International Journal of Applied Operational Research
Vol. 2, No. 1, pp. 65-83, Spring 2012

Journal homepage: www.ijorlu.ir

A Vehicle Routing Problem with Inventory in a Hybrid Uncertain Environment

A. Malekly*

Received: October 10, 2011 ; **Accepted:** March 8, 2012

Abstract Manufacturers, who re-supply a large number of customers, continually struggle with the question of how to formulate a replenishment strategy. The purpose of this paper is to determine the optimal set of routes for a group of vehicles in the transportation network under defined constraints – which is known as the Vehicle Routing Problem (VRP) – delivering new items, and resolving the inventory control decision problem simultaneously since the regular VRP does not. Both the vehicle routing decision for delivery and the inventory control decision affect each other and must be considered together. Hence, a mathematical model of vehicle routing problem with inventory is proposed whose demands are assumed to be hybrid variables (HVRPI) in which fuzziness and randomness are considered together. Then, the problem is transformed into its equivalent deterministic form and presented as a multi-objective mixed integer nonlinear programming. Since finding the optimal solution(s) for HVRPI is a *NP*-hard, a solution algorithm is presented composed of the constrained Nelder–Mead method and a Tabu search algorithm for the vehicle routing to solve the complex problem. The usefulness of the model is validated by experimental results. The findings indicate that the proposed model can provide a practical tool to significantly reduce the logistic cost.

Keywords Vehicle Routing Problem, Hybrid Variable, Nelder–Mead Method, Tabu Search.

1 Introduction

A key issue in logistics is the cost-efficient management of a heterogeneous vehicle fleet providing delivery service to a given set of customers with known demands. The collection/distribution system manager should not only decide on the number and types of vehicles to be used, but also he/she must specify which customers are serviced by which vehicle and what sequence to follow so as to minimize the transportation cost. Products to be delivered are loaded at the depot. Then, every vehicle route must start and finish at the assigned terminal, and both vehicle capacity and demands are to be satisfied. Moreover, each customer must be serviced by exactly one vehicle since split demand is not allowed. This class of logistic problems is usually known as the vehicle routing problem (VRP), and its objective is usually the minimization of the overall distance traveled by the vehicles while servicing all the customers. The interest in VRP problems comes from its practical relevance as well as from the considerable difficulty to solve them precisely. In the field of combinatorial optimization, the VRP is regarded as one of the most challenging problems. It

* **Corresponding Author.** (✉)
E-mail: a_malekly@iau-tnb.ac.ir (A. Malekly)

A. Malekly
Department of Industrial Engineering, North Tehran Branch, Islamic Azad University, Tehran, Iran.

[Downloaded from ijaor.com on 2024-10-12]

is indeed *NP*-hard, so that the task of finding the best set of vehicle tours by solving optimization models is computationally prohibitive for real-world applications. This problem was first introduced by Dantzig and Ramser [1], and was developed by Clarke and Wright [2]. In the past, the resolution of the vehicle routing problem was based on the minimal transportation cost criterion without considering the inventory cost. However, when only one factor is accessed and minimized in the logistic system, the costs of other factors are increased [3]. For example, if the vehicle routing decision problem is considered, and the inventory control decision problem is neglected, the vehicle routing decision for delivery can be effectively made, and however, the inventory control decision cannot, causing manufacturers not be able to reduce the total logistic cost effectively. In contrast, if only the inventory control decision problem is considered and the vehicle routing decision problem is ignored, the transportation cost would increase since the vehicle routing decision cannot be effectively made. Hence, the total logistic cost (transportation cost and inventory cost) would increase. So the vehicle routing problem and the inventory control decision problem need to be considered simultaneously so that the total logistic cost can be minimized.

There are uncertain factors in VRP, such as demands of consumers, travel times between consumers, number of vehicles, and consumers to be visited [4]. Stochastic vehicle routing problems (SVRP) arise whenever some elements of the problem are random. Common examples are stochastic demands and stochastic travel times. Sometimes, the set of customers to be visited is not known with certainty. In such cases, each customer has a probability of being presented. To the best of our knowledge, Tillman [5] was the first to propose an algorithm for the SVRP in which there were several depots, and the algorithm proposed by Tillman was based on Clarke and Wright [2]. A second major article is due to Stewart and Golden [6]. It contained extensions and generalizations of previous results of Golden and Yee [7]. A chance constrained programming model (CCP) and two expected value models (EVM) were presented. After that, many researchers, such as Bodin et al. [8], Warters [4] studied various types of SVRP.

But, in some systems, it is hard to describe the parameters of the problem as random variables because there are not enough data to analyze. For instance, in one problem, customers' demands are often not precise enough, especially for the urban traffic. Generally, it can use fuzzy variables to deal with these uncertain parameters, which are first presented by Teodorović and Pavkovićkm [9] in VRP, and Lai et al. [10] modeled VRP by fuzzy programming with possibility measure. In a recent paper proposed by Zheng and Liu [11], a vehicle routing problem with time windows was solved by the concept of CCP, in which demands were regarded as fuzzy variables. On the other hand, fuzzy set theory now has also made an entry into the inventory control systems. Sommer [12] applied the fuzzy concept to an inventory and production-scheduling problem. Park [13] examined the EOQ formula in the fuzzy set theoretic perspective associating the fuzziness with the cost data. Das et al. [14], Roy and Maiti [15] solved a single objective fuzzy EOQ model using Geometric Programming technique. De and Goswami [16] derived a replenishment policy for items with finite production rate and fuzzy deterioration rate represented by a triangular fuzzy number using extension principle. However, a blend of several types of information is encountered in realistic models to furnish an excellent depiction of the phenomenon which leads to the concept of hybridization. This means that randomness and impreciseness can be combined simultaneously to represent the real world as it is perceived. Such combinations may be represented by hybrid numbers, random fuzzy numbers, fuzzy random numbers, expectation of fuzzy sets, possibility of random variables, and several others. These novel concepts will

[Downloaded from ijaor.com on 2024-10-12]

be described in view of applications in human sciences but such tools can be also used in every scientific research such as operations research, etc. Again a parameter may have different fuzzy values in nature with some non-fuzzy probabilities. These parameters are called fuzzy random parameters. For example, a company may have different securities, share, etc. and by selling these they may raise their capital for budget investment to buy new products which are to be supplied. Since the share market is probabilistic, the amount of money extracted from the market is random. The amount may be ‘‘around \$10 million with probability 0.3’’, ‘‘about \$15 million with probability 0.5’’, etc. By fuzzy random programming we mean the optimization theory in fuzzy random environments. In the case of VRP, He and Xu [17] proposed a class of random fuzzy VRP, and Malekly et al. [18] presented a new fuzzy random model (FRCVRP) for dairy industry. On the other side, there are also rarely-discussed papers in the case of fuzzy random inventory problem [19, 20]. To the best of our knowledge, in VRP with inventory no attempt has been made where fuzziness and randomness coexist. Therefore, there is a strong motivation for further research in the area. So, in this paper the aim is to provide a new formulation for a capacitated VRP with inventory and hybrid demands (HVRPI) which is a common problem in practice.

2 Fuzzy number

The theory of fuzzy sets introduced by Zadeh [21] was developed to describe vagueness and ambiguity in the real world systems. Zadeh defined a fuzzy set $\tilde{a}$ in a universe of discourse X as a class of objects with a continuum of grades of memberships. Such a set is characterized by a membership function $\mu_{\tilde{a}}(x)$ which associates with each point x in X a real number in the interval [0,1]. $\mu_{\tilde{a}}(x)$ represents the grade of membership of x in $\tilde{a}$.

