Uncertain multilevel programming with application to omni-channel vehicle routing problem

Rong Gao1 · Yebao Ma1 · Dan A. Ralescu2

Received: 31 March 2022 / Accepted: 12 September 2022 / Published online: 24 September 2022
© The Author(s), under exclusive licence to Springer-Verlag GmbH Germany, part of Springer Nature 2022

Abstract
Multilevel programming is widely applied to solve decentralized decision-making problems. In practice, indeterminacies are presented in these problems due to volatile factors or emergencies. As a type of indeterminacy, uncertainty is introduced in multilevel programming. For resolving multilevel programming problems with uncertain parameters, this paper constructs the uncertain expected value multilevel programming model and chance-constrained multilevel programming. Then, these models are converted to their equivalent forms. Moreover, the Stackelberg-Nash equilibrium solutions are obtained by using a genetic algorithm. Finally, these models are applied to the omni-channel vehicle routing problem, and a numerical experiment is given. The numerical experiment shows that the established models can optimize the distribution efficiency by coordinating the interests of decision-makers.

Keywords Multilevel programming · Uncertain variable · Omni-channel vehicle routing

1 Introduction
Multilevel programming (MP) is used for connecting each subsystem within a decentralized decision-making system (DDS). Such a system has extensively appeared in numerous large-scale and complicated decision-making problems confronted by government agencies and corporate organizations. Specifically, the MP problem essentially means the upper level decision-maker (the leader) has the authority to control or coordinate the lower level decision-makers (the followers). The leader first makes a decision based on his own constraints to optimize the objective which is dependent on the response of the followers. Then the followers make their decisions and achieve their own goals on the premise of ensuring the benefit of the upper authority. MP originates in two domains: game theory and mathematical programming (Bracken and McGill 1973). Since then, MP has been applied to many scenarios, such as supply chain management (Fathollahi-Fard et al. 2022) and electricity distribution economics (Wei et al. 2021).

Due to the real-life complexity of the social and economic environment, various indeterminate factors should be taken into account. These indeterminate factors are described by random variables in probability theory. For example, Patriksson and Wynter (1999) proposed stochastic bilevel programming which is in fact the expected value bilevel model. Patriksson and Wynter (1999) focused on solving stochastic hierarchical decision-making problems, where the leader controls or coordinates only one follower. Gao et al. (2004) considered multiple followers, and established the expected value multilevel programming (EVMP) and chance-constrained multilevel programming (CCMP), respectively. Gao (2004) also developed a general model for dependent-chance multilevel programming (DCMP) in the stochastic environment. The utility of the above models has been proved in practical application problems. Rahmani and Hosseini (2021) applied a stochastic EVMP model to solving an inventory competitive facility location problem. Ma et al. (2020) employed a stochastic CCMP method to design a water-food-ecology interaction mechanism. Fan and Cheng (2014) took advantage of a stochastic DCMP model to resolve the transmission network planning problem.

However, when samples are unavailable, it is too difficult to find the statistical regularity, as well as ensure that the...
estimated probability distribution adequately approaches the frequency of indeterminate quantity (Liu 2015). At this moment, we have few choices but to request assistance from belief degrees provided by domain experts. To better overcome such a dilemma of indeterminacy, one mathematical system can be employed, uncertainty theory (Liu 2015), which is used to describe indeterminate phenomena associated with experts’ belief degrees. In the past few years, researches on uncertainty theory have substantially increased in various scenarios, such as supply chain management (Ma et al. 2020), renewal reward process (Yao and Zhou 2018), uncertain reliability analysis (Ahmadzade and Gao 2018), operational law (Wang et al. 2018), uncertain differential equation (Ji and Zhou 2018) and uncertain wave equation (Gao and Ralescu 2019).

Mathematical programming is considered as uncertain programming (UP) when it contains uncertain variables (Liu 2015). In recent years, UP has been well developed in applications, such as black transport network design problem (Hosseini and Wadbro 2022) and minimum spanning tree problem (Majumder et al. 2022). With respect to dealing with the uncertainty in a DDS, Liu and Yao (2015) proposed a hybrid uncertain multilevel programming (UMP) model, where expected values of objective functions are maximized under the corresponding chance constraints for decision-makers in all levels. Thereafter, Ning and Su (2017) combined the uncertain multilevel approach and the classic vehicle routing problem with time windows. Jalil et al. (2018) presented a multilevel solid transportation problem with uncertain parameters.

A few existing MP models are proposed with uncertain factors, while many of them are under random or fuzzy environment. However, there are shortcomings in the applicability of stochastic MP and fuzzy MP. For example, probability theory relies on historical data heavily. Fuzzy set theory lacks self-duality, and the possibilities of two opposite events do not add up to 1. In addition, there are limited types of UMP models as well as finite application scenarios. For example, commuters in traffic assignment problems might seek to reach the destination on schedule (Ke et al. 2017), and the model built by Liu and Yao (2015) is improper for this case. For another instance, decision-makers of transportation companies or hotels may prefer the CCMP model to optimize their profits considering the impact of COVID-19 under uncertain environment. Inspired by such research status, EVMP and CCMP models need to be established to enhance their applicability.

This study differs from current literatures because of the following contributions. For one thing, this paper extends the study of Liu and Yao (2015) and suggests two uncertain multilevel models called EVMP and CCMP as well as their deterministic equivalents. For another, we apply proposed models to an omni-channel vehicle routing problem (OVRP), which has not yet been studied via the UMP approach. The proposed models are resolved by a genetic algorithm. The remainder of the study is constructed as follows. Related works are introduced in Sect. 2. Section 3 lists some preliminary knowledge of uncertainty theory. In Sect. 4, the UMP models are respectively built via the EVP and CCP approaches, and then the corresponding deterministic equivalents are presented. In Sect. 5, we introduce a computational approach of the Nash equilibrium for the multiple followers and a genetic algorithm to find the Stackelberg-Nash equilibrium of each launched uncertain multilevel model. And Sect. 6 gives the applications of proposed models in OVRP to evince the availability of the designed formulation and solution procedure. Last but not least, Sect. 7 provides a brief summary.

2 Related work

This section will introduce the related works containing UMP and OVRP.

2.1 Uncertain multilevel programming

MP was first proposed by Bracken and McGill (1973) to solve decentralized decision-making problems in a hierarchical system. So far, MP has been well applied to many real-life scenarios, such as missile defense (Haywood et al. 2022). Furthermore, since MP is NP-hard (Ben-Ayed and Blair 1990), scholars have introduced many algorithms, such as evolutionary algorithm (Abo-Elnaga and Nasr 2022) and branch-and-bound algorithm (Liu et al. 2021).

For handling MP with random parameters, Patriksson and Wynter (1999), Gao et al. (2004) and Gao (2004) studied three commonly used modeling techniques which are EVP, CCP and dependent-chance programming (DCP). After that, Huang and Ke (2016) built a hybrid MP model, where the random variables are handled by EVP for the leader but DCP for the followers. Roghanian et al. (2007) stated the stochastic multilevel multiobjective programming. Moreover, scholars have introduced plenty of applications such as facility location (Rahmani and Hosseini 2021) and energy dispatch (Su et al. 2022).

