vehicle routing with uncertain demands as a stochastic program: Properties of the corresponding solution," *Eur. J. Oper. Res*., vol. 64, no. 3, pp. 432–441, 1993. doi: *10.1016/0377-2217(93)90132-7*.
[45] M. Gendreau, G. Laporte, and R. Séguin, "Stochastic vehicle routing," *Eur. J. Oper. Res*., vol. 88, no. 1, pp. 3–12, 1996. doi: *10.1016/0377-2217(95)00050-X*.
[46] Y. Jin and J. Branke, "Evolutionary optimization in uncertain environments-a survey," *IEEE Trans. Evol. Comput*., vol. 9, no. 3, pp. 303–317, 2005. doi: *10.1109/TEVC.2005.846356*.
[47] P. Rakshit, A. Konar, and S. Das, "Noisy evolutionary optimization algorithms–a comprehensive survey," *Swarm Evol. Comput*., vol. 33, pp. 18–45, 2017. doi: 10.1016/j.swevo.2016.09.002.
[48] J. Liu, D. L. St-Pierre, and O. Teytaud, "A mathematically derived number of resamplings for noisy optimization," in Proc. Genetic Evol. Comput. Conf. Companion, New York: ACM, 2014, pp. 61–62.
[49] J. Liu and O. Teytaud, "A simple yet effective resampling rule in noisy evolutionary optimization," in *Proc. 2019 IEEE Symp. Series Comput. Intell*., pp. 689–696.
[50] P. Lacomme, C. Prins, and W. Ramdane-Cherif, "Competitive memetic algorithms for arc routing problems," *Ann. Oper. Res*., vol. 131, no. 1–4, pp. 159–185, 2004. doi: *10.1023/B:ANOR.0000039517.35989.6d*.
[51] K. Tang, Y. Mei, and X. Yao, "Memetic algorithm with extended neighborhood search for capacitated arc routing problems," *IEEE Trans. Evol. Comput*., vol. 13, no. 5, pp. 1151–1166, Oct. 2009.
[52] Y. Mei, K. Tang, and X. Yao, "A global repair operator for capacitated arc routing problem," *IEEE Trans. Syst., Man, Cybern. B. Cybern*., vol. 39, no. 3, pp. 723–734, 2009.
[53] P. Lacomme, C. Prins, and M. Sevaux, "A genetic algorithm for a bi-objective capacitated arc routing problem," *Comput. Oper. Res*., vol. 33, no. 12, pp. 3473–3493, 2006. doi: *10.1016/j.cor.2005.02.017*.
[54] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput*., vol. 6, no. 2, pp. 182–197, 2002. doi: *10.1109/4235.996017*.
[55] D. Bertsimas and M. Sim, "The price of robustness," *Oper. Res*., vol. 52, no. 1, pp. 35–53, 2004. doi: *10.1287/opre.1030.0065*.
[56] R. Shang, K. Dai, L. Jiao, and R. Stolkin, "Improved memetic algorithm based on route distance grouping for multiobjective large scale capacitated arc routing problems," *IEEE Trans. Cybern*., vol. 46, no. 4, pp. 1000–1013, 2016. doi: *10.1109/TCYB.2015.2419276*.
[57] R. Shang, B. Du, K. Dai, L. Jiao, A. M. Ghalamzan Esfahani, and R. Stolkin, "Quantum-inspired immune clonal algorithm for solving large-scale capacitated arc routing problems," *Memetic Comput*., vol. 10, pp. 81–102, 2018. doi: *10.1007/s12293-017-0224-7*.
[58] K. Tang, J. Wang, X. Li, and X. Yao, "A scalable approach to capacitated arc routing problems based on hierarchical decomposition," *IEEE Trans. Cybern*., vol. 47, no. 11, pp. 3928–3940, 2017. doi: *10.1109/TCYB.2016.2590558*.
[59] J. Liu and X. Yao, "Self-adaptive decomposition and incremental hyperparameter tuning across multiple problems," in *Proc. 2019 IEEE Symp. Series Comput. Intell*., pp. 1590–1597.
[60] S. Astete-Morales, J. Liu, and O. Teytaud, "Noisy optimization convergence rates," in *Proc. Genetic Evol. Comput. Conf. Companion*, ACM, 2013, pp. 223–224.
[61] C. Qian, Y. Yu, and Z.-H. Zhou, "Analyzing evolutionary optimization in noisy environments," *Evol. Comput*., vol. 26, no. 1, pp. 1–41, 2018. doi: *10.1162/evco_a_00170*.