A fuzzy set $\tilde{a}$ in the universe of discourse $\Re$ (set of real numbers) is called a fuzzy number if it satisfies the following conditions:

(*i*) $\tilde{a}$ is normal i.e. there exists at least one $x \in \Re$ such that $\mu_{\tilde{a}}(x) = 1$.

(*ii*) $\tilde{a}$ is convex.

(*iii*) the membership function $\mu_{\tilde{a}}(x)$, $x \in \Re$ is at least piecewise continuous.

2.1 Triangular fuzzy number

Triangular fuzzy number (*TFN*) ($\tilde{a}$) is the fuzzy number with the membership function $\mu_{\tilde{a}}(x)$, a continuous mapping: $\mu_{\tilde{a}}(x)$: $\Re \rightarrow [0,1]$, where

$$\mu_{\tilde{a}}(x) = \begin{cases} 0 & -\infty < x < a_1 \\ \dfrac{x - a_1}{a_2 - a_1} & a_1 \le x < a_2 \\ \dfrac{a_3 - x}{a_3 - a_2} & a_2 \le x \le a_3 \\ 0 & a_3 < x < \infty. \end{cases} \tag{1}$$

[Downloaded from ijaor.com on 2024-10-12]

2.2 η-cut of a fuzzy number

A η-cut of a fuzzy number $\tilde{a}$ is defined as a crisp set
$a_\eta=\{x: \mu_{\tilde{a}}(x) \geq \eta, x\in\Re\}$ where $\eta \in[0,1]$.

2.3 Approximate value of triangular fuzzy number (TFN)

According to Kaufmann and Gupta [22], the approximated value of TFN $\tilde{a} \equiv (a_1,a_2,a_3)$ is given by $\hat{a} = ¼ (a_1 + 2a_2 + a_3)$.

2.4 Algebraic operation of fuzzy numbers

Addition:
Let $\tilde{a} \equiv (a_1,a_2,a_3)$ and $b \equiv (b_1, b_2, b_3)$ be two triangular fuzzy numbers. Using max-min convolution on fuzzy numbers $\tilde{a}$ and b the membership function of the resulting fuzzy number $\tilde{a}$ (+) b can be obtained as $\vee_{z=x+y} (\mu_{\tilde{a}}(x) \wedge \mu_{\tilde{b}}(y))$, $\forall x, y, z \in\Re$ where the symbols '$\wedge$' and '$\vee$' are used for minimum and maximum, respectively. In short we can write $\tilde{a}$ (+) $b = (a_1,a_2,a_3)$ (+) (b_1,b_2,b_3).

Scalar multiplication: For any real constant t,

$$t\tilde{a} = \begin{cases} (ta_1, ta_2, ta_3) & t \geq 0 \\ (ta_3, ta_2, ta_1) & t < 0. \end{cases}$$

2.5 Fuzzy possibility techniques

Let $\tilde{a}$ and b be two fuzzy quantities with membership functions $\mu_{\tilde{a}}(x)$ and $\mu_b(y)$, respectively. Then according to Dubois and Prade [23], Liu and Iwamura [24, 25].

$$pos(\tilde{a}*b) = \sup\{\min(\mu_{\tilde{a}}(x), \mu_b(y)): x,y \in\Re, x*y\}$$

where the abbreviation '*pos*' represents possibility and $*$ is any of the relations $<, >, =, \leq, \geq$.

If $\tilde{a}$ and b are two fuzzy numbers defined on $\Re$ and $\tilde{u} = f(\tilde{a}, b)$ where f: $\Re\times\Re\rightarrow\Re$ is a binary operation then the membership function $\mu_{\tilde{u}}$ of $\tilde{u}$ is defined as $\mu_{\tilde{u}}(u) = \sup\{\min(\mu_{\tilde{a}}(x), \mu_b(y)): x,y \in\Re$ and $u = f(x,y), \forall u \in\Re\}$.

3 Random variable

Let $\underline{L}$ $(=(m,\sigma^2))$ be a continuous random variable with probability density function (pdf) $f_{\underline{L}}(l)$ whose mean and variance are m and σ^2, respectively. Similarly, let $\underline{L}'$ $(=(m', \sigma'^2))$ be another

[Downloaded from ijaor.com on 2024-10-12]

random variable with pdf $f_{\underline{L}'}(l')$. If $\underline{L}$ and $\underline{L}'$ are two independent random variables, then we have the following algebraic operations:
Addition:

$$\underline{L}_1[+]\underline{L}_2 = (m_1, \sigma_1^2)[+](m_2, \sigma_2^2) = (m_1 + m_2, \sigma_1^2 + \sigma_2^2).$$

here, according to sum-product convolution $\underline{\boldsymbol{L}}$ (=$\underline{L}$+$\underline{L}'$) is a random variable with the same type of pdf $f_{\underline{L}}(\boldsymbol{l}) = (\int_{\Re} f(\boldsymbol{l}\text{-}l')f'(l')dl'$ with mean $\boldsymbol{m}$ (= m+m') and variance $\boldsymbol{\sigma^2}$ (= σ^2+σ'^2).

Scalar multiplication:
$t\underline{L} = (tm, t^2\sigma^2)$. Here tL and L have the same type of pdf.

4 Hybrid number [22]

Assume $\underline{\tilde{A}}$(=($\tilde{A}$, $\underline{L}$)) is a hybrid number. Here the couple ($\tilde{A}$,$\underline{L}$) represents the addition to a fuzzy number with a random variable without altering the characteristic of each one and without decreasing the amount of available information where $\tilde{A}$ is a fuzzy number and $\underline{L}$ is the random variable with density function $f_{\underline{L}}(l)$. Let $\underline{\tilde{A}}$(=($\tilde{A}$, $\underline{L}$)) and $\underline{\tilde{A}}'$(=($\tilde{A}'$, $\underline{L}'$)) be two hybrid numbers in $\Re$ where $f_{\underline{L}}(l)$ and $f_{\underline{L}'}(l')$ are the pdfs of L and L', respectively. So a hybrid convolution for addition will be defined as ($\tilde{A}$, $\underline{L}$) $\oplus$ ($\tilde{A}'$, $\underline{L}'$) = ($\tilde{A}$(+) $\tilde{A}'$, $\underline{L}$ [+]$\underline{L}'$) = **($\tilde{A}$, $\underline{L}$)**, where (+) represents the max-min convolution for addition of fuzzy subsets and [+] represents the sum-product convolution for addition of random variables. We denote the couple **($\tilde{A}$, $\underline{L}$)** by the symbol $\boldsymbol{\tilde{A}(+)'\underline{L}}$.
So, $\mu_{\tilde{A}_1(+)\tilde{A}_2}(z) = \vee_{z=x+y}(\mu_{\tilde{A}_1}(x) \wedge \mu_{\tilde{A}_2}(y)), \forall x, y, z \in \Re$ and $f(l) = \int_{\Re} f_1(l-l_2)f_2(l_2)dl_2$ or $\int_{\Re} f_1(l_1)f_2(l-l_1)dl_1$.