For dealing with uncertain variables in MP, Liu and Yao (2015) developed an UMP model. Thereafter, Xue et al. (2020) introduced the UMP into the bus line distribution
problem. Guo et al. (2022) suggested a vehicle scheduling method for electric buses based on the UMP approach.

Existing researches on MP lacks sufficient consideration of uncertain coefficients. Specifically, there are EVP, CCP and DCP in UP, which are not completely studied in UMP. Moreover, UMP needs more research on algorithms such as ant colony optimization (Ramamoorthy and Thangavelu 2022c), and applications such as vehicular ad-hoc networks (Ramamoorthy and Thangavelu 2022b) and OVRP (Abdulkader et al. 2018).

2.2 Omni-channel vehicle routing problem

The vehicle routing problem (VRP) aims at designing the least-cost delivery routes from a depot to scattered nodes under some constraints. By now, many literatures on VRP have emerged, such as open VRP (Dutta et al. 2022a) and green VRP (Dutta et al. 2022b).

The OVRP (Abdulkader et al. 2018) is actually a rich vehicle routing problem under the omni-channel retailing scenario. In addition, Dethlefs et al. (2022) combined the rapid integrated order fulfillment and multi-depot VRP for omni-channel retailing. Schubert et al. (2021) integrated the order picking problem and VRP for same-day delivery. Janjevic et al. (2021) established a three-tiered supply chain network for omni-channel retail. However, the coordination of multiple channels cannot be achieved overnight. Guo et al. (2021) employed an auction mechanism for omni-channel on-demand logistics. There also exist studies on OVRP containing random variables. For example, Martins et al. (2020) proposed a stochastic OVRP model with the random vehicle travel time.

The related researches on OVRP is currently lacking consideration for more types of indeterminacies such as fuzzyness (Barma et al. 2021) and uncertainty (Ning and Su 2017), and modelling techniques such as MP (Guo et al. 2022) and multiobjective programming (Barma et al. 2022). Additionally, more algorithms can be applied to solving OVRP, such as bio-inspired routing algorithm (Ramamoorthy and Thangavelu 2022a), ant colony optimization (Ramamoorthy and Thangavelu 2022b), and genetic algorithm (Katoch et al. 2021).

3 Preliminary

This part of the paper briefly introduces some fundamental definitions and properties in uncertainty theory and MP.

3.1 Uncertainty theory

Suppose that \( I \) is a nonempty set and let \( L \) represent a \( \sigma \)-algebra over \( I \). Any element \( A \) in \( L \) is an event. However, we pay more attention to the belief degree of the event \( A \) happening. To cope with belief degrees, Liu (2015) presented the definition of uncertain measure.

Definition 2.1 (Liu 2015) The set function \( M \) is called an uncertain measure if it satisfies

\[
M\left( \bigcup_{i=1}^{\infty} A_i \right) \leq \sum_{i=1}^{\infty} M\{A_i\};
\]

where \( A^c \) is the opposite event of \( A \).

Suppose that \( \xi \) is an uncertain variable with uncertainty distribution \( \Phi(x) \) which is defined by \( \Phi(x) = M\{\xi \leq x\} \). If \( \Phi(x) \) has the unique inverse function \( \Phi^{-1}(a) \) for each \( a \in (0, 1) \), then the uncertainty distribution \( \Phi(x) \) becomes regular.

Definition 2.2 (Liu 2015) The uncertain variables \( \xi_1, \ldots, \xi_n \) are thought to be independent if

\[
M\left( \bigcap_{i=1}^{n} \{\xi_i \in B_i\} \right) = \prod_{i=1}^{n} M\{\xi_i \in B_i\}
\]

for any Borel sets \( B_1, \ldots, B_n \).

Theorem 2.1 (Liu 2015) Suppose \( \xi_1, \ldots, \xi_n \) are independent uncertain variables and have uncertainty distributions \( \Phi_1, \ldots, \Phi_n \), respectively. Assume that \( f(x_1, \ldots, x_n) \) is a real continuous function, which monotonously increases as \( x_1, \ldots, x_m \) increase and monotonously decreases as \( x_{m+1}, \ldots, x_n \) decrease. Then \( \xi = f(\xi_1, \ldots, \xi_n) \) is an uncertain variable and has an inverse uncertainty distribution

\[
\Phi^{-1}(a) = f(\Phi_1^{-1}(a), \ldots, \Phi_n^{-1}(a)),
\]

where \( a \) is a real number and \( 0 \leq a \leq 1 \).

Definition 2.3 (Liu 2015) The expected value of an uncertain variable \( \xi \) is defined by

\[
E[\xi] = \int_{-\infty}^{0} M\{\xi \leq x\} dx - \int_{0}^{\infty} M\{\xi \geq x\} dx
\]

on the premise that one integral is finite at least.
4 Uncertain multilevel programming

Consider a two-level DDS with one leader and \( m \) followers, and the upper and lower levels have their own decisions and objectives. A simple instance of the DDS is shown in Fig. 1.

To better meet the different needs in practical circumstances, this section will respectively introduce the EVMP and CCMP under uncertain environment.

4.1 Expected value multilevel programming

In UP, the objective and the constraints cannot be directly ranked as both of them contain uncertain variables. One method is to compare the corresponding expected values. The expected value model is the mathematical programming where both the uncertain objective function and the uncertain constraints are handled by the expected value approach. Based on this point of view, we propose an expected value uncertain multilevel programming (EVUMP) model.

Firstly, the upper level authority makes the decision to minimize the leader’s objective function which is dependent on the response of the followers. Then, after observing the leader’s control vector, the followers generate their own decision vectors to achieve their own goals on the premise of ensuring the benefit of the upper authority. Then Ke et al. (2017) listed the following general MP model

\[
\begin{align*}
\min_{x} & \quad F(x, y^*_1, \ldots, y^*_m) \\
\text{s.t.} & \quad G(x) \leq 0 \\
& \quad (y^*_1, \ldots, y^*_m) \text{ resolves the following subproblems (i = 1, 2, \ldots, m)} \\
& \quad \min_{y_i} f_i(x, y^*_1, \ldots, y^*_m) \\
& \quad \text{s.t.} \\
& \quad g_i(x, y^*_1, \ldots, y^*_m) \leq 0,
\end{align*}
\]

where \( x \) is the leader’s control vector, \( y_i \) is the \( i \)th follower’s control vector, \( F(x, y^*_1, \ldots, y^*_m) \) is the leader’s objective, \( G(x) \leq 0 \) is the leader’s constraint, \( f_i(x, y^*_1, \ldots, y^*_m) \) is the \( i \)th follower’s objective, \( g_i(x, y^*_1, \ldots, y^*_m) \leq 0 \) is the \( i \)th follower’s constraint, and \( i = 1, \ldots, m \).

4.1 Expected value multilevel programming

In UP, the objective and the constraints cannot be directly ranked as both of them contain uncertain variables. One method is to compare the corresponding expected values. The expected value model is the mathematical programming where both the uncertain objective function and the uncertain constraints are handled by the expected value approach. Based on this point of view, we propose an expected value uncertain multilevel programming (EVUMP) model.

