A Stochastic Single Vehicle Routing Problem with a Predefined Sequence of Customers and Collection of Two Similar Materials

Epaminondas G. Kyriakidis, Theodosis D. Dimitrakos & Constantinos C. Karamatsoukis

Methodology and Computing in Applied Probability

ISSN 1387-5841

Methodol Comput Appl Probab
DOI 10.1007/s11009-019-09759-9

ONLINE FIRST

Springer

Your article is protected by copyright and all rights are held exclusively by Springer Science+Business Media, LLC, part of Springer Nature. This e-offprint is for personal use only and shall not be self-archived in electronic repositories. If you wish to self-archive your article, please use the accepted manuscript version for posting on your own website. You may further deposit the accepted manuscript version in any repository, provided it is only made publicly available 12 months after official publication or later and provided acknowledgement is given to the original source of publication and a link is inserted to the published article on Springer's website. The link must be accompanied by the following text: "The final publication is available at link.springer.com".

Springer

Author's personal copy

Methodology and Computing in Applied Probability
https://doi.org/10.1007/s11009-019-09759-9

A Stochastic Single Vehicle Routing Problem with a Predefined Sequence of Customers and Collection of Two Similar Materials

Epaminondas G. Kyriakidis[1] · **Theodosis D. Dimitrakos**[2] · **Constantinos C. Karamatsoukis**[3]

Received: 17 May 2018 / Revised: 12 November 2019 / Accepted: 27 November 2019 /
Published online: 16 January 2020
© Springer Science+Business Media, LLC, part of Springer Nature 2020

Abstract

We suppose that a vehicle visits N ordered customers in order to collect from them two similar but not identical materials. The actual quantity and the actual type of material that each customer possesses become known only when the vehicle arrives at the customer's location. It is assumed that the vehicle has two compartments. We name these compartments, Compartment 1 and Compartment 2. It is assumed that Compartment 1 is suitable for loading Material 1 and Compartment 2 is suitable for loading Material 2. However it is permitted to load items of Material 1 into Compartment 2 and items of Material 2 into Compartment 1. These actions cause extra costs that are due to extra labor. It is permissible for the vehicle to interrupt its route and go to the depot to unload the items of both materials. The costs for travelling from each customer to the next one and the costs for travelling from each customer to the depot are known. The objective is to find the routing strategy that minimizes the total expected cost among all possible strategies for servicing all customers. A dynamic programming algorithm is designed for the determination of the routing strategy that minimizes the total expected cost among all possible strategies. The structure of optimal routing strategy is characterized by a set of critical numbers for each customer.

Keywords Stochastic dynamic programming · Vehicle routing problem

Mathematics Subject Classification (2010) Primary 90C39 · Secondary 90B06

✉ Epaminondas G. Kyriakidis
ekyriak@aueb.gr

Theodosis D. Dimitrakos
dimitheo@aegean.gr

Constantinos C. Karamatsoukis
ckaramatsoukis@sse.gr

[1] Department of Statistics, Athens University of Economics and Business, Patission 76, 10434, Athens, Greece

[2] Department of Mathematics, University of the Aegean, 83200, Samos, Greece

[3] Department of Military Sciences, Hellenic Military Academy, Vari, 16673, Attica, Greece

Author's personal copy

1 Introduction

One of the most widely studied topics in the field of Operations Research is the vehicle routing problem (VRP). It is a combinatorial optimization problem that generalizes the well-known traveling salesman problem (TSP). It is also an important problem in the fields of transportation, distribution and logistics. The main topic of the VRP is the determination of the optimal routing strategy of a single vehicle or of a fleet of vehicles which deliver products to customers that are located at different points in a geographical area. A first version of the VRP appeared in a paper by Dantzig and Ramser (1959), where an algorithm was developed and was applied to petrol deliveries. Clarke and Wright (1964) improved Dantzig and Ramser's approach using an effective greedy approach called the savings algorithm. In many VRPs it is assumed that the vehicles collect goods from the customers. The VRP has been extensively studied in the optimization literature during the last fifty-five years. Recent surveys of research on VRP can be found in Pillac et al. (2013), Toth and Vigo (2014), Psaraftis et al. (2016) and Ritzinger et al. (2016). It is noteworthy that a great amount of research is related to Stochastic VRP (see e.g. Gendreau et al. (1996), Haugland et al. (2007), Nguyen et al. (2016)) that contains stochastic elements, as the demands of the customers, the vehicle travel times and the service times of the customers.

Two important variations of the VRP are the VRP with time windows and the capacitated VRP. In the VRP with time windows, the customers are served within predefined time windows. In the capacitated VRP (with or without time windows), the vehicles have limited carrying capacity of the goods that must be delivered. In the last nineteen years various capacitated routing problems with additional characteristics such as a single compartment, with multiple compartments, with pickup and delivery, with deterministic demands, with stochastic demands, with full or partial satisfaction of demands, with time windows, with delivery of two similar products have been studied. In these capacitated vehicle routing problem versions, a single vehicle starts its route from a depot and serves N customers according to a predefined order. We refer to the papers by Yang et al. (2000), Kyriakidis and Dimitrakos (2008, 2013), Tsirimpas et al. (2008), Tatarakis and Minis (2009), Minis and Tatarakis (2011), Pandelis et al. (2012), Pandelis et al. (2013a, b), Dimitrakos and Kyriakidis (2015), Zhang et al. (2016), Kyriakidis et al. (2019). These problems have been solved by implementing suitable dynamic programming algorithms. It was shown that the structure of the optimal routing strategy is of threshold-type, i.e. the optimal action, when the vehicle visits a customer, depends on some critical numbers. In the present work we consider another vehicle routing problem under the assumption that the customers are serviced according to a predefined sequence. The vehicle starts its route from a depot. It collects from the customers two similar but not identical materials (Material 1, Material 2). The size of an item is the same for both materials. The vehicle has two compartments (Compartment 1, Compartment 2) with same capacity. Compartment 1 is suitable for loading items of Material 1 and Compartment 2 is suitable for loading items of Material 2. If a compartment is full, it is permissible to load the corresponding material into the other compartment. In this case a penalty cost is incurred that is due to some extra labor for separating the two materials when the vehicle returns to the depot to unload. Each customer has only one type of material. The probability that a customer has items of Material 1 or Material 2 is known before the arrival of the vehicle at the customer's site. The quantity of the material that a customer possesses is a random variable with known distribution. The type of the material that a customer possesses and its actual quantity are revealed only when the vehicle visits him/her. The vehicle may interrupt its route by returning to the depot to unload

Author's personal copy

the materials in both compartments. The total cost for servicing all customers consists of (i) costs for travelling from a customer to the next one, (ii) costs for travelling from the customers to the depot and (iii) penalty costs. A dynamic programming algorithm is given for the determination of the optimal routing strategy of the vehicle. It is shown that the optimal routing strategy has a specific threshold-type structure. This characterization enables us to construct an efficient special-purpose dynamic programming algorithm that operates over the routing strategies having this structure.

The vehicle routing problem that we introduce in the present paper can be considered as complementary to the vehicle routing problem that was studied in Kyriakidis et al. (2019). In Kyriakidis et al. (2019), it was assumed that the vehicle transfers two similar but different materials that are stored in a single compartment of the vehicle. The materials are delivered to N ordered customers according to their preferences. In the present paper, it is assumed that the vehicle has two compartments that are suitable for loading two similar but different materials. The vehicle visits N ordered customers to collect the materials that are loaded in suitable compartments. The problems are solved by dynamic programming algorithms. In both problems, the decision epochs are chosen as the time epochs at which the vehicle visits each customer for the first time and the maximum possible service has been offered. The possible actions that are selected in the collection problem differ significantly from the possible actions that are selected in the delivery problem. For example, in the delivery problem, in some cases, the vehicle after the first visit at a customer's site may return to the depot to restock with some items of Material 1 and some other items of Material 2 and then proceed to the next customer, while, in the collection problem, the vehicle after the first visit at a customer's site may return to the depot to unload both compartments, and then proceed to the next customer.