Note 1. A fuzzy number is a special case of a hybrid number if $\underline{\tilde{A}}$ = ($\tilde{A}$,$\underline{0}$), where $\underline{0}$ is the trivial random variable with the following probabilities:

$$\begin{aligned} P(l) &= 1, \quad l = 0 \\ &= 0, \quad l \neq 0. \end{aligned}$$

Note 2. A random variable is also a special case of a hybrid number if $\underline{L}$ = (0, $\underline{L}$), where 0 is the trivial fuzzy number with membership function

$$\begin{aligned} \mu_{\tilde{0}}(x) &= 1, \quad x = 0 \\ &= 0, \quad x \neq 0. \end{aligned}$$

Note 3. $\underline{0}$ = (0,0) is the neutral for addition of hybrid numbers.
If $\tilde{u}_1$ is a fuzzy cost, $\underline{u}_2$ is a random cost and u_3 is a fixed cost then the total cost can be expressed as

$$\tilde{u}_1[+]\underline{u}_2[+]u_3 = (\tilde{u}_1, 0)[+](0, \underline{u}_2)[+](0, u_3) = (\tilde{u}_1, \underline{u}_2(+)'u_3) = (\tilde{u}_1(+)u_3, \underline{u}_2). \tag{2}$$

we can consider the fixed number like a sum of two parts $u_3 = u_3' + u_3''$ and write for (2)

[Downloaded from ijaor.com on 2024-10-12]

$$\widetilde{u}_1[+]\underline{u}_2[+]u_3 = (\widetilde{u}_1[+]u_3', \underline{u}_2[+]u_3''). \tag{3}$$

The mathematical expectation of a hybrid number is defined as follows.
A function $\phi(x)$ in $\Re$ that is nonnegative and monotonically increasing is:

$$\begin{aligned} &\forall x_1, x_2 \in \Re: \\ &(x_1 > x_2) \Rightarrow (\phi(x_2) \geq \phi(x_1)). \end{aligned} \tag{4}$$

for a closed interval of $\Re$, $[a_\alpha^1, a_\alpha^2]$ we have:

$$[\phi(a_\alpha^1), \phi(a_\alpha^2)] \subset \Re \tag{5}$$

and for $l \in \Re$:

$$[\phi(a_\alpha^1 + l), \phi(a_\alpha^2 + l)] \subset \Re. \tag{6}$$

if l is the value of the random variable L, the lower and upper bounds of (6) depend only on l for a given level α. The mathematical expectation for each bound is now computed:

$$E[\phi(a_\alpha^1 + l), \phi(a_\alpha^2 + l)] = \left[\int_{l_1}^{l_2} \phi(a_\alpha^1 + l).f(l)dl, \int_{l_1}^{l_2} \phi(a_\alpha^2 + l).f(l)dl\right] \tag{7}$$

Theorem The membership function of the mathematical expectation of a hybrid number ($\tilde{A}$, $\underline{L}$) is the membership of $\tilde{A}$ shifted by the mathematical expectation of $\underline{L}$ [22].

Proof. Using the intervals of confidence of level α:

$$\begin{aligned} E_\alpha(\widetilde{A}[+]\underline{L}) &= \left[\int_{l_1}^{l_2} (a_\alpha^1 + l).f(l)dl, \int_{l_1}^{l_2} (a_\alpha^2 + l).f(l)dl\right] \\ &= \left[a_\alpha^1.\int_{l_1}^{l_2} f(l)dl + \int_{l_1}^{l_2} l.f(l)dl, a_\alpha^2.\int_{l_1}^{l_2} f(l)dl + \int_{l_1}^{l_2} l.f(l)dl\right] \\ &= \left[a_\alpha^1 + E(l), a_\alpha^2 + E(l)\right] \end{aligned} \tag{8}$$

Hence, in a hybrid sum, if the random variables satisfy their random expectation, they will have the same effect as ordinary numbers, shifting the sum of fuzzy numbers.
Using the notation ($\tilde{A}$,$\underline{L}$) = $\tilde{A}(+)'\underline{L}$, where $\tilde{A}$ is a triangular fuzzy number, the following example is illustrated.

Example Let $\widetilde{A}_1 = (3,5,9)(+)'(6,1.2)$ and $\widetilde{A}_2 = (6,7,10)(+)'(7,1.8)$ be two hybrid numbers, then

$$\begin{aligned} \underline{\widetilde{A}}_1 \oplus \underline{\widetilde{A}}_2 &= [(3,5,9)(+)'(6,1.2)] \oplus [(6,7,10)(+)'(7,1.8)] \\ &= [(9,11,15)(+)'(0,1.2)] \oplus [(13,14,17)(+)'(0,1.8)] \\ &= (22,25,32)(+)'(0,3.0). \end{aligned}$$

[Downloaded from ijaor.com on 2024-10-12]

5 Problem statement

The problem considered can be stated as follows. There are k vehicles of the same capacity under an EOQ model inventory system delivering some goods from a warehouse to a set of customer nodes $N = \{1, 2, ..., n\}$ in a complete directed graph with arc set Λ. $\Lambda = \{(i, j): i, j \in N, i \neq j\}$ Euclidean distance is an arc set which assumed that the underlying distance matrix is symmetric and satisfies the triangle inequalities. At the beginning of the planning horizon, customer $i \in N$ supplied with a delivery quantity, and this process lasts to the end of the period. Each customer $i \in N$ is characterized by a demand known in advance, and may not be satisfied in an infinite time horizon which means shortage assumption is permitted. Considering the differentiations in customers' time periods, the delivery process continues while total demands fulfill. Similar planning will be projected for the next periods; therefore, restarting each period, there is a routing policy with known delivery quantities. Also it is considered that a limited amount of inventory can be stored at the customer sites as well as the warehouse from which it is delivered; however, transfers between sites are not allowed [26]. The vehicle working time is made of a set of heterogeneous routes K where each route starts and ends at the warehouse. We assume, without loss of generality, that the routes are served in the order 1, 2, ..., k. The warehouse is denoted by 0; the symbol N^+ is used for $N \cup 0$ and Λ^+ for $\Lambda = \{(i, j): i, j \in N^+, i \neq j\}$. The goal is to determine inventory policies and routing strategy such that the long-run costs are minimized to serve all customers while satisfying the capacity constraints. This problem is prevailing for some fields of food distribution systems –e.g. distributing flour to bakeries.

In the development of the model, we make use of the notation shown in Table 1.

Table 1 List of notations

Notation	Meaning
Parameters	
D_i	hybrid demand of customer $i \in N$
Q_{veh}	capacity of each vehicle
Q_{space}	capacity of warehouse
w_i	storage space per unit of product to customer $i \in N$
c_{1i}	holding cost of delivering per unit of product to customer $i \in N$
c_{2i}	shortage cost per unit of product to customer $i \in N$
c_{3i}	setup cost per unit of product to customer $i \in N$
c_{4i}	production cost per unit of product to customer $i \in N$
d_{ij}	travel distance from customer $i \in N$ to customer $j \in N$
ρ	cost value of travel distance
Decision variables	
x_{ijr}	1 if arc $(i, j) \in \Lambda^+$ is part of a vehicle route $r \in K$; 0 otherwise
v_i	load on a vehicle immediately before making a delivery to customer $i \in N$
S_i	shortage level of supply to customer $i \in N$
Q_i	amount delivered to customer $i \in N$

5.1 Model development

Let the amount of stock for the ith customer ($i \in N$) be R_i at time $t = 0$. In the interval (0, T_i (= $t_{1i} + t_{2i}$)), the inventory level gradually decreases to meet demands. By this process the

[Downloaded from ijaor.com on 2024-10-12]

inventory level reaches zero level at time t_{1i} and then shortages are allowed to occur in the interval (t_{1i}, T_i). The cycle then repeats itself (Fig. 1).