Firstly, the upper level authority makes the decision to minimize the leader’s objective function which is dependent on the response of the followers. Then, after observing the leader’s control vector, the followers generate their own decision vectors to achieve their own goals on the premise of ensuring the benefit of the upper authority. Then we have the following EVUMP model

\[
\begin{align*}
\min_{x} & \quad E[F(x, y^*_1, \ldots, y^*_m, \xi)] \\
\text{s.t.} & \quad E[G(x, \xi)] \leq 0 \\
& \quad (y^*_1, \ldots, y^*_m) \text{ resolves the following subproblems (i = 1, 2, \ldots, m)} \\
& \quad \min_{y_i} E[f_i(x, y^*_1, \ldots, y^*_m, \xi)] \\
& \quad \text{s.t.} \\
& \quad E[g_i(x, y^*_1, \ldots, y^*_m, \xi)] \leq 0,
\end{align*}
\]

where \( x \) is the leader’s control vector, \( y_i \) is the \( i \)th follower’s control vector, \( \xi = (\xi_1, \ldots, \xi_m) \) is the uncertain vector with independent uncertain variables \( \xi_1, \ldots, \xi_m \), \( F(x, y^*_1, \ldots, y^*_m, \xi) \) is the leader’s objective, \( G(x, \xi) \leq 0 \) is the leader’s constraint, \( f_i(x, y^*_1, \ldots, y^*_m, \xi) \) is the \( i \)th follower’s objective, \( g_i(x, y^*_1, \ldots, y^*_m, \xi) \leq 0 \) is the \( i \)th follower’s constraint, and \( i = 1, \ldots, m \).

In Model (7), \( F, G, f_i \) and \( g_i \) (\( i = 1, \ldots, m \)) are thought to be uncertain variables caused by the uncertain vector \( \xi \). This
makes ranking difficult, so we optimize the corresponding expected value.

**Definition 3.1.1** For a feasible decision $x$ of the upper level, the feasible array $(y^*_1, \ldots, y^*_m)$ is said to be a Nash equilibrium of the lower level if it satisfies

$$E[f_i(x, y^*_1, \ldots, y^*_{i-1}, y_i, y^*_{i+1}, \ldots, y^*_m, \xi_1, \ldots, \xi_n)] \geq E[f_i(x, y^*_1, \ldots, y^*_m, \xi_1, \ldots, \xi_n)]$$

(8)

on the premise that $(y^*_1, \ldots, y^*_{i-1}, y_i, y^*_{i+1}, \ldots, y^*_m)$ is feasible for all $i = 1, 2, \ldots, m$.

**Definition 3.1.2** The array $(x^*, y^*_1, \ldots, y^*_m)$ is said to be a Stackelberg-Nash equilibrium to Model (7) if there exists a feasible control vector $x^*$ of the upper level such that

$$E[F(x^*, y^*_1, \ldots, y^*_m, \xi_1, \ldots, \xi_n)]$$

(9)

holds for any feasible decision $\bar{x}$, where $(\bar{y}_1, \ldots, \bar{y}_m)$ and $(y^*_1, \ldots, y^*_m)$ are the followers' Nash equilibrium solutions and respectively dependent on $\bar{x}$ and $x^*$.

Now, we transform Model (7) into its crisp equivalent form by using Theorem 2.1 and Theorem 2.2.

Suppose that $F$ is a real continuous function, which monotonously increases as $\xi_1, \ldots, \xi_k$ increase and monotonously decreases as $\xi_{k+1}, \ldots, \xi_n$ decrease. In addition, $\xi_1, \ldots, \xi_n$ are independent uncertain variables and have uncertainty distributions $\Phi_1, \ldots, \Phi_n$, respectively. Then $E[F(x, y^*_1, \ldots, y^*_m, \xi_1, \ldots, \xi_n)]$ equals to

$$\int_0^1 F(x, y^*_1, \ldots, y^*_m, \Phi_1^{-1}(\eta), \ldots, \Phi_k^{-1}(\eta), \Phi_{k+1}^{-1}(1 - \eta), \ldots, \Phi_n^{-1}(1 - \eta)) \, d\eta.$$  

(10)

Assume that $f_i$ is a real continuous function, which monotonously increases as $\xi_1, \ldots, \xi_k$ increase and monotonously decreases as $\xi_{k+1}, \ldots, \xi_n$ decrease, and $\xi_1, \ldots, \xi_n$ are independent. Then $E[f_i(x, y_1, \ldots, y_m, \xi_1, \ldots, \xi_n)]$ equals to

$$\int_0^1 f_i(x, y_1, \ldots, y_m, \Phi_1^{-1}(\eta), \ldots, \Phi_{k+1}^{-1}(1 - \eta), \ldots, \Phi_n^{-1}(1 - \eta)) \, d\eta.$$  

(11)

where $i = 1, 2, \ldots, m$.

Suppose that $G$ is a real continuous function, which monotonously increases as $\xi_1, \ldots, \xi_k$ increase and monotonously decreases as $\xi_{k+1}, \ldots, \xi_n$ decrease, and $\xi_1, \ldots, \xi_n$ are independent. Then $E[G(x, \xi_1, \ldots, \xi_n)] \leq 0$ equals to

$$\int_0^1 G(x, \Phi_1^{-1}(\eta), \ldots, \Phi_n^{-1}(\eta), \Phi_{k+1}^{-1}(1 - \eta), \ldots, \Phi_n^{-1}(1 - \eta)) \, d\eta \leq 0.$$  

(12)

Suppouse that $g_i$ is a real continuous function, which monotonously increases as $\xi_1, \ldots, \xi_n$ increase and monotonously decreases as $\xi_{k+1}, \ldots, \xi_n$ decrease, and $\xi_1, \ldots, \xi_n$ are independent. Then $E[g_i(x, y_1, \ldots, y_m, \xi_1, \ldots, \xi_n)] \leq 0$ is equivalent to

$$\int_0^1 g_i(x, y_1, \ldots, y_m, \Phi_1^{-1}(\eta), \ldots, \Phi_n^{-1}(\eta), \Phi_{k+1}^{-1}(1 - \eta), \ldots, \Phi_n^{-1}(1 - \eta)) \, d\eta \leq 0,$$

(13)

where $i = 1, 2, \ldots, m$.

Then Model (7) is equivalent to Model (15).

### 4.2 Chance-constrained multilevel programming

CCP remains an effective tool for UP. The core principal of CCP is that control vectors are permitted to mismatch the constraint conditions in a way, but should at least make the constraint conditions hold with a confidence level. This confidence level actually reflects the decision-maker’s desire that the uncertain constraints hold. Motivated by this ideology, we construct the following chance-constrained uncertain multilevel programming (CCUMP).