A realistic application of the problem could be the collection of empty plastic and empty glass bottles with same size from minimarkets that are located in a geographical area. A vehicle with two compartments starts its route from a recycling plant and visits the minimarkets in order to collect the bottles. The plastic bottles are placed in Compartment 1 of the vehicle and the glass bottles are placed in Compartment 2. It is permissible to load bottles in unsuitable compartments. In this case penalty costs are incurred that are due to extra labor which is needed to separate the two kinds of bottles when the vehicle returns to the recycling plant to unload. We refer to a recent paper by Markov et al. (2020), who studied a routing problem for the collection of recyclable waste. Another real-world application of the proposed model (see Elgesem et al. (2018)) could be related to maritime transportation where a cargo ship with two compartments collects dry bulk (ores, cotton, corn) or liquid bulk (crude oil, fuels, chemicals) from charterers (customers) in order to unload the materials to the central port (depot).

The rest of the paper is organized as follows. In Section 2 the problem is specified and analyzed for the case in which the quantities of the materials that are collected from the customers are discrete random variables. A dynamic programming algorithm is presented that leads to the optimal routing strategy of the vehicle. The form of the optimal routing strategy is presented and an efficient special-purpose dynamic programming algorithm is designed. In Section 3 similar structural results concerning the optimal routing strategy are obtained when the quantities of the materials that are collected from the customers are continuous random variables. In Section 4 the theoretical results are illustrated by numerical examples. In Section 5 we investigate the more general problem without assuming that the customers are ordered. A summary of the main results of the paper and a topic for future research are presented in the last section.

Author's personal copy

2 The Problem and the Optimal Routing Strategy

2.1 The Problem

We assume that a vehicle starts its route from a depot and visits N customers in order to collect from them two similar but not identical materials. We name these materials, Material 1 and Material 2. An item of Material 1 does not differ in size from an item of Material 2. For example, an item of Material 1 could be an empty plastic bottle and an item of Material 2 could be an empty bottle of glass with the same size. The customers are serviced according to a predefined sequence $1 \rightarrow 2 \rightarrow \cdots \rightarrow N$. This means that, first, the materials of customer 1 are collected, then the materials of customer 2 are collected, then the materials of customer 3 are collected and so on. As soon as all materials of the last customer N have been collected, the vehicle returns to the depot and its route is completed. Suppose that (i) the vehicle has two compartments, Compartment 1 and Compartment 2. Material 1 is loaded in Compartment 1, while Material 2 is loaded in Compartment 2, (ii) both compartments have the same capacity, which is equal to Q items, (iii) each customer $i \in \{1, \ldots, N\}$ has items only of Material 1 or only of Material 2, (iv) the quantity $\xi_i \in \{0, \ldots, Q\}$ that will be collected from customer $i \in \{1, \ldots, N\}$ is a discrete random variable with known distribution, (v) Material 1 will be collected from customer $i \in \{1, \ldots, N\}$ with probability p_i, (vi) Material 2 will be collected from customer $i \in \{1, \ldots, N\}$ with probability $1 - p_i$, (vii) the material and the actual quantity that will be collected from customer $i \in \{1, \ldots, N\}$ is revealed only when the vehicle arrives at customer's i site, (viii) it is permissible to load units of Material 1 in Compartment 2, if Compartment 1 is full. In this case a penalty cost is incurred that is equal to π_i per unit of Material 1 that is loaded in Compartment 2. It is also permissible to load units of Material 2 in Compartment 1, if Compartment 2 is full. In this case a penalty cost is incurred that is equal to π_i per unit of Material 2 that is loaded in Compartment 1. Let $c_{i,i+1},\ \ i = 1, \ldots, N-1$ be the travel cost from customer i to customer $i+1$. Let also c_{i0} and $c_{0i},\ \ i = 1, \ldots, N$ be the travel cost from customer i to the depot and the travel cost from the depot to customer i, respectively. These costs can be considered as the costs of the required fuel that the vehicle needs to cover the distances between consecutive customers and the distances between customers and the depot. It is reasonable to assume that they satisfy the following properties:

$$c_{i0} = c_{0i},\ \ i = 1, \ldots, N \text{ (symmetric property)}$$

and

$$c_{0i} + c_{i,i+1} \geq c_{0,i+1},\ \ i = 1, \ldots, N-1 \text{ (triangle property)}$$

The road network is depicted in Fig. 1.

The vehicle may interrupt its route and return to the depot to unload units of Material 1 and Material 2.

Suppose that the vehicle has visited customer $i \in \{1, \ldots, N\}$. The material and the actual quantity that customer i possesses have been revealed. The maximum possible quantity of the material that customer i possesses is collected and put in the suitable compartment of the vehicle. Let (z_1, z_2) be the state of the process after the first visit to customer i, where $z_i,\ \ i = 1, 2$ is the total number of items of Material 1 and of Material 2 that have been loaded in compartment i of the vehicle after the first visit to customer i and after loading the maximum possible quantity of the items of the materials that customer possesses into the suitable compartment of the vehicle. There are three cases:

Author's personal copy

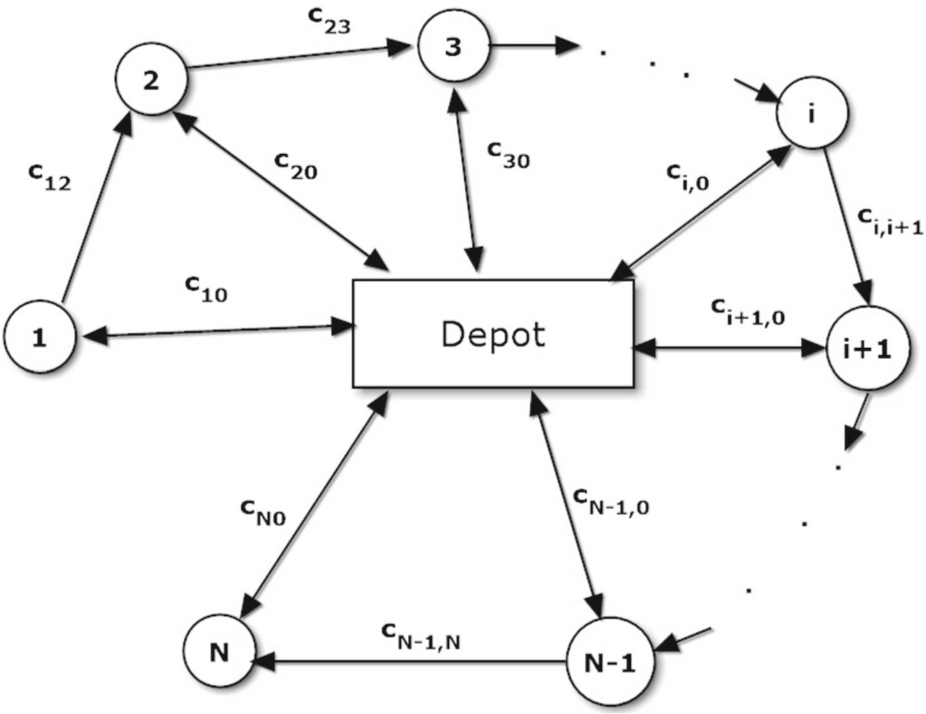

Fig. 1 The road network

Case 1: $0 \le z_1 \le Q,\ 0 \le z_2 \le Q$. In this case all items of the material that customer i possesses are loaded in the suitable compartment of the vehicle.

Case 2: $Q + 1 \le z_1 \le 2Q,\ 0 \le z_2 \le Q$. In this case customer i has units of Material 1 and there is no space in Compartment 1 for $z_1 - Q$ units of Material 1. We separate this case into Case 2a when $z_1 - Q \le Q - z_2$ and Case 2b when $Q - z_2 < z_1 - Q$. In Case 2a the whole quantity of $z_1 - Q$ units of Material 1 can be loaded in Compartment 2. In Case 2b a quantity up to $Q - z_2$ units of Material 1 can be loaded in Compartment 2.