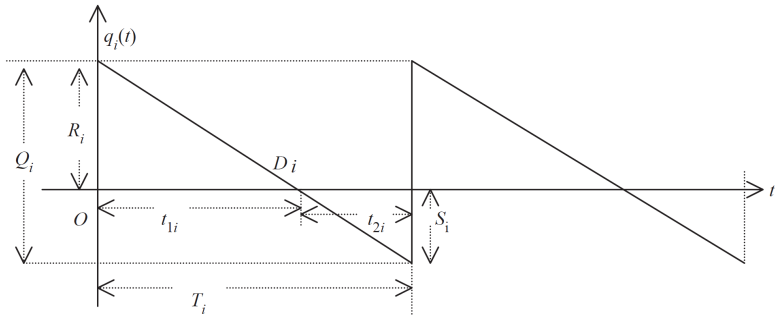

Fig. 1 Inventory level of ith customer

The differential equation for the instantaneous inventory $q_i(t)$ at time t in $(0, T_i)$ is given by

$$\frac{\mathrm{d}q_i(t)}{\mathrm{d}t} = -\underline{\tilde{D}}_i; \qquad 0 \le t \le t_{1i}$$
$$= -\underline{\tilde{D}}_i; \qquad t_{1i} \le t \le T_i$$

with the initial conditions $q_i(0) = R_i(= Q_i - S_i)$, $q_i(T_i) = -S_i$, $q_i\,(t_{1i}) = 0$.

For each period a fixed amount of shortage is allowed, and there is a penalty cost c_{2i} per items of unsatisfied demand per unit time. From the above differential equation,

$$q_i(t) = R_i - \underline{\tilde{D}}_i t; \qquad 0 \le t \le t_{1i}$$
$$= \underline{\tilde{D}}_i (t_{1i} - t); \qquad t_{1i} \le t \le T_i$$

so,

$$R_i = \underline{\tilde{D}}_i t_{1i}, \qquad S_i = \underline{\tilde{D}}_i t_{2i}, \qquad Q_i = \underline{\tilde{D}}_i T_i.$$

Holding cost $= c_{1i}\int_0^{t_{1i}} q_i(t)dt = \dfrac{c_{1i}(Q_i - S_i)^2}{2Q_i} T_i.$

Shortage cost $= c_{2i}\int_{t_{1i}}^{T_i} (-q_i(t))dt = \dfrac{c_{2i}S_i^2}{2Q_i} T_i.$

Production cost $= c_{4i}Q_i.$

the total cost = production cost + set up cost + holding cost + shortage cost

$$= c_{4i}Q_i + c_{3i} + c_{1i}\frac{(Q_i - S_i)^2}{2Q_i}T_i + c_{2i}\frac{S_i^2}{2Q_i}T_i$$

the total average cost for ith customer is

$$c_{4i}\underline{\tilde{D}}_i + c_{3i}\underline{\tilde{D}}_i \big/ Q_i + c_{1i}\frac{(Q_i - S_i)^2}{2Q_i} + c_{2i}\frac{S_i^2}{2Q_i}, \qquad i \in N.$$

[Downloaded from ijaor.com on 2024-10-12]

Model 1

$$\text{Min} \quad \sum_{i\in N}\left\{c_{4i}\underline{\tilde{D}}_i + c_{3i}\,\underline{\tilde{D}}_i \Big/ Q_i + c_{1i}\frac{(Q_i - S_i)^2}{2Q_i} + c_{2i}\frac{S_i^2}{2Q_i}\right\} + \rho\sum_{r\in K}\sum_{(i,j)\in\Lambda} d_{ij}x_{ijr} \tag{9}$$

s.t.

$$\sum_{j\in N^+}\sum_{r\in K} x_{ijr} = 1 \qquad i\in N \tag{10}$$

$$\sum_{i\in N}\sum_{r\in K} x_{ijr} = 1 \qquad j\in N^+ \tag{11}$$

$$\sum_{i\in N} x_{i\ell r} = \sum_{j\in N} x_{\ell jr} \qquad \ell\in N,\quad r\in K \tag{12}$$

$$\sum_{j\in N} x_{0jr} \le 1 \qquad r\in K \tag{13}$$

$$\sum_{i\in N}\sum_{j\in N} Q_j x_{ijr} \le Q_{veh} \qquad r\in K \tag{14}$$

$$\sum_{i\in N} w_i Q_i \le Q_{space} \tag{15}$$

$$\upsilon_j \ge \upsilon_i - Q_i + \underline{\tilde{D}}_i(1 - x_{ijr}) \qquad i\in N,\quad j\in N^+,\quad r\in K \tag{16}$$

$$x_{ijr} \in \{0,1\} \qquad (i,j)\in\Lambda^+,\quad r\in K \tag{17}$$

$$0 \le \upsilon_i \le Q_{veh} \qquad i\in N \tag{18}$$

$$0 \le S_i \le U_{si} \qquad i\in N \tag{19}$$

$$Q_i \ge 0 \qquad i\in N \tag{20}$$

The objective function (9) includes both inventory costs of each customer and transportation costs. Constraints (10)–(13) represent the routing aspect of the problem. Constraint (10) requires that only one vehicle can leave from retailer i once. Constraint (11) denotes that only one vehicle can arrive at retailer j once. Constraint (12) states that for each retailer ℓ, the entering vehicle must eventually leave this node. Constraint (13) designates that each vehicle can leave the supplier once at most. Constraint (14) is a vehicle capacity constraint and constraint (15) is the available storage area.

Constraints (16) keeps track of the load on the vehicles and guarantees, if customer i is the immediate predecessor of customer j on a route, then the load on the vehicle before visiting customer j must be less than or equal to the load just before visiting customer i minus the amount delivered, which is represented by the variable Q_i. Because the load on each vehicle is monotonically decreasing as customers are visited, (16) provides the added benefit of eliminating sub-tours.

Constraint (17) designates x_{ijr} as a 0–1 integer variable. After all deliveries are made, the fleet returns to the warehouse empty so y_0 can be set to 0. To conclude the formulation, variables are defined in (18)–(20).

5.2 Model extension

Considering the imperfect nature of the demands, the model is to convert into a deterministic version. Then by definition of $D_i = (D_{1i},D_{2i},D_{3i})(+)'(\mu_i,\sigma_i^2)$ for $i\in N$, and following the mathematical theory of hybrid numbers as described earlier the objective function (9) and constraints (16) of Model 1 extend to

[Downloaded from ijaor.com on 2024-10-12]

Model 2

$$\text{Min TC}=E\tilde{\text{T}}\text{C}(+)'(0,\text{VTC}) \tag{21}$$

s.t.

$$E\left(y_i - y_j - Q_i + \underline{\tilde{D}}_i(1-x_{ijr})\right) \le 0 \qquad i \in N, \quad j \in N^+, \quad r \in K, \tag{22}$$

$$V\left(y_i - y_j - Q_i + \underline{\tilde{D}}_i(1-x_{ijr})\right) \le \delta, \qquad i \in N, \quad j \in N^+, \quad r \in K \tag{23}$$