Firstly, the upper level authority makes the decision to minimize the leader’s objective function which is dependant on the response of the followers. Then, after observing the leader’s control vector, the followers generate their own decision vectors to achieve their own goals on the premise of ensuring the benefit of the upper authority. Then we have a CCUMP model as follows

\[ \begin{align*}
\min \bar{F} \\
\text{s.t.} \\
M \{ F(x, y^*_1, \ldots, y^*_m, \xi) \leq \bar{F} \} \geq a_0 \\
M \{ G(x, \xi) \leq 0 \} \geq \beta_0 \\
(y^*_1, \ldots, y^*_m) \text{ resolves the following subproblems } (i = 1, \ldots, m) \\
\min f_i(y^*_i) \\
\text{s.t.} \\
M \{ f_i(x, y_1, \ldots, y_m, \xi) \leq \bar{F}_i \} \geq a_i \\
M \{ g_i(x, y_1, \ldots, y_m, \xi) \leq 0 \} \geq \beta_i,
\end{align*} \]

where $x$ is the leader’s control vector, $y_i$ is the $i$th follower’s control vector, $\xi = (\xi_1, \ldots, \xi_n)$ is the uncertain vector with
independent uncertain variables $\xi_1, \ldots, \xi_n, F(x, y_1^*, \ldots, y_m^*, \xi)$ is the leader’s objective, $G(x, \xi) \leq 0$ is the leader’s constraint, $f_i(x, y_1^*, \ldots, y_m^*, \xi)$ is the $i$th follower’s objective, $g_i(x, y_1^*, \ldots, y_m^*, \xi) \leq 0$ is the $i$th follower’s constraint, $\bar{F}$ and $\tilde{F}$ are respectively critical values for the leader and the $i$th follower’s objective, $\alpha_0, \beta_0, \alpha_i$, and $\beta_i$ are given confidence levels, and $i = 1, \ldots, m$.

In Model (14), the CCP approach is employed as well as critical values for the leader and the followers. This is because the objective functions and the constraint conditions are uncertain variables, and directly minimizing them makes no sense mathematically.

$$\begin{align*}
\min_x & \int_0^1 F(x, y_1^*, \ldots, y_m^*, \Phi_1^{-1}(\eta), \ldots, \\
& \hspace{1cm} \Phi_k^{-1}(\eta), \Phi^{-1}_{k+1}(1-\eta), \ldots, \Phi_n^{-1}(1-\eta)) d\eta \\
\text{s.t.} & \\
& \int_0^1 G(x, \Phi_1^{-1}(\eta), \ldots, \Phi_k^{-1}(\eta), \\
& \hspace{1cm} \Phi_{k+1}^{-1}(1-\eta), \ldots, \Phi^{-1}_n(1-\eta)) d\eta \leq 0 \\
& \{y_1^*, \ldots, y_m^*\} \text{ resoves the following subproblems} (i = 1, \ldots, m) \\
\min_{y_i} & \int_0^1 f_i(x, y_1^*, \ldots, y_m^*, \Phi_1^{-1}(\eta), \ldots, \\
& \hspace{1cm} \Phi_k^{-1}(\eta), \Phi^{-1}_{k+1}(1-\eta), \ldots, \Phi_n^{-1}(1-\eta)) d\eta \\
\text{s.t.} & \\
& \int_0^1 g_i(x, y_1^*, \ldots, y_m^*, \Phi_1^{-1}(\eta), \ldots, \Phi_k^{-1}(\eta), \\
& \hspace{1cm} \Phi_{k+1}^{-1}(1-\eta), \ldots, \Phi_n^{-1}(1-\eta)) d\eta \leq 0.
\end{align*}$$

(15)

**Definition 3.2.1** For a feasible decision $x$ of the upper level, the feasible array $(y_1^*, \ldots, y_m^*)$ is said to be a Nash equilibrium of the lower level if it satisfies

$$\begin{align*}
\min \{\tilde{F}|M\{f(x, y_1^*, \ldots, y_m^*, \xi_1, \ldots, \xi_n) \leq \tilde{F}\} \geq \alpha_i\} \\
\geq \min \{\tilde{F}|M\{f(x, y_1^*, \ldots, y_m^*, \xi_1, \ldots, \xi_n) \leq \tilde{F}\} \geq \alpha_i\}
\end{align*}$$

(16)

on the premise that $(y_1^*, \ldots, y_{i-1}^*, y_i, y_{i+1}^*, \ldots, y_m^*)$ is feasible for all $i = 1, \ldots, m$.

**Definition 3.2.2** The array $(x^*, y_1^*, \ldots, y_m^*)$ is said to be a Stackelberg-Nash equilibrium to Model (15) if there exists a feasible control vector $x^*$ of the upper level such that

$$\begin{align*}
\min\{\tilde{F}|M\{F(x, y_1^*, \ldots, y_m^*, \xi_1, \ldots, \xi_n) \\
\leq \tilde{F}\} \geq \alpha_i\} \\
\geq \min \{\tilde{F}|M\{F(x^*, y_1^*, \ldots, y_m^*, \xi_1, \ldots, \xi_n) \leq \tilde{F}\} \geq \alpha_i\}
\end{align*}$$

(17)

holds for any feasible decision $\bar{x}$, where $(\bar{y}_1, \ldots, \bar{y}_m)$ and $(\bar{y}_1^*, \ldots, \bar{y}_m^*)$ are the followers’ Nash equilibrium solutions and respectively dependent on $\bar{x}$ and $x^*$.

According to Theorem 2.1, Model (14) is able to be transformed to a deterministic equivalent model.

Assume that $F$ is a real continuous function, which monotonously increases as $\xi_1, \ldots, \xi_k$ increase and monotonously decreases as $\xi_{k+1}, \ldots, \xi_n$ decrease. In addition, $\xi_1, \ldots, \xi_n$ are independent uncertain variables and have uncertainty distributions $\Phi_1, \ldots, \Phi_n$, respectively. We have

$$\begin{align*}
M\{F(x, y_1^*, \ldots, y_m^*, \xi_1, \ldots, \xi_n) \leq \bar{F}\} \geq \alpha_0
\end{align*}$$

(18)

equals to

$$\begin{align*}
F(x, y_1^*, \ldots, y_m^*, \Phi^{-1}_1(\alpha_0), \ldots, \Phi^{-1}_n(\alpha_0), \\
\Phi^{-1}_{k+1}(1-\alpha_0), \ldots, \Phi^{-1}_n(1-\alpha_0)) \leq \bar{F}
\end{align*}$$

(19)

**Proof** Assume that $\Psi$ is the uncertainty distribution of $F(x, y_1^*, \ldots, y_m^*, \xi_1, \ldots, \xi_n)$. According to the definition of uncertainty distribution, we have Formula (18) equals to $\Psi(\bar{F}) \geq \alpha_0$.

Since $\Psi^{-1}(\Psi(\bar{F})) \geq \Psi^{-1}(\alpha_0)$ equals to $\Psi^{-1}(\alpha_0) \leq \bar{F}$, we have Formula (18) is equivalent to $\Psi^{-1}(\alpha_0) \leq \bar{F}$.

It follows from Theorem 2.1 that $\Psi^{-1}(\alpha_0)$ equals to

$$\begin{align*}
F(x, y_1^*, \ldots, y_m^*, \Phi^{-1}_1(\alpha_0), \ldots, \Phi^{-1}_k(\alpha_0), \\
\Phi^{-1}_{k+1}(1-\alpha_0), \ldots, \Phi^{-1}_n(1-\alpha_0))
\end{align*}$$

(20)

Hence, Formula (18) holds if and only if Formula (19) holds.