Case 3: $0 \le z_1 \le Q,\ Q + 1 \le z_2 \le 2Q$. In this case customer i has units of Material 2 and there is no space in Compartment 2 for $z_2 - Q$ units of Material 2. We separate this case into Case 3a when $z_2 - Q \le Q - z_1$ and Case 3b when $Q - z_1 < z_2 - Q$. In Case 3a the whole quantity of $z_2 - Q$ units of Material 2 can be loaded in Compartment 1. In Case 3b a quantity up to $Q - z_1$ units of Material 2 can be loaded in Compartment 1.

Suppose $i \in \{1, \dots, N - 1\}$:

In Case 1 the possible actions are Action 1 and Action 2. Action 1 means that the vehicle proceeds to customer $i + 1$ and Action 2 means that it goes to the depot to unload and then goes to customer $i + 1$. In Case 2a the possible actions are Action 3, Action 4, Action $5_\theta,\ \theta \in \{0, \dots, z_1 - Q - 1\}$ and Action 6. Action 3 means that $z_1 - Q$ units of Material 1 are put in Compartment 2 and the vehicle proceeds to customer $i + 1$. Action 4 means that $z_1 - Q$ units of Material 1 are put in Compartment 2, the vehicle goes to the depot to unload and then goes to customer $i + 1$. Action 5_θ means that θ units of Material 1 are

Author's personal copy

put in Compartment 2, then the vehicle goes to the depot to unload, returns to customer i to put $z_1 - Q - \theta$ items of Material 1 to Compartment 1 and then goes to customer $i + 1$. Action 6 means that the vehicle goes to the depot to unload, returns to customer i to put $z_1 - Q$ units of Material 1 to Compartment 1, makes a second trip to the depot to unload and then goes to customer $i + 1$. In Case 2b the possible actions are Action 6 and Action 7_θ, $\theta \in \{0, \ldots, Q - z_2\}$. Action 7_θ means that θ units of Material 1 are put in Compartment 2, then the vehicle goes to the depot to unload, returns to customer i to put $z_1 - Q - \theta$ items of Material 1 to Compartment 1 and then goes to customer $i + 1$. Note that Actions 3 and 4 cause a penalty cost that is equal to $\pi_i(z_1 - Q)$, while Actions 5_θ and 7_θ cause a penalty cost that is equal to $\pi_i \theta$. It is assumed that if Action 5_θ or Action 7_θ is selected, there are no extra units of Material 1 or 2 to be collected when the vehicle returns to customer i, i.e. ξ_i remains unaltered.

Suppose that $i = N$:

In Case 1 the only possible action for the vehicle is to return to the depot to terminate its route. In Case 2a the possible actions are Action 8 and Action 9. Action 8 means that $z_1 - Q$ units of Material 1 are put in Compartment 2 of the vehicle and then the vehicle goes to the depot to terminate its route. Action 9 means that the vehicle goes to the depot to unload then returns to customer N to collect $z_1 - Q$ units of Material 1 that are put in Compartment 1 and then goes to the depot to terminate its route. If Action 8 is selected a penalty cost is incurred that is equal to $\pi_N(z_1 - Q)$. In Case 2b the only possible action is Action 9. It is assumed that if Action 9 is selected then there are no extra units of Material 1 or 2 to be collected when the vehicle returns to customer N, i.e. ξ_N remains unaltered.

Note that in Case 3a and Case 3b for $i \in \{1, \ldots, N\}$ the possible actions are the same as in Case 2a and Case 2b by taking into account that there is no space in Compartment 2 for units of Material 2. Our objective is to find the routing strategy that minimizes the total expected cost for the service of all customers. This routing strategy minimizes the expected total cost from the beginning of the route until its end. The total cost consists of travel costs between consecutive customers and between customers and the depot and penalty costs that incur when units of Material 1 are loaded in Compartment 2 of the vehicle or units of Material 2 are loaded in Compartment 1 of the vehicle. The optimal routing strategy can be found by implementing a suitable dynamic programming algorithm.

2.2 Dynamic Programming Equations

Let $f_i(z_1, z_2)$ denote the minimum expected future cost from the first visit of the vehicle to customer $i \in \{1, \ldots, N\}$ until the end of the route, where (z_1, z_2) is the state of the process that has been defined above. For $i \in \{1, \ldots, N - 1\}$ we give below the dynamic programming equations (1)-(3) for Case 1, Case 2a and Case 2b. For Case 3a and Case 3b the dynamic programming equations are the same as Eqs. 2 and 3 if we interchange z_1 and z_2.

If $0 \leq z_1 \leq Q,\ 0 \leq z_2 \leq Q$, then

$$f_i(z_1, z_2) = min\left\{A_i(z_1, z_2), B_i\right\}, \tag{1}$$

where,

$$\begin{aligned} A_i(z_1, z_2) &= c_{i,i+1} + p_{i+1}\mathbb{E}f_{i+1}(z_1 + \xi_{i+1}, z_2) + (1 - p_{i+1})\mathbb{E}f_{i+1}(z_1, z_2 + \xi_{i+1}), \\ B_i &= c_{i0} + c_{0,i+1} + p_{i+1}\mathbb{E}f_{i+1}(\xi_{i+1}, 0) + (1 - p_{i+1})\mathbb{E}f_{i+1}(0, \xi_{i+1}). \end{aligned}$$

Author's personal copy

If $Q+1 \leq z_1 \leq 2Q,\ 0 \leq z_2 \leq Q,\ z_1 - Q \leq Q - z_2$, then

$$f_i(z_1, z_2) = min\left\{C_i(z_1, z_2),\, D_i(z_1),\, E_i(z_1),\, F_i\right\}, \tag{2}$$

$$\begin{aligned}
C_i(z_1, z_2) &= \pi_i(z_1 - Q) + c_{i,i+1} + p_{i+1}\mathbb{E}f_{i+1}(Q + \xi_{i+1}, z_1 + z_2 - Q) \\
&\quad + (1 - p_{i+1})\mathbb{E}f_{i+1}(Q, z_1 + z_2 - Q + \xi_{i+1}), \\
D_i(z_1) &= \pi_i(z_1 - Q) + c_{i0} + c_{0,i+1} + p_{i+1}\mathbb{E}f_{i+1}(\xi_{i+1}, 0) \\
&\quad + (1 - p_{i+1})\mathbb{E}f_{i+1}(0, \xi_{i+1}), \\
E_i(z_1) &= 2c_{i0} + c_{i,i+1} + \min_{0 \leq \theta \leq z_1 - Q - 1}\left\{\pi_i\theta + p_{i+1}\mathbb{E}f_{i+1}(z_1 - Q - \theta + \xi_{i+1}, 0)\right. \\
&\quad \left. + (1 - p_{i+1})\mathbb{E}f_{i+1}(z_1 - Q - \theta, \xi_{i+1})\right\}, \\
F_i &= 3c_{i0} + c_{i,i+1} + p_{i+1}\mathbb{E}f_{i+1}(\xi_{i+1}, 0) + (1 - p_{i+1})\mathbb{E}f_{i+1}(0, \xi_{i+1}).
\end{aligned}$$

If $Q+1 \leq z_1 \leq 2Q,\ 0 \leq z_2 \leq Q, Q - z_2 < z_1 - Q$, then

$$f_i(z_1, z_2) = min\left\{G_i(z_1, z_2),\, F_i\right\} \tag{3}$$

where,

$$\begin{aligned}
G_i(z_1, z_2) &= 2c_{i0} + c_{i,i+1} + \min_{0 \leq \theta \leq Q - z_2}\left\{\pi_i\theta + p_{i+1}\mathbb{E}f_{i+1}(z_1 - Q - \theta + \xi_{i+1}, 0)\right. \\
&\quad \left. + (1 - p_{i+1})\mathbb{E}f_{i+1}(z_1 - Q - \theta, \xi_{i+1})\right\}.
\end{aligned}$$