(10)–(15) and (17)–(20)

where $E(\cdot)$ and $V(\cdot)$ are mean and variance operators, respectively, and δ is the preset tolerable variance level of the hybrid demand delivered to customers. On the other hand, $E\tilde{T}C = (ETC_1, ETC_2, ETC_3)$ with

$$ETC_m = \sum_{i\in N}\left\{c_{4i}(D_{ri}+\mu_{i_i}) + c_{3i}\frac{(D_{ri}+\mu_{i_i})}{Q_i} + c_{1i}\frac{(Q_i - S_i)^2}{2Q_i} + c_{2i}\frac{S_i^2}{2Q_i}\right\}, \qquad m = 1,2,3.$$

so the approximated value of $E\tilde{T}C$ is

$$\begin{aligned} E\hat{T}C &= \tfrac{1}{4}(ETC_1 + 2ETC_2 + ETC_3) \\ &= \sum_{i\in N}\left\{c_{4i}(\hat{D}_i+\mu_{i_i}) + c_{3i}\frac{(\hat{D}_i+\mu_{i_i})}{Q_i} + c_{1i}\frac{(Q_i - S_i)^2}{2Q_i} + c_{2i}\frac{S_i^2}{2Q_i}\right\} \end{aligned}$$

if $\hat{D}_i = \tfrac{1}{4}(D_{1i} + 2D_{2i} + D_{3i})$.

hence Model 2 is reduced to a multiobjective mixed integer nonlinear programming problem as follows:

Model 3

$$\text{Min} \quad \{\text{AETC},\text{VTC}\} \tag{24}$$

s.t.

$$y_j \ge y_i - Q_i + (\hat{D}_i + \mu_i)(1-x_{ijr}), \qquad i \in N, \quad j \in N^+, \quad r \in K \tag{25}$$

$$\sigma_i^2(1-x_{ijr})^2 \le \delta, \qquad i \in N, \quad j \in N^+, \quad r \in K \tag{26}$$

(10)–(15) and (17)–(20)

where $AETC = E\hat{T}C + \rho\sum_{r\in K}\sum_{(i,j)\in\Lambda} d_{ij}x_{ijr}$, and $VTC = \sum_{i\in N}\left\{c_{4i}^2\sigma_i^2 + c_{3i}^2\frac{\sigma_i^2}{Q_i^2}\right\}$.

6 Solution algorithm

The proposed mixed integer nonlinear programming model is very difficult to solve. Thus, we decompose the decision variables $\{Q_i, S_i, x_{ijr}\}$ into two groups: $\{Q_i, S_i\}$ and $\{x_{ijr}\}$. The first group is associated with an inventory problem and the second group is subject to a VRP.
With the concept of decomposition, Model 3 can be rearranged as follows:

[Downloaded from ijaor.com on 2024-10-12]

Upper level:

$$\begin{aligned} &\text{Min} \quad z_1 = E\hat{T}C + z_{VRP} \\ &\text{Min} \quad z_2 = \sum_{i \in N} \left\{ c_{4i}^2 \sigma_i^2 + c_{3i}^2 \frac{\sigma_i^2}{Q_i^2} \right\} \\ &\text{s.t.} \\ &\qquad \{Q_i, S_i\} \in \Omega_1 \end{aligned} \tag{27}$$

where Ω_1 is the feasible region represented by nonnegative constraints (19) and (20) and z_{VRP} is calculated as follows:

Lower level:

$$\begin{aligned} &\text{Min} \quad z_{VRP} = \rho \sum_{r \in K} \sum_{(i,j) \in \Lambda} d_{ij} x_{ijr} \\ &\text{s.t.} \\ &\qquad \{x_{ijr}\} \in \Omega_2 \end{aligned} \tag{28}$$

where Ω_2 is the feasible region represented by constraints (10)–(15), (17), (25) and (26) with $\{Q_i, S_i\}$ given. If a given $\{Q_i, S_i\}$ causes problem (28) to be in feasible, simply let z_{VRP} equal infinity.

Model 3 is now converted into a multiobjective nonlinear programming model (27) with nonnegative constraints and a VRP in the objective function. Model (27) can be solved using either a sensitivity-analysis based or a direct search algorithm. The former uses sensitivity analysis to obtain the derivative information of the reaction function (either explicitly or implicitly) while the latter employs only functional evaluations. Since the interdependence between delivery quantity and shortage variables $\{Q_i, S_i\}$ and vehicle routes $\{x_{ijr}\}$ are too complicated and the derivative information is not available in this problem, we adopted a direct search algorithm to solve the problem. One of the most widely used direct search methods for solving nonlinear unconstrained optimization problems is the Nelder–Mead simplex algorithm [27].

In the next two subsections, the Nelder–Mead method with boundary constraints is adopted to solve the inventory problem (27) and a heuristic is proposed to solve the VRP (28).

6.1 Solving the multiobjective inventory problem

A ‘‘simplex’’ is a geometrical figure consisting, in n-dimensions, of (n+1) points $y^0; \ldots ; y^n$ [27]. If any point of a simplex is taken as the origin, the n other points define vector directions that span the n-dimension vector space.

If we randomly draw as initial starting point y^0, then we generate the other n points y^i according to the relation $y^i = y^0 + \lambda y^0 I_i$, where the I_i are n unit vectors, and λ is a turbulence factor which is which is typically equal to one (but may be adapted to the problem characteristics).

Through a sequence of elementary geometric transformations (reflection, contraction, expansion and multi-contraction; internal/external), the initial simplex y^0 moves, expands or contracts. To select the appropriate transformation, the method only uses the values of the function to be optimized at the vertices of the simplex considered. After each transformation,

[Downloaded from ijaor.com on 2024-10-12]

the current worst vertex is replaced by a better one. Trial moves shown on Fig. 2 are generated according to the following basic operations (where $\hat{y}$ called center of gravity and defined by $\hat{y}= (\Sigma_i y_i)/n$, and α, β, γ are constants):

reflection: $y^r = \hat{y} + \alpha (\hat{y} - y^n)$
expansion: $y^e = \hat{y} + \beta (y^r - \hat{y})$
internal contraction: $y^c = \hat{y} + \gamma (y^n - \hat{y})$
external contraction: $\acute{y}^c = \hat{y} + \gamma (y^r - \hat{y})$

At the beginning of the algorithm, one moves only the point of the simplex, where the objective function is worst (this point is called ''high''), and one generates another point image of the worst point. This operation is the reflection. If the reflected point is better than all other points, the method expands the simplex in this direction; otherwise, if it is at least better than the worst one, the algorithm performs again the reflection with the new worst point. The contraction step is performed when the worst point is at least as good as the reflected point, in such a way that the simplex adapts itself to the function landscape and finally surrounds the optimum. If the worst point is better than the contracted point, the multi-contraction is performed. For each rejected contraction step, we replace all y_i of the simplex by $\frac{1}{2}(y^i+y^1)$ (y^1 is the vertex of the simplex where the objective function is ''low''); thus we obtain the multi-contraction (internal/external) of the simplex, and the process restarts.