Assume that $f_i$ is a real continuous function, which monotonously increases as $\xi_1, \ldots, \xi_k$ increase and monotonously decreases as $\xi_{k+1}, \ldots, \xi_n$ decrease, and $\xi_1, \ldots, \xi_n$ are independent. Similar to the above proof, we have

$$\begin{align*}
M\{f_i(x, y_1^*, \ldots, y_m^*, \xi_1, \ldots, \xi_n) \leq \tilde{F}\} \geq \alpha_i
\end{align*}$$

(21)
where $i = 1, 2, \ldots, m$.

Let $G$ be a real continuous function, which monotonously increases as $\xi_1, \ldots, \xi_i$ increase and monotonously decreases as $\xi_{i+1}, \ldots, \xi_n$ decrease, and $\xi_1, \ldots, \xi_n$ are independent. Similarly, we have $M \{ G(x, \xi_1, \ldots, \xi_n) \leq 0 \} \geq \beta_0$ equals to

$$G(x, \Phi^{-1}_i(\beta_0), \ldots, \Phi^{-1}_n(1-\beta_0)) \leq 0. \quad (22)$$

Let $g_i$ be a real continuous function, which monotonously increases as $\xi_1, \ldots, \xi_i$ increase and monotonously decreases as $\xi_{i+1}, \ldots, \xi_n$ decrease, and $\xi_1, \ldots, \xi_n$ are independent. Then $M \{ g_i(x,y_1, \ldots, y_m, \xi_1, \ldots, \xi_n) \leq 0 \} \geq \beta_i$ equals to

$$g_i(x,y_1, \ldots, y_m, \Phi^{-1}_i(\beta_0), \ldots, \Phi^{-1}_n(1-\beta_0)) \leq 0, \quad (23)$$

where $i = 1, 2, \ldots, m$.

Thus Model (14) is equivalent to Model (28).

### 5 Intelligent algorithm

To calculate the Stackelberg-Nash equilibrium, we intend to present a genetic algorithm with the Nash equilibrium computation based on Liu (1998).

#### 5.1 Nash equilibrium computation

Define $y_{-i} = (y_1, y_2, \ldots, y_{i-1}, y_{i+1}, \ldots, y_m)$. After observing decisions of the other decision-makers in MP, the $i$th follower is able to make the optimal response $y_i = h_i(y_{-i})$ by solving the following problem

$$\begin{align*}
\min_{y_i} f_i(x, y_1, \ldots, y_m, \xi) \\
\text{s.t.} \\
g_i(x, y_1, \ldots, y_m, \xi) \leq 0.
\end{align*} \quad (24)$$

The formula $y_i = h_i(y_{-i})$ essentially turns the subproblem of the $i$th follower to a parametric optimization problem, where $x$ and $y_{-i}$ form the parameters. Then, the followers’ Nash equilibrium solution should solve the following subproblem for EVUMP

$$\begin{align*}
\min_{y_i} & \sum_{i=1}^{m} \| y_i - h_i(y_{-i}) \| \\
\text{s.t.} \\
E[g_i(x,y_1, \ldots, y_m, \xi)] & \leq 0, \quad i = 1, \ldots, m.
\end{align*} \quad (25)$$

or the following optimization model for CCUMP

$$\begin{align*}
\min_{y_i} & \sum_{i=1}^{m} \| y_i - h_i(y_{-i}) \| \\
\text{s.t.} \\
M \{ g(x,y_1, \ldots, y_m, \xi) \leq 0 \} & \geq \beta_i,
\end{align*} \quad (26)$$

We regard $(y^*_1, \ldots, y^*_m)$ as the followers’ Nash equilibrium if it satisfies

$$\sum_{i=1}^{m} \| y_i - h_i(y_{-i}) \| = 0. \quad (27)$$

Otherwise, we suggest that there is not a Nash equilibrium solution with respect to a given decision $x$.

Define $x_{-i} = (x_1, x_2, \ldots, x_{i-1}, x_{i+1}, \ldots, x_m)$. After observing decisions of the other decision-makers in MP, the $i$th leader is able to make the optimal response $x_i = f_i(x_{-i})$ by solving the following problem

$$\begin{align*}
\min_{x_i} f_i(x, x_1, \ldots, x_m, \xi) \\
\text{s.t.} \\
F(x, \xi) & \geq \bar{F},
\end{align*} \quad (28)$$

Combined with the Nash equilibrium computation, we give the following procedures of the genetic algorithm as well as a pseudo-code.

**Step 1:** Initialization. Generate a fixed amount of feasible chromosomes to form a population.

**Step 2:** Crossover. Randomly choose chromosomes to cross and keep feasible offsprings.

**Step 3:** Mutation. Randomly choose chromosomes for mutation and remove the infeasible ones.

**Step 4:** Fitness evaluation. For each chromosome, calculate the related Nash equilibrium to compute the leader’s objective which is the fitness.

**Step 5:** Selection. Run the tournament selection operator for a specified scale.
Step 6: Return to step 2 for a given number of iterations, and print out the best chromosome.

**Algorithm 1 Genetic algorithm**

**INPUTS**
1: Input basic coefficients
**INITIALIZATION**
2: Generate initial and feasible chromosomes
3: while generation < given number do
   **CROSSOVER**
   4: parents ← randomly choose chromosomes from population[generation]
   5: population[generation+1] = crossover(parents)
      (the crossover operator is designed according to specific problem)
   **MUTATION**
   6: mutants ← randomly choose chromosomes from population[generation+1]
   7: population[generation+1] ← population[generation +1] + Mutate(Mutants)
      (the mutation operator is designed according to specific problem)
   **FITNESS EVALUATION**
   8: for x in population[generation+1] do
      9:    Compute the corresponding Nash equilibrium (Subsection 4.1)
      10:   Compute the leader’s objective value F with penalty function method
      11:   fitness = F
   12:   end for
   **TOURNAMENT SELECTION**
   13: group ← randomly chose a fixed amount of chromosomes
   14: for each group do
      15:    Rank chromosomes according to their fitness
      16:    Find n/m winners of the group
      17:    end for
      18: Retain the elite
      19: population[generation+1] ← all obtained winners
      20: generation ← generation + 1
   21: end while
**OUTPUT**
22: Find best chromosome of the last population
23: Compute the Nash equilibrium (Section 4.1)
24: Calculate the leader and followers’ objective functions
25: Output decision vectors and objective values

6 Uncertain multilevel omni-channel vehicle routing problem

UMP will be introduced into the classic OVRP (Abdulkader et al. 2018) in this part, which is inspired by the following two factors. For one thing, an omni-channel approach seeks to provide a coherent cross-channel experience for consumers by coordinating the divided resources of multiple channels (Liu et al. 2020). For another, the multilevel approach has been studied in the transport field (Ning and Su 2017).

6.1 Problem description

In an OVRP, online consumers’ orders and offline replenishment orders are fulfilled by a same fleet simultaneously, which differs from classic VRPs. We consider an omni-channel distribution system that includes a warehouse, retail stores, consumers and homogeneous vehicles. Some assumptions are listed below:

- One vehicle is dispatched only once.
- Both the start and end points are the warehouse.
- The start time of all vehicles is 0.
- Each node is only visited once, except the warehouse.
- Consider various products with known inventory levels.
- Demands of all nodes are known.
- The vehicle capacity should not be exceeded, and consumer demands could be ignored at this time.
- Consumers are only supplied by retail stores, and retail stores are only supplied by the warehouse.
- Consumers are only visited after the retail store that supplies them.