The boundary conditions are given below for Case 1, for Case 2a and for Case 2b. For Case 3a and Case 3b the boundary conditions are the same as for Case 2a and Case 2b if we interchange z_1 and z_2.
If $Q \leq z_1 \leq Q,\ 0 \leq z_2 \leq Q$, then

$$f_N(z_1, z_2) = c_{N0}. \tag{4}$$

If $Q+1 \leq z_1 \leq 2Q,\ 0 \leq z_2 \leq Q, Q - z_2 < z_1 - Q$, then

$$f_N(z_1, z_2) = 3c_{N0}. \tag{5}$$

If $Q+1 \leq z_1 \leq 2Q,\ 0 \leq z_2 \leq Q, z_1 - Q \leq Q - z_2$, then

$$f_N(z_1, z_2) = \min\left\{\pi_N(z_1 - Q) + c_{N0}, 3c_{N0}\right\}. \tag{6}$$

The minimum expected total cost during a visit cycle is equal to

$$f_0 = c_{01} + p_1\mathbb{E}f_1(\xi_1, 0) + (1 - p_1)\mathbb{E}f_1(0, \xi_1).$$

In the above equations the expected values are taken with respect to the random variables $\xi_i,\ i = 1, \ldots, N$. The terms $A_i(z_1, z_2)$ and B_i in the right-hand-side of Eq. 1 correspond to Action 1 and Action 2, respectively. The terms $C_i(z_1, z_2)$, $D_i(z_1)$, $E_i(z_1)$, F_i in the right-hand-side of Eq. 2 correspond to Action 3, Action 4, Actions 5_θ ($\theta \in \{0, \ldots, z_1 - Q - 1\}$), Action 6, respectively. The terms $G_i(z_1, z_2)$ and F_i in the right-hand-side of Eq. 3 correspond to Actions 7_θ ($\theta \in \{0, \ldots, Q - z_2\}$) and Action 6, respectively. The terms in the curly brackets in the right-hand-side of Eq. 6 correspond to Action 8 and Action 9, respectively. Lemma 1 below will be used in the proof of Theorem 1 that describes the structure of the optimal routing strategy.

2.3 Structure of the Optimal Policy

Lemma 1 $f_i(z_1, z_2),\ i = 1, \ldots, N$, *is increasing with respect to* z_1 *and* z_2.

Author's personal copy

Proof The proof is by induction on i. From Eqs. 4, 5, 6 it can be seen that $f_N(z_1, z_2)$ is increasing in z_1 and z_2. Assuming that $f_{i+1}(z_1, z_2)$ is increasing in z_1 and z_2 we will show that $f_i(z_1, z_2)$ is increasing in z_1 and z_2. We will restrict ourselves to Case 1 and Case 2, since Case 3 is similar to Case 2. Let some fixed $z_1 \in \{0, \ldots, 2Q\}$. In view of the induction hypothesis, it follows from Eqs. 1, 2, 3 that, to prove that $f_i(z_1, z_2)$ is increasing in z_2, it is enough to show that $f_i(z_1, 2Q - z_1) \le f_i(z_1, 2Q + 1 - z_1),\ z_1 \in \{Q + 1, \ldots, 2Q\}$.

The last inequality is equivalent to

$$\min\{C_i(z_1, 2Q - z_1), D_i(z_1), E_i(z_1), F_i\} \le \min\{G_i(z_1, 2Q + 1 - z_1), F_i\},\ z_1 \in \{Q+1, \ldots, 2Q\},$$

which holds since $E_i(z_1) = G_i(z_1, 2Q + 1 - z_1),\ z_1 \in \{Q + 1, \ldots, 2Q\}$. Let some fixed $z_2 \in \{0, \ldots, Q\}$. In view of the induction hypothesis, it follows from Eqs. 1, 2, 3 that to prove that $f_i(z_1, z_2)$ is increasing in z_1, it is enough to show that

$$E_i(z_1) \le E_i(z_1 + 1),\quad Q + 1 \le z_1 \le 2Q - 1, \tag{7}$$

$$f_i(Q, Q) \le f_i(Q + 1, Q), \tag{8}$$

$$f_i(Q, z_2) \le f_i(Q + 1, z_2),\ 0 \le z_2 \le Q - 1, \tag{9}$$

$$f_i(2Q - z_2, z_2) \le f_i(2Q - z_2 + 1, z_2),\ 1 \le z_2 \le Q - 1. \tag{10}$$

Note that $H(z_1, z_1 - Q - 1) \le H(z_1 + 1, z_1 - Q)$,
where,

$$\begin{aligned} H(z_1, \theta) &= \pi_i\theta + p_{i+1}\mathbb{E} f_{i+1}(z_1 - Q - \theta + \xi_{i+1}, 0) \\ &\quad +(1 - p_{i+1})\mathbb{E} f_{i+1}(z_1 - Q - \theta, \xi_{i+1}),\ Q + 1 \le z_1 \le 2\theta,\ 0 \le \theta \le Q - z_2. \end{aligned}$$

Therefore, in view of the induction hypothesis we deduce that (7) holds.

Inequality (8) is equivalent to $\min\{A_i(Q, Q), B_i\} \le \min\{G_i(Q + 1, Q), F_i\}$. This inequality holds since $B_i \le F_i$ and $B_i \le G_i(Q + 1, Q)$, which is valid due to the induction hypothesis. Inequality (9) is equivalent to

$$\min\{A_i(Q, z_2), B_i\} \le \min\{C_i(Q + 1, z_2), E_i(Q + 1), F_i\},\quad 0 \le z_2 \le Q - 1.$$

The above inequality holds since $A_i(Q, z_2) \le C_i(Q + 1, z_2)$ (due to the induction hypothesis) and $B_i \le D_i(Q + 1)$, $B_i \le E_i(Q + 1)$ (due to induction hypothesis), $B_i \le F_i$. Inequality (10) is equivalent to

$$\min\{C_i(2Q - z_2, z_2), D_i(2Q - z_2), E_i(2Q - z_2), F_i\} \le \min\{G_i(2Q - z_2 + 1, z_2), F_i\},\ 1 \le z_2 \le Q-1.$$

To show the above inequality it is enough to prove that

$$E_i(2Q - z_2) \le G_i(2Q - z_2 + 1, z_2),\ 1 \le z_2 \le Q - 1,$$

or equivalently,

$$\begin{aligned} &\min_{0\le\theta\le Q-z_2-1} \{\pi_i\theta + p_{i+1}\mathbb{E} f_{i+1}(Q - z_2 - \theta + \xi_{i+1}, 0) + (1 - p_{i+1})\mathbb{E} f_{i+1}(Q - z_2 - \theta, \xi_{i+1})\} \\ &\le \min_{0\le\theta\le Q-z_2} \{\pi_i\theta + p_{i+1}\mathbb{E} f_{i+1}(Q - z_2 + 1 - \theta + \xi_{i+1}, 0) + (1 - p_{i+1})\mathbb{E} f_{i+1}(Q - z_2 + 1 - \theta, \xi_{i+1})\},\ 1 \le z_2 \le Q - 1. \end{aligned}$$

By taking into account the induction hypothesis we deduce that the last inequality holds since the quantity in the curly brackets for $\theta = Q - z_2 - 1$ in its left-hand-side is smaller than the quantity in the curly brackets for $\theta = Q - z_2$ in its right-hand-side. □

Theorem 1 *For each customer $i \in \{1, \ldots, N - 1\}$ the structure of the optimal routing strategy is described in the following five cases:*