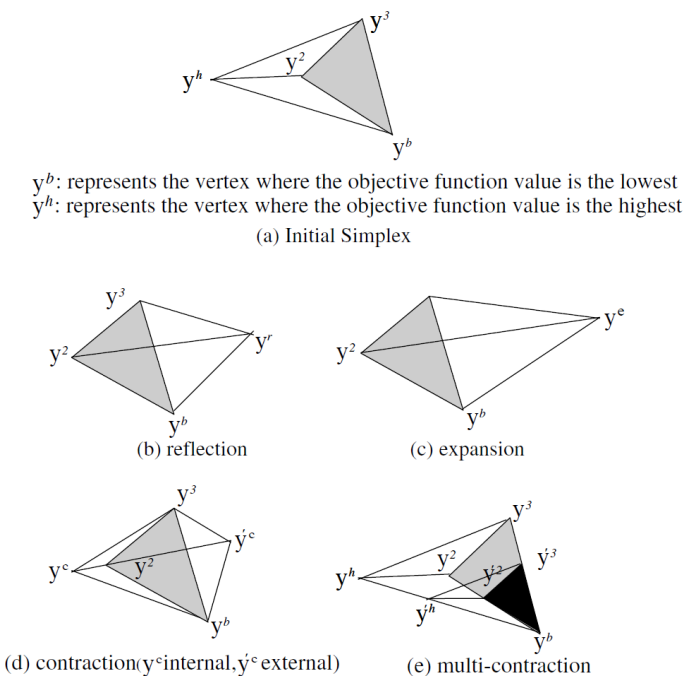

Fig. 2 Available moves in the Nelder–Mead simplex method, in the case of 3 variables

The stopping criterion is a measure of how far the simplex was moved from one iteration v to the following one (v + 1). The algorithm stops when:

$$\frac{1}{n}\sum_{i=1}^{n}\left\|y_i^v - y_i^{v+1}\right\|^2 < \varepsilon, \tag{29}$$

[Downloaded from ijaor.com on 2024-10-12]

where y^{v+1} is the vertex replacing y^{v} at the iteration (v + 1), and ε is a given ''small'' positive real number.

Because the Nelder–Mead method is originally applied to an unconstrained problem, an adjustment is necessary that projects its coordinates on the bounds if the new point is out of the domain. However, since the inventory part of the problem is of multiobjective form, it also needs a preparation step before the adjustment.

We start the preparation with a topic of normalized normal constraint method (NNCM; [28]. This method normalizes the design space and introduces new constraints. Considering the new constraints, optimization of only one of the objectives returns a non-dominated solution. When several of these single-objective optimization problems are solved, several non-dominated solutions are obtained. The difference between this method and varying user preferences in a non-generating method is that here the set of constraints are introduced to spread the final solutions uniformly in the criterion space. NNCM is an algorithm for generating a set of evenly spaced solutions on a Pareto frontier [28]. This method yields Pareto optimal solutions, and its performance is independent of the scale of the objective functions. NNCM method and some related definitions are presented in this section.

Definition 1. (utopia point) Considering a multiobjective optimization problem, a point $F^{o} \in \omega$ in the criterion space (ω) is called a utopia point if and only if:

$$f_i^o = \min\left\{ f_i(y) \middle| y \in \zeta \right\} \qquad \forall i \tag{30}$$

where $\zeta \subset \Re^{n}$ is the feasible region in the design space. Because of contradicting objectives, the utopia point is unattainable.

Definition 2. (anchor point) A non-dominated point $F^{o} \in \omega$ is an anchor point if and only if it is Pareto optimal and at least for one i, $f_i^{**} = \min_y \{ f_i(y) | y \in \zeta \}$.

The first step in NNCM is to normalize the design space. For this purpose, the utopia and the anchor points are required. These points are found by optimizing only one of the objectives at a time. After finding these points, the criterion space is normalized using the following transformation.

$$\bar{f}_i = \frac{f_i - f_i^o}{f_i^{\max} - f_i^o} \tag{31}$$

$$f_i^{\max} = \max\left\{ f_i(y); y \notin Y^* \right\} \tag{32}$$

where Y^* is All Pareto optimal points in the design space.

The normalization process locates the utopia point at the origin and the anchor points at the unit coordinates. Fig. 3a shows the original criterion space and the Pareto frontier of a generic bi-objective problem. Fig. 3b represents the Pareto frontier of the same problem after normalization. The next step is to form the utopia hyperplane, which is a hyperplane with vertices located at the anchor points. For a bi-objective problem, the utopia hyperplane is a line as shown in Fig. 3c. Next, a grid of evenly distributed points on the utopia hyperplane is generated. The number of points in this grid is defined by the user. Fig. 3c shows, for

[Downloaded from ijaor.com on 2024-10-12]

example, a grid of six points on the utopia line. If these points are projected onto the Pareto frontier, several Pareto optimum solutions are obtained. To find the Pareto optimum solution corresponding to each point in this grid, a single-objective optimization problem must be solved. This problem entails minimizing one of the normalized objectives with an additional inequality constraint. For example, the Pareto optimum solution corresponding to point P in Fig. 3c can be found by minimizing $\bar{f}_2$ while the feasible region is cut by the line passing through this point and perpendicular to the utopia line. The feasible region of this single-objective optimization problem is shown in Fig. 3c. The solution of this problem, $\bar{f}^*$, is a Pareto optimum solution for the original multiobjective problem. Other Pareto optimal points can be found by repeating the same procedure for other points on the utopia line.

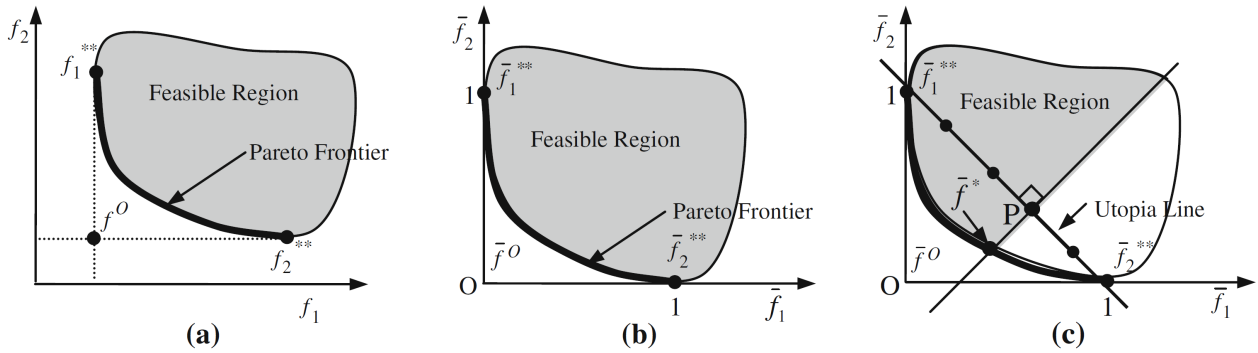

Fig. 3 a A typical bi-criterion space, **b** normalized criterion space, **c** a normal constraint introduced by NNCM and the feasible region of the resulted single-objective problem (min $\bar{f}_2$).

If the objective functions have local optima, it is possible to have some dominated solutions among the final solutions. Model 3 has local optima; therefore, dominated solutions are expected.

In order to find each Pareto optimum solution, NNCM requires solving a single-objective optimization problem. Since this algorithm is proposed for solving Model 3, in which the gradients of the objectives are not available; a direct optimization method is required. On the other hand, considering the time consuming analysis of the model, an evolutionary algorithm may not be a good choice due to the low rate of convergence. Hence, integrating with Nelder–Mead simplex algorithm would be an appropriate choice we implement here. We now proceed to formally state the solution algorithm, as follows:

Step 1: Initialization

*Step*1.1: Find an initial solution $\{Q_i, S_i\}$ (designated as y^0) of (27) as follows, and solve corresponding vehicle routing problem (28) considering NNCM.

The initial delivery quantity Q_i is usually set as the mean value of the stochastic demand quantity with the initial shortage value (S_i) of zero.

Calculate the value of objective function (27).