In addition, the travel time of vehicles is considered to be an uncertain variable due to the changeful weather and traffic jams.

6.2 Mathematical models

For the online channel and omni-channel retail industries, one year can be simply divided into two periods: peacetime and shopping festivals such as Black Friday in North America. For these two different periods, we will construct the EVMP and CCMP models for the OVRP, respectively.

In MP, decision-makers have their own decisions and objectives. In this problem, we presume that the retailing company first determines the promising assignments and then the travelers optimize the routes of these assignments. In addition, we assume that the retailing company desires to minimize the total operational time for improving distribution efficiency and providing seamless services for consumers. The seamless services are highlighted by Abdulkader et al. (2018) in OVRP. For the lower level, the travelers seek to reduce the cost by minimizing the total travel distance.
Some notations are listed below before modelling:

- $O = \{0\}$: collection of the central warehouse;
- $R = \{1, \ldots, r\}$: collection of retail stores;
- $C = \{r + 1, \ldots, r + c\}$: collection of consumers;
- $N = \{0, 1, \ldots, r + c\}$: collection of all nodes;
- $K = \{1, \ldots, m\}$: collection of vehicles;
- $P = \{1, \ldots, e\}$: collection of product categories;
- $Q$: vehicle capacity;
- $\xi_{ij}$: uncertain travel time from $i$ to $j$, $i, j \in N$;
- $\Phi_{ij}$: uncertainty distribution of $\xi_{ij}$, $i, j \in N$;
- $d_{ij}^p$: demand of consumer $i$ for product $p$, $i \in C$, $p \in P$;
- $d_i$: demand of store $i$ provided by the warehouse, $i \in R$;
- $I_{ij}^p$: inventory level of product $p$ at store $i$, $i \in R$, $p \in P$;
- $D_{ij}$: travel distance of vehicles from $i$ to $j$, $i, j \in N$.

The leader’s decision vectors are listed as follows,

- $u_{ij}$: binary variable equals 1 if the product consumer $j$ ordered is provided by retail store $i$ and 0 otherwise, $i \in R$, $j \in C$.
- $v_{k,i}$: binary variable equals 1 if retail store $i$ is visited by vehicle $k$ and 0 otherwise, $i \in R$, $k \in K$.

For ease of demonstration, let $u$ and $v$ respectively denote the control vector containing every $u_{ij}$ and $v_{k,i}$ for $i \in R$, $j \in C$ and $k \in K$. Let $R_k$ and $C_i$ indicate the set of retail stores assigned to vehicle $k$ and the set of consumers assigned to retail store $i$, respectively. Then we have $R_k = \{r \cdot v_{k,1}, \ldots, r \cdot v_{k,R}\}$, and $C_i = \{(r + 1) \cdot u_{i(v_{k,R}+1)}, \ldots, (r + c) \cdot u_{i(v_{k,R}+c)}\}$. Let $N_k$ present the set of nodes visited by vehicle $k$. For any $k \in K$, we have $N_k = R_k \cup C_{v_{k,1}} \cup \cdots \cup C_{v_{k,R}}$.

Suppose that there are $n_k$ elements in $N_k$. The $k$th ($k \in K$) follower’s decision vector is listed as follows:

$x_k = (x_{k,1}^1, \ldots, x_{k,R}^{n_k})$: integer control vector indicating $n_k$ nodes of $N_k$ with $x_{k,i} \in N_k$ for all $i = 1, \ldots, n_k$ and $x_{k,i} \neq x_{k,j}$ for all $i \neq j$, $i, j = 1, \ldots, n_k$. Namely, $\{x_{k,1}^1, \ldots, x_{k,R}^{n_k}\}$ is a rearrangement of all elements in $N_k$.

Let $\tau_{ij}(x_k, \xi)$ present the departure time of vehicle $k$ at node $x_{k,i}$, where $x_{k,i} \in N_k$. For any vehicle $k$ with $k \in K$, if there is any non-zero element in the $N_k$, then we have $\tau_{ij}(x_k, \xi) = \xi_{m_k+1}$, and $\tau_{ij}(x_k, \xi) = \tau_{ij-1}(x_k, \xi) + \xi_{m_k+1}$ for $2 \leq j \leq n_k$.

### 6.2.1 EVUMP models for OVRP

In peacetime, the number of orders should not be particularly large, and it is unnecessary to worry about delayed deliveries. In this case, decision-makers are risk-neutral, so we have the EVUMP model as follows:

$$\begin{align*}
\min \quad & F = E \left[ \sum_{k=1}^{m} \left( \tau_{ij}(x_k, \xi') + \xi_{m_k+1}' \right) \right] \\
\text{s.t.} \quad & \sum_{i \in R_k} d_i \leq Q, k \in K \\
& \sum_{j \in C} u_{ij}d_{ij}^p \leq I_{ij}^p, i \in R, p \in P \\
& \sum_{i \in R_k} u_{ij} = 1, j \in C \\
& \sum_{k \in K} v_{k,i} = 1, i \in R \\
& u_{ij}, v_{k,i} \in \{0, 1\}, i \in R, j \in C, k \in K \\
\end{align*}$$

$(29)$

### 6.2.2 CCUMP models for OVRP

A large number of orders are produced during the shopping festivals. In these periods, decision-makers are risk-averse and desire to avoid express delivery backlogs. Consequently, we launch the following CCUMP model
\[
\begin{align*}
&\min \bar{F} \\
&s.t.
\begin{cases}
M \left\{ \sum_{k=1}^{m} (r_{x_k}^v (x_k, \xi) + \xi_{x_k}^v) \leq \bar{F} \right\} \geq \alpha_0 \\
\sum d_i \leq Q, k \in K \\
\sum_{j \in C} u_{i,j} d_{j}^p \leq I_{i}^p, i \in R, p \in P \\
\sum_{i \in R_k} v_{k,i} = 1, j \in C \\
\sum_{k \in K} u_{i,j} v_{k,i} = \{0, 1\}, i \in R, j \in C, k \in K \\
(x_1^v, \cdots, x_m^v) \text{ resolves the following subproblems} (k = 1, \cdots, m) \\
&\min f_k(x_k) = D_{0_{x_k}} + \sum_{j=1}^{n_k-1} D_{x_k,j} + D_{x_k,0} \\
&s.t.
\begin{cases}
M \left\{ u_{i,j} r_{j} (x_k, \xi) \leq r_j (x_k, \xi) \right\} \geq \beta_i, i,j \in N \\
x_k^i \neq x_k^j, i \neq j, i,j = 1, \cdots, n_k, x_k^i, x_k^j \in N_k,
\end{cases}
\end{cases}
\end{align*}
\]
where \( R_k = \{1 \cdot v_{k,1}, \cdots, r \cdot v_{k,r}\} \),
\( C_i = \{(r + 1) \cdot u_{i,v_{k,r+1}}, \cdots, (r + c) \cdot u_{i,v_{k,r+c}}\} \),
\( N_k = R_k \cup C_1 \cup \cdots \cup C_r \),
\( \alpha_0, \beta_i \) are known confidence levels, and \( i \in N \).