(i) *For $z_1 \in \{0, \ldots, Q\}$ there exists a critical integer $s_1(z_1) \ge 0$ such that if $z_2 \in \{s_1(z_1), \ldots, Q\}$ then Action 2 is optimal and if $z_2 \in \{0, \ldots, s_1(z_1) - 1\}$ then Action 1 is optimal. Moreover, $s_1(z_1)$ is non-increasing in z_1.*

Author's personal copy

(ii) *For $z_2 \in \{1, \ldots, Q\}$ there exists a critical integer $s_2(z_2) \in \{2Q - z_2 + 1, \ldots, 2Q\}$ such that if $z_1 \in \{s_2(z_2), \ldots, 2Q\}$, then Action 6 is optimal and if $z_1 \in \{2Q - z_2 + 1, \ldots, s_2(z_2) - 1\}$, then Action 7 is optimal. Moreover, $s_2(z_2)$ is non-increasing in z_2.*

(iii) *For $z_2 \in \{0, \ldots, Q-1\}$ there exists a critical integer $s_3(z_2) \in \{Q+1, \ldots, 2Q-z_2\}$ such that if $z_1 \in \{Q + 1, \ldots, s_3(z_2) - 1\}$ then Action 3 or Action 4 or Action 5_θ is optimal and if $z_1 \in \{s_3(z_2), \ldots, 2Q - z_2\}$ then Action 6 is optimal. Moreover, $s_3(z_2)$ is non-increasing in z_2.*

(iv) *For $z_1 \in \{1, \ldots, Q\}$ there exists a critical integer $s_4(z_1) \in \{2Q - z_1 + 1, \ldots, 2Q\}$ such that if $z_2 \in \{s_4(z_1), \ldots, 2Q\}$ then Action 6 is optimal and if $z_2 \in \{2Q - z_1 + 1, \ldots, s_4(z_1)-1\}$, then Action 7_θ is optimal. Moreover, $s_4(z_1)$ is non-increasing in z_1.*

(v) *For $z_1 \in \{0, \ldots, Q-1\}$ there exists a critical integer $s_5(z_1) \in \{Q+1, \ldots, 2Q-z_1\}$ such that if $z_2 \in \{Q + 1, \ldots, s_5(z_1) - 1\}$, then Action 3 or Action 4 or Action 5_θ is optimal and if $z_2 \in \{s_5(z_1), \ldots, 2Q - z_1\}$ then Action 6 is optimal. Moreover, $s_5(z_1)$ is non-increasing in z_1.*

Proof From Lemma 1 it follows that $A_i(z_1, z_2)$ is increasing in z_1 and z_2. Part (i) is a direct consequence of this result. From Lemma 1 it follows that $G_i(z_1, z_2)$ is increasing in z_1. It can also be seen that $G_i(z_1, z_2)$ is increasing in z_2. Part (ii) is a direct consequence of these results. From Lemma 1 it follows that $C_i(z_1, z_2)$ is increasing in z_1 and z_2. It can be seen that $D_i(z_1)$ is increasing in z_1. In the proof of Lemma 1 it has been shown that $E_i(z_1)$ is increasing in z_1. Part (iii) is a direct consequence of these results. Part (iv) and Part (v) can be proved in a similar way as Part (ii) and Part (iii), respectively. □

2.4 Special-Purpose Dynamic Programming Algorithm

The optimal routing strategy, i.e. the critical integers $s_1(z_1),\ z_1\{0, \ldots, Q\},\ s_2(z_2),\ z_2 \in \{1, \ldots, Q\},\ s_3(z_2),\ z_2 \in \{0, \ldots, Q-1\},\ s_4(z_1),\ z_1 \in \{1, \ldots, Q\},\ s_5(z_1),\ z_1 \in \{0, \ldots, Q-1\}$ for each customer $i \in \{1, \ldots, N-1\}$ can be found by a special-purpose dynamic programming algorithm that takes into account the structure of the optimal routing strategy as given in Theorem 1. The part of the algorithm that computes the critical integers $s_1(z_1),\ z_1\{0, \ldots, Q\},\ s_2(z_2),\ z_2 \in \{1, \ldots, Q\},\ s_3(z_2),\ z_2 \in \{0, \ldots, Q-1\}$ is presented below. The complete special-purpose dynamic programming algorithm includes the computation of the critical integers $s_4(z_1),\ z_1 \in \{1, \ldots, Q\},\ s_5(z_1),\ z_1 \in \{0, \ldots, Q-1\}$ that is similar to the computation of the critical integers $s_2(z_2),\ z_2 \in \{1, \ldots, Q\}$ and $s_3(z_2),\ z_2 \in \{0, \ldots, Q-1\}$, respectively.

Algorithm for the determination of the critical integers $s_1(z_1),\ z_1 \in \{0, \ldots, Q\},\ s_2(z_2),\ z_2 \in \{1, \ldots, Q\},\ s_3(z_2),\ z_2 \in \{0, \ldots, Q-1\}$.

Step 0. Set $f_N(z_1, z_2) = c_{N0}$ if $z_1, z_2 \in \{0, \ldots, Q\}$,
$f_N(z_1, z_2) = 3c_{N0}$, if $z_1 \in \{Q+1, \ldots, 2Q\},\ z_2 \in \{0, \ldots, Q\},\ z_1 + z_2 < 2Q$.
$f_N(z_1, z_2) = \min\{3c_{N0}, \pi_N(z_1 - Q) + c_{N0}\}$, if $z_1 \in \{Q+1, \ldots, 2Q\}$,
$z_2 \in \{0, \ldots, Q\},\ z_1 + z_2 \geq 2Q$.
Set $i = N - 1$.

Step 1. (Determination of critical integers $s_1(z_1),\ z_1 \in \{0, \ldots, Q\}$)
Compute B_i.
For $z_1 = 0, \ldots, Q$ do the following:
For $z_2 = 0, 1, \ldots$ compute $A_i(z_1, z_2)$ until $A_i(z_1, z_2) > B_i$ or $z_2 = Q + 1$.
Set $s_1(z_1) = z_2 - 1$.
Set $f_i(z_1, z_2) = A_i(z_1, z_2),\ z_2 \in \{0, \ldots, s_1(z_1)\}$ and $f_i(z_1, z_2) = B_i$,
$z_2 \in \{s_1(z_1) + 1, \ldots, Q\}$.

Author's personal copy

Step 2. (Determination of critical integers $s_2(z_2),\ z_2 \in \{1, \ldots, Q\}$)
$F_i = 2c_{i0} + B_i$.
For $z_2 = 1, \ldots, Q$ do the following:
For $z_1 = 2Q - z_2 + 1, 2Q - z_2 + 2, \ldots$ compute $G_i(z_1, z_2)$ until $G_i(z_1, z_2) > F_i$ or $z_1 = 2Q + 1$.
Set $s_2(z_2) = z_1$.
Set $f_i(z_1, z_2) = G_i(z_1, z_2),\ z_1 \in \{2Q - z_2 + 1, \ldots, s_2(z_2) - 1\}$ and $f_i(z_1, z_2) = F_i$,
$z_1 \in \{s_2(z_2), \ldots, 2Q\}$.

Step 3. (Determination of critical integers $s_3(z_2),\ z_2 \in \{0, \ldots, Q - 1\}$)
For $z_2 = 0, \ldots, Q - 1$ do the following:
For $z_1 = Q + 1, Q + 2, \ldots$ compute $C_i(z_1, z_2), D_i(z_1), E_i(z_1)$ until $F_i < min\{C_i(z_1, z_2), D_i(z_1), E_i(z_1)\}$ or $z_1 = 2Q - z_2 + 1$.
Set $s_3(z_2) = z_1$.
Set $f_i(z_1, z_2) = min\{C_i(z_1, z_2), D_i(z_1), E_i(z_1)\}$,
$z_1 \in \{Q + 1, \ldots, s_3(z_2) - 1\}$ and
$f_i(z_1, z_2) = F_i,\ z_1 \in \{s_3(z_2), \ldots, 2Q - z_2\}$.