Step 1.2: Determine other vertices $y^1, \ldots, y^n$ of the initial simplex by disturbing y^0 as follows: $y^i = y^0 + \lambda\, y^0 I_i \quad \forall\, i = 1 \ldots n$ where λ is a turbulence factor and I_i is a unit base vector. Project its coordinates on the bounds, if y^i is out of the domain. Solve the corresponding vehicle routing problem (28) and calculate the value of objective function (27), respectively.

Step 2: Identify the vertices with the highest function value as y^h, the vertices with the lowest function value as y^l, the vertices with the second lowest function value as $\hat{y}$, the center of gravity of the simplex (without y^l), and the corresponding

[Downloaded from ijaor.com on 2024-10-12]

objective function values as $z(y^h)$, $z(y^l)$, $z(\hat{y})$; where z is the combined objective functions z_1, z_2 calculated by NNCM.

Step 3: Apply a reflection with respect to y^l: $y^r = \hat{y} + \alpha(\hat{y} - y^l)$

Project its coordinates on the bounds, if y^r is out of the domain.

Step 4: Update the simplex. We distinguish between three cases:

(a) If $z(y^r) > z(y^h)$, it means that the reflection created a better solution. We attempt to get an even better point through expansion of y^r:

$y^e = \hat{y} + \beta(y^r - \hat{y})$. Project its coordinates on the bounds, if necessary. Replace y^l with y^e if $z(y^e) > z(y^r)$; otherwise, replace y^l with y^r.

(b) If $z(y^r) \geq z(\hat{y})$, replace y^l with y^r.

(c) If $z(y^r) \leq z(y^l)$, it was probably wrong to do the reflection along that direction.

An internal contraction from y^l in direction $\hat{y} - y^l$ will be applied:

$y^c = \hat{y} + \gamma(y^l - \hat{y})$, project its coordinates on the bounds, if necessary. Or else, if $z(y^l) < z(y^r) < z(\hat{y})$, the selected direction may be right. However, since all vertices except y^l are better than y^r, it can be concluded to go closer to the simplex again. An external contraction from y^r will be applied:

$\acute{y}^c = \hat{y} + \gamma(y^r - \hat{y})$, project its coordinates on the bounds, if necessary. After the internal or external contraction, if $z(y^c) > z(y^l)$ (or if $z(\acute{y}^c) > z(y^l)$), replace y^l with y^c (or $\acute{y}^c$). Otherwise, a total contraction is performed since all attempts to get improvement failed.

$y^i = y^h + \gamma(y^i - y^h)\ \forall i \neq h$

Step 5: Check convergence. If the distance between y^h and any other vertices is smaller than a certain tolerance, then stop; y^h and its corresponding vehicle route is the best solution. Otherwise, go to Step2. Another choice of stopping criterion which is more applicable, according to (29), is the difference of $z(y^h)$-$z(y^l)$ less than a preset tolerance.

6.2 Solving the VRP

A review of the VRP literature reveals that the problem has been studied extensively. A number of exact and approximate algorithms exist. While exact algorithms can only solve relatively small-size problems, several heuristic algorithms have proved very successful. Tabu search (TS) is declared to be the best meta-heuristic for the VRP by Cordeau et al. [29] and Laporte et al. [30]. This global optimization meta-heuristic was initially proposed by Glover [31]. The basic idea of this method is exploring the solution space by moving at each iteration from the current solution s to the best solution in its neighborhood $N(s)$. Since the current solution may deteriorate during the search, anti-cycling rules must be implemented. In our implementation, we use a Tabu search method to solve VRP. The solution algorithm is used as follows:

Step 1: *Input data*

Input the amount delivered and shortage level of supply to customer $\{Q_i, S_i\}$ calculated from the inventory problem.

[Downloaded from ijaor.com on 2024-10-12]

Step 2: *Initial solution*

In order to generate an initial solution for our TS, we make use of the Nearest Neighborhood method where customers are placed in an array sorted in the non-decreasing order of a distance. In this method, the customer with the lowest distance is appended to a route. When the next to-be-inserted customer's demand exceeds the vehicle capacity on the current route, a new route is initiated.

Step 3: *Neighborhood structure*

Our neighborhood structure is a neighborhood heuristic that is based on exchanging one node (customer) between a given set of initial vehicle routes. These exchanges are called 1-exchange. The 1-exchange scheme involves shifting a node from one route to another and swapping two nodes between two given routes. A conventional tabu list is created to prevent a customer that has been moved to return to its original route. Whenever one node is moved from a route to another one at iteration v, it may not be reinserted into previous route until iteration $v+v'$, where v' is randomly selected on in some interval $[\theta,\theta']$. In our implementation, we used $\theta=5$ and $\theta'=10$. This Tabu tenure mechanism was first suggested by Gendreau et al. [32] and virtually eliminates the probability of cycling. As is common in Tabu search, the algorithm uses an aspiration criterion that overrides the Tabu status of a node whenever moving it results in a new best value. Tabu search terminates when any one of two stopping criteria is satisfied. The first criterion is the total number of iterations performed. The second criterion is the maximum permissible number of iterations during which the best feasible does not improve – the one which we used in this paper.

7 Computational result

All computations were performed on a 2.53 GHz processor with 512 MB of RAM. The optimization models and the Tabu search were coded with MATLAB 7. For testing purposes, we used the three data sets containing 10 instances of 5, 6, 7, 8, 9, 10, 15, 20, 50 and 100 customer problems and holding costs, shortage cost, setup cost, production cost, storage space, respectively, $c_{1i} = 2$, $c_{2i} = 1$, $c_{3i} = 30$ and $w_i = 1$ for all $i \in N$, and since the first term (relating to c_{4i}) in (9) is constant as that is independent of decision variables, therefore at present the term can be omitted from the analysis.

These instances were randomly generated on a 100 × 100 Euclidean grid. For each customer, demand value lie within [0,100]. Capacity of each vehicle $Q_{veh} = 50$, capacity of warehouse $Q_{space} = 200$, cost value of travel distance $\rho = 0.1$.

On the other side, according to the original Nelder–Mead method, the coefficients in the simplex iteration should satisfy $\alpha > 0$, $\beta > 1$ and $0 < \gamma < 1$. The standard, nearly universal, choices for these values are $\alpha = 1$, $\beta = 2$ and $\gamma = ½$. We also use these values in this paper.

To show the efficiency, we also compare the resulted solution with mathematical software LINGO 8.0. Since (27) is a multiobjective problem which its objective functions has conflict, we used LP-metric approach to compare the result with our proposed algorithm. In short, we solved (27) regarding to each objective function separately and reformulated it, aim to minimize the summation of normalized difference between each objective and its optimal values.

[Downloaded from ijaor.com on 2024-10-12]

As the complexity of this Problem is *NP*-hard, only small problems can be solved. The test results are summarized in Table 2. In each case, LINGO 8.0 returns a local optimal solution.

Table 2 Comparisons between proposed algorithm and LINGO for different problem sizes

Number of customers	Objective Value		CPU time (Sec)		Gap of Objective value (%)
	Proposed algorithm	LINGO	Proposed algorithm	LINGO	
5	286.34	301.66	0	4560	5.07
6	396.26	445.85	0	5898	11.12
7	502.80	673.01	1	4244	25.29
8	592.39	696.27	1	6774	14.91
9	711.08	856.22	4	4552	16.95
10	834.89	971.75	9	5667	14.08
15	1229.23	-	28	-	-
20	1433.67	-	122	-	-
50	4825.34	-	864	-	-
100	8777.91	-	71378	-	-

Table 2 shows that the solutions of proposed algorithm are better than the local optimal solutions found by LINGO 8.0 in all cases. The CPU times of the proposed algorithm are less than 10 s in the cases which have 5 to 10 customers. However, LINGO 8.0 takes hours to find a local optimal solution in those cases. Also the proposed algorithm solutions are 14.56% better, on average, than the LINGO 8.0 solutions.