### 6.3 Numerical experiment

This section suggests a numerical example for UMOVRP. Assume there are 1 warehouse, 4 retailing stores and 6 consumers. Three vehicles form the fleet and the vehicle capacity is 100. Three types of products can be ordered online by consumers. Assume that the travel time from node \( i \) to node \( j \) is a linear uncertain variable \( L(0.1|i - j| - 1, 0.1|i - j| + 1), i \neq j, i,j = 0, \cdots, 10 \). And we also assume that \( \Psi \) is the uncertainty distribution of the uncertain departure time \( r \). More data about the distribution network are listed in Table 1, where the notation “X” and “Y” jointly denote the two-dimensional coordinates of nodes and other notations are already declared in Sect. 5.1.

**Table 1** Data of the OVRP

| \( i \) | X | Y | \( d_i \) | \( d_1^i \) | \( d_2^i \) | \( d_3^i \) | \( l_1 \) | \( l_2 \) | \( l_3 \) |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 97 | 4  | | | | | | | |
| 1 | 2  | 0  | 37 | 2  | 3  | 1  | | | |
| 2 | 98 | 17 | 34 | 1  | 2  | 1  | | | |
| 3 | 3  | 25 | 37 | 2  | 3  | 1  | | | |
| 4 | 84 | 66 | 30 | 2  | 3  | 1  | | | |
| 5 | 68 | 53 | 0  | 1  | 0  | | | | |
| 6 | 56 | 84 | 0  | 0  | 1  | | | | |
| 7 | 15 | 16 | 1  | 0  | 0  | | | | |
| 8 | 3  | 41 | 0  | 1  | 0  | | | | |
| 9 | 83 | 80 | 1  | 0  | 0  | | | | |
| 10 | 25 | 89 | 0  | 1  | 0  | | | | |
Algorithm 1 will be used in this subsection. In the algorithm, the crossover operator is the single point crossover for \((u, v)\) but the partially matched crossover for \(x_k (k \in K)\). The mutation operator is the simple inversion mutation for each control vector of both levels. And the selection operator is the tournament selection with an elitism strategy for both levels. For more details, interested readers may refer to Katoch et al. (2021).

### 6.3.1 Expected value model

In the peacetime, Model (29) for this example can be converted to Model (31) according to Sect. 3.1.

Run the intelligent algorithm and then we have the following Stackelberg-Nash equilibrium

\[
\begin{aligned}
& u_{1.5} = u_{1.8} = u_{3.7} = u_{2.6} = u_{4.9} = u_{1.10} = 1, \\
& v_{1.1} = v_{1.3} = v_{3.2} = v_{3.4} = 1, \\
& x^* = ((1, 3, 7, 8.5), (2, 4, 9, 6, 10)), \\
& F^* = 4.2, \quad f_1^* = 258.9, \quad f_2^* = 0.0, \quad f_3^* = 248.1.
\end{aligned}
\]

The optimal route plan (see Fig. 2a) is

- vehicle 1: 0 → 1 → 3 → 7 → 8 → 5 → 0,
- vehicle 2: not used,
- vehicle 3: 0 → 2 → 4 → 9 → 6 → 10 → 0,

and the formulation of assignments is that store 1 serves consumers 5, 6, 9, and store 3 is not required to provide products.

### 6.3.2 Chance-constrained model

During shopping festivals, we assume that confidence levels \(a_0, \beta_i\) all equal to 0.90 for \(i = 1, \ldots, 10\). For this instance, we have Model (32) which is the crisp equivalent of Model (30) according to Sect. 3.2.

The Stackelberg-Nash equilibrium is obtained by the intelligent algorithm

\[
\begin{aligned}
& u_{1.7} = u_{2.8} = u_{2.10} = u_{4.5} = u_{4.6} = u_{4.9} = 1, \\
& v_{1.2} = v_{1.4} = v_{2.1} = v_{2.3} = 1, \\
& x^* = ((2, 4, 5, 8, 10, 6, 9), (1, 3, 7, 1)), \\
& F^* = 4.0, \quad f_1^* = 339.5, \quad f_2^* = 218.0, \quad f_3^* = 0.0.
\end{aligned}
\]

The optimal route plan (see Fig. 2b) is

- vehicle 1: 0 → 2 → 4 → 5 → 8 → 10 → 6 → 9 → 0,
- vehicle 2: 0 → 1 → 3 → 7 → 0,
- vehicle 3: not used,

and the formulation of assignments is that store 1 serves consumer 7, store 2 serves consumers 8, 10, store 4 serves consumers 5, 6, 9, and store 3 is not required to provide products.

### Table 2 Comparison of results

|          | Classic OVRP | EV model | CC model |
|----------|--------------|----------|----------|
| Total cost | 517.8        | 534.0    | 557.5    |
| Total time | 5            | 4.2      | 4.0      |
\[ C_i = \{ 5 \cdot u_{v_i,5}, \ldots, 10 \cdot u_{v_i,10} \}, \]

and \[ N_k = R_k \cup C_{1:v_i} \cup \cdots \cup C_{4:v_i}. \]

Like (Xue et al. 2020), we compare the results of the above two models with the classic OVRP (Abdulkader et al. 2018) before using the UMP approach. The result of the classic OVRP is obtained by Algorithm 1 without Nash equilibrium computation.

According to Table 2, the travel cost of the EVUMP model increased by 3.1\% compared to the classic OVRP, while the total operational time decreased by 16\%. Although disappointing about the increase in cost, this result shows that EVUMP helps to coordinate the interests of the upper and lower levels in peacetime. For the CCUMP model, its travel cost increased by 7.7\% compared to the classic OVRP, while the total operational time decreased by 20\%. This result is reasonable because retailing companies are willing to avoid delayed deliveries and the loss of consumers by sacrificing a part of profits for higher efficiency during shopping festivals.

7 Conclusion

The contributions of this work are as follows. For one thing, this paper built EVUMP and CCUMP models as well as their equivalent crisp models. For another, this paper applied the presented UMP models to the OVRP. Then, a numerical example is given. The results of the numerical experiment are obtained by an intelligent algorithm. The results show that the proposed models are able to optimize the distribution efficiency by coordinating the interests of the leader and followers.

For future research, we have the following directions. Firstly, it is worthwhile to consider the multi-objective case in UMP. Thereafter, another research direction is to apply UMP to many more fields, such as the attacker-defenders scenario. Furthermore, we will continue to design more effective algorithms.

Acknowledgements This research was supported by National Natural Science Foundation of China (Grant No. 72101080), Natural Science Foundation of Hebei Province (Grant No. F2020202056), and the Key Project of Hebei Education Department (Grant No. ZD2020125).

Declarations

Conflicts of Interest The authors declare that there is no conflict of interest.