Step 4. Set $i = i - 1$. If $i \geq 1$ go to Step 1. Otherwise stop.

The above special-purpose dynamic programming algorithm is based on the structure of the optimal routing strategy described in Theorem 1. The complexity of this algorithm can be calculated by considering Definition 7.1 in Sipser (2013) and it is found to be $O(NQ^3)$. It is more efficient than the initial dynamic programming algorithm since it requires less computations. For example, for $i = 1, \ldots, N - 1$, the quantities $A_i(z_1, z_2),\ z_2 \in \{s_1(z_1) + 2, \ldots, Q\}$, for $z_1 \in \{0, \ldots, Q\}$ are not computed, while these quantities are computed in the initial dynamic programming algorithm. In Section 4 we will compare the computations times of these algorithms in a numerical example.

3 The Problem when the Quantities that are Collected are Continuous Random Variables

3.1 The Optimal Routing Strategy with Continuous Demands

We modify the problem that we introduced in Section 2 by assuming that the quantities $\xi_i,\ i = 1, \ldots, N$ of the materials that are collected from the customers are continuous random variables and take values in the interval $[0, Q]$ with probability density function $\phi_i(x)$. A practical example with continuous demands could be the collection of two different kinds of seeds or two different kinds of building materials, for example lime and pebble. The states (z_1, z_2) of the process, where $0 \leq z_1 \leq Q,\ 0 \leq z_2 \leq Q$ or $Q + 1 \leq z_1 \leq 2Q,\ 0 \leq z_2 \leq Q$ or $0 \leq z_1 \leq Q,\ Q + 1 \leq z_2 \leq 2Q$ after the first visit at a customer's site and Action 1, Action 2, Action 3, Action 4, Actions 5_θ $(0 \leq \theta < z_1 - Q)$, Action 6, Actions 7_θ $(0 \leq \theta \leq Q - z_2)$, Action 8, Action 9 are the same as those defined in Section 2. The minimum expected future cost $f_i(z_1, z_2)$ for $i = 1, \ldots, N$ satisfies the dynamic programming equations (1)–(3) and the boundary conditions (4)–(6). The structure of the optimal routing strategy is the same as in the case of discrete $\xi_i,\ i = 1, \ldots, N$ and is given in the theorem below.

Author's personal copy

Theorem 2 *For each customer* $i \in \{1, \ldots, N-1\}$ *the structure of the optimal routing strategy is defined in the following five cases:*

(i) *For* $z_1 \in [0, Q]$ *there exists a critical number* $s_1(z_1) \geq 0$ *such that if* $z_2 \in [s_1(z_1), Q]$ *then Action 2 is optimal and if* $z_2 \in [0, s_1(z_1))$ *then Action 1 is optimal. Moreover,* $s_1(z_1)$ *is non-increasing in* z_1.

(ii) *For* $z_2 \in (0, Q]$ *there exists a critical number* $s_2(z_2) \in (2Q - z_2, 2Q]$ *such that if* $z_1 \in [s_2(z_2), 2Q]$ *then Action 6 is optimal and if* $z_1 \in (2Q - z_2, s_2(z_2))$ *then Action* 7_θ *is optimal. Moreover,* $s_2(z_2)$ *is non-increasing in* z_2.

(iii) *For* $z_2 \in [0, Q)$ *there exists a critical number* $s_3(z_2) \in (Q, 2Q - z_2]$ *such that if* $z_1 \in (Q, s_3(z_2))$ *then Action 3 or Action 4 or Action* 5_θ *is optimal and if* $z_1 \in [s_3(z_2), 2Q - z_2]$ *then Action 6 is optimal. Moreover,* $s_3(z_2)$ *is non-increasing in* z_2.

(iv) *For* $z_1 \in (0, Q]$ *there exists a critical number* $s_4(z_1) \in (2Q - z_1, 2Q]$ *such that if* $z_2 \in [s_4(z_1), 2Q]$ *then Action 6 is optimal and if* $z_2 \in [2Q - z_1, s_4(z_1))$ *then Action* 7_θ *is optimal. Moreover,* $s_4(z_1)$ *is non-increasing in* z_1.

(v) *For* $z_1 \in [0, Q)$ *there exists a critical number* $s_5(z_1) \in (Q, 2Q - z_1]$ *such that if* $z_2 \in (Q, s_5(z_1))$ *then Action 3 or Action 4 or Action* 5_θ *is optimal and if* $z_2 \in [s_5(z_1), 2Q - z_1]$ *then Action 6 is optimal. Moreover,* $s_5(z_1)$ *is non-increasing in* z_1.

3.2 Discretization of the State Space

The state space after the first visit of the vehicle at customer's $i \in \{1, \ldots, N\}$ site is the set:

$$S = \{(z_1, z_2) : 0 \leq z_1, z_2 \leq Q\} \cup \{(z_1, z_2) : 0 < z_1 \leq 2Q,\ 0 \leq z_2 \leq Q\} \cup \{(z_1, z_2) : 0 \leq z_1 \leq Q,\ Q < z_2 \leq 2Q\}.$$

A discretization of the state space is necessary for the implementation of the dynamic programming algorithm. Let ρ a relatively small number (e.g. $\rho = 0.05$ or $\rho = 0.01$). We discretize S by restricting our attention only to its points that belong to the set:

$$\tilde{S} = \{(k\rho, l\rho) : k, l = 0, \ldots, Q/\rho\} \cup \{(k\rho, l\rho) : k = Q/\rho + 1, \ldots, 2Q/\rho, l = 0, \ldots, Q/\rho\} \cup \{(k\rho, l\rho) : k = 0, \ldots, Q/\rho, l = Q/\rho + 1, \ldots, 2Q/\rho\}.$$

The minimum expected cost $f_N(k\rho, l\rho),\ (k\rho, l\rho) \in \tilde{S}$ is found by using (4)–(6) with $z_1 = k\rho,\ z_2 = l\rho$. The minimum expected cost $f_i(k\rho, l\rho),\ (k\rho, l\rho) \in \tilde{S}$ and the corresponding optimal decisions are found, recursively, for $i = N-1, N-2, \ldots, 1$ by using the dynamic programming equations (1)–(3) with $z_1 = k\rho,\ z_2 = l\rho$. The parameter θ in these equations takes values in finite sets. For example in the equation for $E_i(k\rho)$ the parameter θ takes values in the set $B = \{u\rho : u = 0, \ldots, k - Q/\rho - 1\}$. The expectations are computed approximately. For example $E_i(k\rho)$, is computed approximately as follows:

$$E_i(k\rho) = 2c_{i0} + c_{i,i+1} + \min_{\theta \in B} \Bigg[\pi_i \theta + p_{i+1} \sum_{x=0}^{Q/\rho - 1} f_{i+1}(k\rho - Q - \theta + x\rho, 0)\phi_{i+1}(x\rho)\rho + (1 - p_{i+1}) \sum_{x=0}^{Q/\rho - 1} f_{i+1}(k\rho - Q - \theta, x\rho)\phi_{i+1}(x\rho)\rho \Bigg].$$

Author's personal copy

3.3 Special-Purpose Dynamic Programming Algorithm

As in the case of discrete demands, the optimal routing strategy, i.e. the critical numbers $s_1(k\rho),\ k = 0, \ldots, Q/\rho,\ s_2(l\rho),\ l = 1, \ldots, Q/\rho,\quad s_3(l\rho),\ l = 0, \ldots, Q/\rho - 1,\quad s_4(k\rho),\ k = 1, \ldots, Q/\rho,\quad s_5(k\rho),\ k = 0, \ldots, Q/\rho - 1$ can be found by a special-purpose dynamic programming algorithm that takes into account the structure of the optimal routing strategy as given in Theorem 2. The part of this algorithm that computes the critical numbers $s_1(k\rho),\ k = 0, \ldots, Q/\rho,\ s_2(l\rho),\ l = 1, \ldots, Q/\rho,\quad s_3(l\rho),\ l = 0, \ldots, Q/\rho - 1$, is presented below. The complete special-purpose dynamic programming algorithm includes the computation of the critical numbers $s_4(k\rho),\ k = 1, \ldots, Q/\rho$ and $s_5(k\rho),\ k = 0, \ldots, Q/\rho - 1$ that is similar to the computation of the critical numbers $s_2(l\rho),\ l = 1, \ldots, Q/\rho$, and $s_3(l\rho),\ l = 0, \ldots, Q/\rho - 1$, respectively.