From the above test results, we found that the proposed algorithm can solve this problem efficiently and returns a reliable solution. The efficiency of the algorithm makes it suitable for solving a real case, which is normally large scale.

8 Conclusion

In this paper, we have successfully developed a multi-objective mixed integer nonlinear programming model and proposed an effective and efficient heuristic method (in which demands are considered as hybrid numbers, i.e. mixture of both fuzzy and random numbers of normal density functions) for the vehicle routing problem with inventory (HVRPI). The proposed heuristic method is better than LP-metric approach, based on the minimal transportation cost criterion in terms of average logistic cost. Other contribution and some conclusions are summarized as follows:

The HVRPI is tactically decomposed into two mutually dependent “simpler” sub-problems and each sub-problem can be readily solved by some existing algorithms. Even with a small number of customers, solving the HVRPI by means of optimization software may take a long time. Apparently such optimization software is not suitable for solving HVRPI. However, the proposed algorithm in this paper is much more efficient and effective in solving HVRPI.

Due to the limitations of this paper, some factors such as multiple products, time windows, etc. are not considered. So considering these factors would help the vehicle routing and inventory decision made more realistically.

[Downloaded from ijaor.com on 2024-10-12]

The Tabu search (TS) algorithm described in this paper utilized only short term memory. Although it provided substantial improvement in solution quality, it may be possible to further improve performance by implementing longer term memory of TS. Incorporating long term strategies into TS is crucial to fully utilizing its capabilities. We leave long term implementation for future work.

References

1. Dantzig, G. B., Ramser, J. H., (1959). The truck dispatching problem. Management Science, 6(1), 80–91.
2. Clarke, C., Wright, J. W., (1964). Scheduling of vehicles from a central depot to a number of delivery points, Operations Research, 12(4), 568–581.
3. Savelsbergh, M., Song, J. H., (2007). Inventory routing with continuous moves. Computers & Operations Research, 34(6), 1744–1763.
4. Warters, C. D. J., (1989). Vehicle-scheduling problems with uncertainty and omitted customers. Journal of the Operational Research Society, 40(12), 1099–1108.
5. Tillman, F. A., (1969). The multiple terminal delivery problem with probabilistic demands. Transportation Science, 3(3), 192–204.
6. Stewart, J., Golden, B. L., (1983). Stochastic vehicle routing: A comprehensive approach. European Journal of Operational Research, 14(4), 371–385.
7. Golden, B. L., Yee, J. R., (1979). A framework for probabilistic vehicle routing. IIE Transactions, 11(2), 109–112.
8. Bodin, L. D., Golden, B. L., Assad, A. A., Ball, M. O., (1983). Routing and scheduling of vehicles and crews: The state of the art, Computers & Operations Research, 10(2), 63–211.
9. Teodorović, D., Pavkovićkm, G., (1996). The fuzzy set theory approach to the vehicle routing problem when demand at nodes is uncertain. Fuzzy Sets and Systems, 82(3), 307–317.
10. Lai, K. K., Liu, B., Peng, J., (2003). Vehicle routing problem with fuzzy travel times and its genetic algorithm, Technical Report.
11. Zheng, Y., Liu, B., (2006). Fuzzy vehicle routing model with credibility measure and its hybrid intelligent algorithm. Applied Mathematics and Computation, 176(2), 673–683.
12. Sommer, G., (1981). Fuzzy inventory scheduling. in: G. Lasker (Ed.), Applied Systems and Cybernetics, Academic Press, New York.
13. Park, K. S., (1987). Fuzzy set theoretic interpretation of economic order quantity. IEEE Transactions on Systems, Man, and Cybernetic, 17(6), 1082–1084.
14. Das, K., Roy, T. K., Maiti, M., (2000). Multi-item inventory model with quantity dependent inventory costs and demand-dependent unit cost under imprecise objective and restrictions: a geometric programming approach, Production Planning & Control, 11(8), 781–788.
15. Roy, T. K., Maiti, M., (1997). A fuzzy EOQ model with demand-dependent unit cost under limited storage capacity. European Journal of Operational Research, 99(2), 425–432.
16. De, S., Goswami, A., (2001). A replenishment policy for items with finite production rate and fuzzy deterioration rate, OPSEARCH, 38(4), 419–430.
17. He, Y., Xu, J., (2005). A class of random fuzzy programming model and its application to vehicle routing problem, World Journal of Modelling and Simulation, 1(1), 3–11.
18. Malekly, H., Haddadi, B., Tavakkoli-Moghadam, R., (2009). A Fuzzy Random Vehicle Routing Problem: The Case of Iran. In IEEE proceeding of 39rd Computers & Industrial Engineering, Troyes, France, 233–238.
19. Dutta, P., Chakraborty, D., Roy, A. R., (2005). A Single-Period Inventory Model with Fuzzy Random Variable Demand, Mathematical and Computer Modelling, 41(8-9), 915-922.
20. Dutta, P., Chakraborty, D., Roy, A. R., (2007). Continuous review inventory model in mixed fuzzy and stochastic environment. Applied Mathematics and Computation, 188(1), 970–980.
21. Zadeh, L. A., (1965). Fuzzy sets. Information and control, 8(3), 338–353.
22. Kaufmann, A., Gupta, M. M., (1991). Introduction to Fuzzy Arithmetic: Theory and Applications, Van Nostrand, Reignhold, New York.
23. Dubois, D., Prade, H., (1980). Fuzzy Sets and Systems: Theory and Applications, Academic Press, New York.

[Downloaded from ijaor.com on 2024-10-12]

24. Liu, B., Iwamura, K. B., (1998). A note on chance constrained programming with fuzzy Coefficients. Fuzzy Sets and Systems, 100(1-3), 229–233.
25. Liu, B., Iwamura, K. B., (1998). Chance constraint Programming with fuzzy parameters. Fuzzy Sets and Systems, 94(2), 227–237.
26. Herer, Y. T., Tzur, M., Yücesan, E., (2006). The multi-location transshipment problem. IIE Transactions, 38(3), 185–200.
27. Nelder, J. A., Mead, R., (1965). A Simplex for Function Minimization. Computer Journal, 7(4), 308–313.
28. Messac, A., Ismail-Yahaya, A., Mattson, C. A., 2003. The normalized normal constraint, method for generating the Pareto frontier. Structural and Multidisciplinary Optimization, 25(2), 86–98.
29. Cordeau, J. F., Gendreau, M., Laporte, G., Potvin, J. Y., Semet, F., (2002). A guide to vehicle routing heuristics. Journal of the Operational Research Society, 53(5), 512–522.
30. Laporte, G., Gendreau, M., Potvin, J. Y., Semet, F., (2000). Classical and modern heuristics for the vehicle routing problem. International Transactions in Operational Research, 7(4), 285–300.
31. Glover, F., (1986). Future paths for integer programming and links to artificial intelligence. Computers & Operations Research, 13(5), 533-549.
32. Gendreau, M., Hertz, A., Laporte, G., (1994). A tabu search algorithm for the vehicle routing problem. Management Science, 40(10), 1276–1290.

[Downloaded from ijaor.com on 2024-10-12]