References

Abdulkader MMS, Gajpal Y, Elmekkawy TY (2018) Vehicle routing problem in omni-channel retailing distribution systems. Int J Prod Econ 196:43–55. https://doi.org/10.1016/j.ijpe.2017.11.011

Abo-Elnaga Y, Nasr S (2022) K-means cluster interactive algorithm-based evolutionary approach for solving bilevel multi-objective programming problems. Alexandria Eng J 61(1):811–827. https://doi.org/10.1016/j.aej.2021.04.098

Ahmadzade H, Gao R (2018) Reversed hazard function of uncertain lifetime. Fuzzy Opt Decision Making 17(4):387–400. https://doi.org/10.1007/s10700-017-9281-0

Barma PS, Dutta J, Mukherjee A et al (2021) A hybrid GA-BFO algorithm for the profit-maximizing capacitated vehicle routing problem under uncertain paradigm. J Intell Fuzzy Syst 40(5):8709–8725. https://doi.org/10.3233/JIFS-192134

Barma PS, Dutta J, Mukherjee A et al (2022) A multi-objective ring star vehicle routing problem for perishable items. J Ambient Intell Human Comput 13:2355–2380. https://doi.org/10.1007/s12652-021-03059-2

Ben-Ayed O, Blair CE (1990) Computational difficulties of bilevel linear programming. Oper Res 38(3):556–560. https://doi.org/10.1287/opre.38.3.556

Bracken J, McGill JT (1973) Mathematical programs with optimization problems in the constraints. Oper Res 21(1):37–44. https://doi.org/10.1287/opre.21.1.37

Dethlefs C, Ostermeier M Hübner A (2022) Rapid fulfillment of online orders in omnichannel grocery retailing. EURO Journal on Transportation and Logistics, Vol. 11, Article 100082. https://doi.org/10.1007/s12545-022-100082

Dutta J, Barma PS, Mukherjee A et al (2022) A hybrid multi-objective evolutionary algorithm for open vehicle routing problem through cluster primary-route secondary approach. Int J Manage Sci Eng Manag 17(2):132–146. https://doi.org/10.1080/17509653.2021.2000901

Dutta J, Barma PS, Mukherjee A et al (2022) Multi-objective green mixed vehicle routing problem under rough environment. Transport 37(1):51–63. https://doi.org/10.3846/transpor.2021.14464

Fan H, Cheng HZ (2014) Transmission network expansion planning based on dependent-chance bi-level programming method. J Int Council Electr Eng 4(1):11–15. https://doi.org/10.5370/JICEE.2014.4.1.011

Fathollahi-Fard AM, Hajiaghaei-Keshhtel M, Tavakkoli-Moghaddam R et al (2022) Bi-level programming for home health care supply chain considering outsourcing. J Ind Inf Integr 25:100246. https://doi.org/10.1016/j.jii.2021.100246

Gao JW (2004). Models and algorithms of uncertain multilevel programming. Dissertation. Tsinghua University

Gao JW, Liu B, Gen M (2004) A hybrid intelligent algorithm for stochastic multilevel programming. IEEE Trans Electron Inform Syst 124(10):1991–1998. https://doi.org/10.1016/j.ieeeiiss.2014.4.1.011

Gao R, Ralescu DA (2019) Uncertain wave equation for vibrating string. IEEE Trans Fuzzy Syst 27(7):1323–1331. https://doi.org/10.1109/TFUZZ.2018.2877628

Guo C, Thompson RG, Foliente G et al (2021) An auction-enabled collaborative routing mechanism for omnichannel on-demand logistics through transshipment. Transp. Res. Part E: Logistics and Transp Rev 146:102206. https://doi.org/10.1016/j.tre.2020.102206

Guo J, Xue YQ Guan HZ (2022) Research on the combinatorial optimization of EBs departure interval and vehicle configuration based on uncertain bi-level programming. Transportation Letters. https://doi.org/10.1080/19427867.2022.2077583

Haywood AB, Lunday BJ Robbins MJ (2022) Intruder detection and interdiction modeling: A bilevel programming approach for ballistic missile defense asset location. Omega, Vol 110, Article 102640. https://doi.org/10.1016/j.omega.2022.102640

Hosseini A, Wadbro E (2022) A hybrid greedy randomized heuristic for designing uncertain transport network layout. Expert systems with
Uncertain multilevel programming with application to omni-channel vehicle routing problem

9171

Patriksson M, Wynter L (1999) Stochastic mathematical programs with equilibrium constraints. Oper Res Lett 25(4):159–167. https://doi.org/10.1016/S0167-6377(99)00052-8

Roghani E, Sadjadi SJ, Aryanezhad MB (2007) A probabilistic bi-level linear multi-objective programming problem to supply chain planning. Appl Math Comput 188(1):786–800. https://doi.org/10.1016/j.amc.2006.10.032

Rahmani A, Hosseini M (2021) A competitive stochastic bi-level inventory location problem. Int J Manag Sci Eng Manag 16(3):209–220. https://doi.org/10.1080/17599653.2021.1930602

Ramamoorthy R, Thangavelu M (2022) An enhanced bio-inspired routing algorithm for vehicular ad hoc networks, Trends in Sciences, 19(10), Article 4188. https://doi.org/10.48048/tis.2022.4188

Ramamoorthy R, Thangavelu M (2022) An enhanced distance and residual energy-based congestion aware ant colony optimization routing for vehicular ad hoc networks, International Journal of Communication Systems, 35(11), Article 5179. https://doi.org/10.1002/dac.5179

Ramamoorthy R, Thangavelu M (2022) An enhanced hybrid ant colony optimization routing protocol for vehicular ad-hoc networks. J Ambient Intell Human Comput 13(8):3837–3868. https://doi.org/10.1007/s12652-021-03176-y

Schubert D, Kuhn H, Holzapfel A (2021) Same-day deliveries in omnichannel retail: integrated order picking and vehicle routing with vehicle-site dependencies. Naval Res Logistics 68(6):721–744. https://doi.org/10.1002/nav.21954

Su S, Li Z, Jin XL, et al (2022) Bi-level energy management and pricing for community energy retailer incorporating smart buildings based on chance-constrained programming. International Journal of Electrical Power and Energy Systems, Vol. 138, Article 107894. https://doi.org/10.1016/j.ijepes.2021.107894

Wang K, Zhao M, Zhou J et al (2018) The operational law of uncertain variables with continuous uncertainty distributions. IEEE Trans Fuzzy Syst 26(5):2926–2937. https://doi.org/10.1109/TFUZZ.2018.2807378

Wei JD, Zhang Y, Wang JX et al (2021) Distribution Imp-based demand management in industrial park via a bi-level programming approach. IEEE Trans Sustain Energy 12(3):1695–1706. https://doi.org/10.1109/TSTE.2021.3062044

Xue YQ, Guo J, An J, et al (2020) Uncertain Bi-level Programming Model for Vehicle Allocation Problem of Bus Lines. Journal of Transportation Systems Engineering and Information Technology, 20(2), 145-150. https://doi.org/10.16097/j.cnki.1009-6744.2020.02.022

Yao K, Zhou J (2018) Renewal reward process with uncertain inter-arrival times and random rewards. IEEE Trans Fuzzy Syst 26(3):1757–1762. https://doi.org/10.1109/TFUZZ.2017.2718965

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.