Algorithm for the determination of the critical numbers $s_1(k\rho),\quad k = 0, \ldots, Q/\rho,\ \ s_2(l\rho),\ l = 1, \ldots, Q/\rho,\ s_3(l\rho),\ l = 0, \ldots, Q/\rho - 1.$

Step 0. Set $f_N(k\rho, l\rho) = c_{N0}$ if $k, l \in \{0, \ldots, Q/\rho\}$, $f_N(k\rho, l\rho) = 3c_{N0}$, if
$k \in \{Q/\rho + 1, \ldots, 2Q/\rho\}$,
$l = 0, \ldots, Q/\rho,\ k + l < 2Q/\rho$.
$f_N(k\rho, l\rho) = min\,\{3c_{N0}, \pi_N(k\rho - Q) + c_{N0}\}$, if $k \in \{Q/\rho + 1, \ldots, 2Q/\rho\}$,
$l \in \{0, \ldots, Q/\rho\},\ k + l \geq 2Q/\rho$.
Set $i = N - 1$.

Step 1. (Determination of critical numbers $s_1(k\rho),\ k = 0, \ldots, Q/\rho$)
Compute B_i.
For $k = 0, \ldots, Q/\rho$ do the following:
  For $k = 0, \rho, 2\rho, \ldots$ compute $A_i(k\rho, z_2)$ until $A_i(k\rho, z_2) > B_i$ or $z_2 = Q + \rho$.
  Set $s_1(k\rho) = z_2 - \rho$.
  Set $f_i(k\rho, l\rho) = A_i(k\rho, l\rho),\ 0 \leq l \leq s_1(k\rho)/\rho,\ f_i(k\rho, l\rho) = B_i$,
  $l = s_1(k\rho)/\rho + 1, \ldots, Q/\rho$.

Step 2. (Determination of critical numbers $s_2(l\rho),\ l = 1, \ldots, Q/\rho$)
$F_i = 2c_{i0} + B_i$.
For $l = 1, \ldots, Q/\rho$ do the following:
  For $z_1 = 2Q - l\rho + \rho, 2Q - l\rho + 2\rho, \ldots$ compute $G_i(z_1, l\rho)$ until
  $G_i(z_1, l\rho) > F_i$ or $z_1 = 2Q + \rho$.
  Set $s_2(l\rho) = z_1$.
  Set $f_i(k\rho, l\rho) = G_i(k\rho, l\rho),\ k = 2Q/\rho - l + 1, \ldots, s_2(l\rho)/\rho - 1$ and
  $f_i(k\rho, l\rho) = F_i,\ k = s_2(l\rho)/\rho, \ldots, 2Q/\rho$.

Step 3. (Determination of critical numbers $s_3(l\rho),\ l = 0, \ldots, Q/\rho - 1$)
For $l = 0, \ldots, Q/\rho - 1$ do the following:
  For $z_1 = Q, Q + \rho, \ldots$ compute $C_i(z_1, l\rho), D_i(z_1), E_i(z_1)$
  until $F_i < min\{C_i(z_1, l\rho), D_i(z_1), E_i(z_1)\}$ or $z_1 = 2Q - l\rho + \rho$.
  Set $s_3(l\rho) = z_1$.
  Set $f_i(k\rho, l\rho) = min\{C_i(k\rho, l\rho), D_i(k\rho), E_i(k\rho)\},\ k = Q/\rho + 1, \ldots,$
  $s_3(l\rho)/\rho - 1$ and $f_i(k\rho, l\rho) = F_i,\ k = s_3(l\rho)/\rho, \ldots, 2Q/\rho - l$.

Step 4. Set $i = i - 1$. If $i \geq 1$ go to Step 1. Otherwise stop.

The above special-purpose dynamic programming algorithm is based on the structure of the optimal routing strategy described in Theorem 2. Its complexity is $O(N[Q/\rho]^3)$

Author's personal copy

(see Definition 7.1 in Sipser (2013)). It requires less computations than the initial dynamic programming algorithm. For example, for $i = 1, \ldots, N-1$, the quantities $A_i(k\rho, l\rho),\ l \in \{s_1(k\rho)/\rho+2, \ldots, Q/\rho\}$ for $k \in \{0, \ldots, Q/\rho\}$ are not computed, while these quantities are computed in the initial dynamic programming algorithm. A numerical example is presented in Section 4 that shows that the difference of the computation times of these algorithms is significant especially for high values of the number of customers N.

4 Numerical Results

In the following numerical results, we implemented the initial dynamic programming algorithm and the special-purpose dynamic programming algorithm by running the corresponding Matlab programs on a personal computer equipped with an Intel Core i5-3230 M, 2.6 GHz processor and 4 GB of RAM. In Example 1, we assume that the quantities of the materials that are collected from the customers are discrete random variables and in Example 2, we assume that the quantities of the materials that are collected from the customers are continuous random variables. These examples confirm the structural results presented in Theorem 1 and in Theorem 2.

Example 1 Suppose that $N = 11$ and $Q = 15$. The travel costs between customer i and $i+1,\ i \in \{1, \ldots, 10\}$, are given by: $c_{12} = 8,\ c_{23} = 10,\ c_{34} = 9,\ c_{45} = 12,\ c_{56} = 10,\ c_{67} = 14,\ c_{78} = 12,\ c_{89} = 9,\ c_{9,10} = 11$ and $c_{10,11} = 15$. The travel costs between customers $i,\ i = 1, \ldots, 11$ and the depot are given by: $c_{10} = 12,\ c_{20} = 10,\ c_{30} = 12,\ c_{40} = 11,\ c_{50} = 9,\ c_{60} = 12,\ c_{70} = 13,\ c_{80} = 15,\ c_{90} = 12,\ c_{10,0} = 14$ and $c_{11,0} = 13$. Note that these costs satisfy the triangle inequality. We assume that the penalty costs $\pi_1, \ldots, \pi_{11}$ incurred if an item of Material 1 is loaded in Compartment 2 or if an item of Material 2 is loaded in Compartment 1 are elements of the row vector $\pi = (3, 2, 5, 4, 3, 4, 5, 6, 2, 5, 4)$. We further assume that the quantity ξ_i that is collected from each customer $i \in \{1, \ldots, 11\}$ is a discrete random variable which follows the binomial distribution $Bin(Q, 0.3)$ i.e.

$$Pr(\xi_i = x) = \binom{Q}{x} 0.3^x 0.7^{Q-x},\ x = 0, \ldots, Q.$$

We assume that the probabilities $p_1, \ldots, p_{11}$ that Material 1 will be collected from customers $1, \ldots, 11$ are elements of the row vector $p = (0.5, 0.7, 0.6, 0.8, 0.3, 0.7, 0.5, 0.9, 0.4, 0.5, 0.6)$. In Figs. 2 and 3, we present the optimal decisions for customers 3 and 9. If $0 \le z_1 \le Q,\ 0 \le z_2 \le Q$, the action of proceeding directly to next customer (Action 1) is denoted by green right-pointing triangles and the action of going to the depot for unloading and then going to the next customer (Action 2) is denoted by blue squares.

If $Q+1 \le z_1 \le 2Q,\ 0 \le z_2 \le Q$, or if $0 \le z_1 \le Q,\ Q+1 \le z_2 \le 2Q$, we use yellow diamonds for Action 3 that corresponds to quantity $C_i(z_1, z_2)$, red x-marks for Action 4 that corresponds to quantity $D_i(z_1)$ (or $D_i(z_2)$) cyan hexagrams for Action 5_θ that corresponds to quantity $E_i(z_1)$ (or $E_i(z_2)$) magenta plus signs for Action 6 that corresponds to quantity F_i and black asterisks pluses for Action 7_θ that corresponds to quantity $G_i(z_1, z_2)$.

Author's personal copy

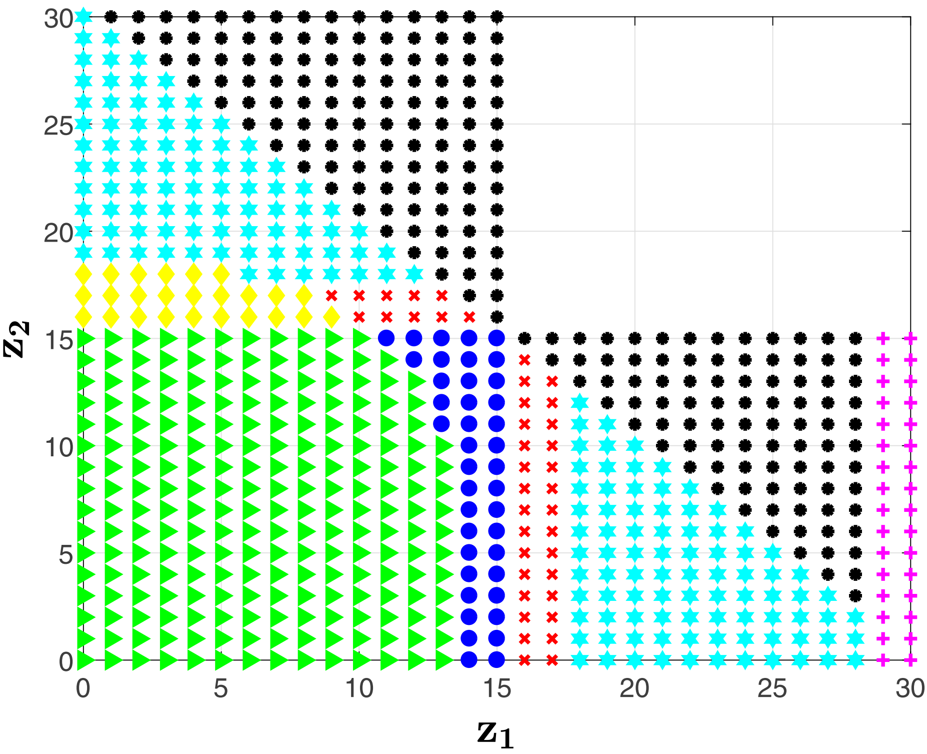

Fig. 2 The optimal decisions for customer 3

The value of the minimum expected total cost f_0 is found to be approximately equal to 161.11. The computation time of the special-purpose dynamic programming algorithm for the calculation of f_0 is approximately equal to 0.44 seconds. It is considerably smaller than the corresponding computation time of the initial dynamic programming algorithm which is approximately equal to 3 seconds.

Both algorithms enable us to determine the optimal values of θ when the optimal actions are the actions 5_θ and 7_θ. For example, for customer 9, if the state is $(z_1, z_2) = (29, 1)$, then the optimal action for the vehicle is the action 5_θ with $\theta = 4$. According to this action, the vehicle loads $\theta = 4$ items of Material 1 in Compartment 2, it goes to the depot to unload, it returns to customer 9 to load $z_1 - Q - \theta = 29 - 15 - 4 = 10$ items of Material 1 in Compartment 1 and then goes to customer 10.

If again for customer 9, the state is $(z_1, z_2) = (27, 11)$, then the optimal action for the vehicle is the action 7_θ with $\theta = 2$. According to this action, the vehicle loads $\theta = 2$ items of Material 1 in Compartment 2, it goes to the depot to unload, it returns to customer 9 to load $z_1 - Q - \theta = 27 - 15 - 2 = 10$ items of Material 1 in Compartment 1 and then proceeds to customer 10.

In Table 1 for customer 9 and for some states (z_1, z_2), for which the optimal action is the action 5_θ the optimal values of θ are presented.

Author's personal copy

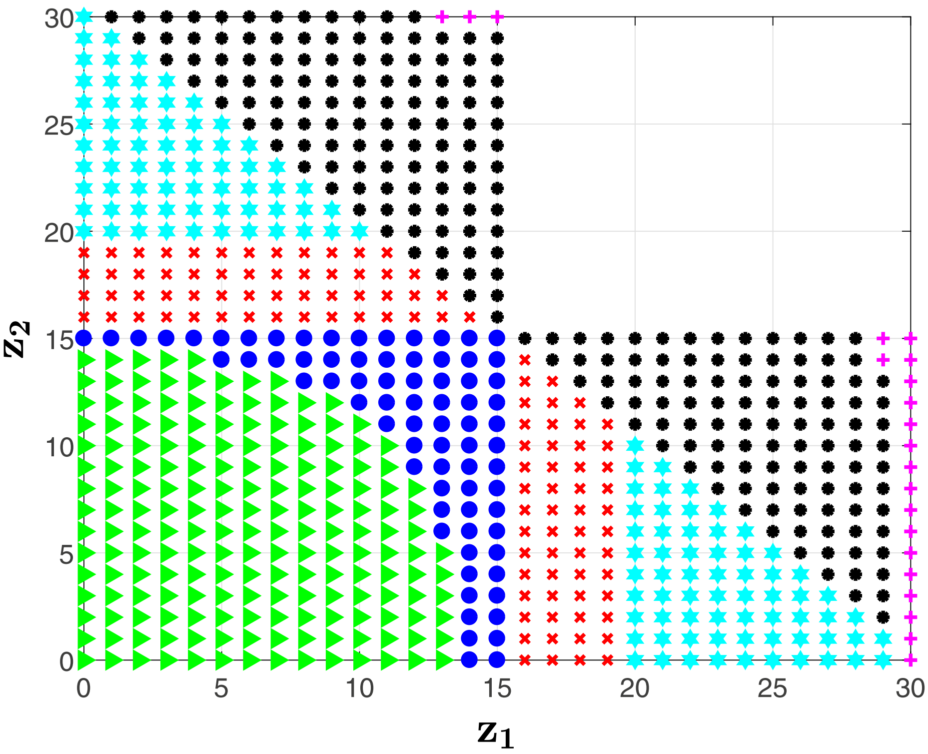

Fig. 3 The optimal decisions for customer 9

In Table 2 for customer 9 and for some states (z_1, z_2), for which the optimal action is the action 7_θ, the optimal values of θ are presented.

In Fig. 4, we present a graph that shows the variation of the minimum expected total cost f_0 as the probability p of the binomial distribution $Bin(Q, p)$ of the quantity ξ_i takes values in the set $\{0.1, 0.2, \ldots, 0.8, 0.9\}$. We see that as p takes values in the set $\{0.1, \ldots, 0.7\}$ the minimum expected total cost increases rather quickly and approximately linearly. When p takes values in the set $\{0.8, 0.9\}$ the minimum expected total cost increases rather slowly.

Table 1 The optimal values of θ for customer 9 when action 5_θ is optimal

States (z_1, z_2)	Optimal values of θ
(28,2)	3
(0,20)	0
(29,0)	4
(3,27)	1
(2,28)	2

Author's personal copy

Table 2 The optimal values of θ for customer 9 when action 7_θ is optimal

States (z_1, z_2)	Optimal values of θ
(29,3)	4
(28,10)	3
(27,13)	2
(26,11)	1
(28,15)	0

In Fig. 5, we present graphs that show, as Q varies in the set $\{10, 12, \ldots, 78, 80\}$ the variation of the required computation times (expressed in seconds) for the calculation of f_0 when the initial dynamic programming algorithm and the special-purpose dynamic programming algorithm are implemented.

We observe that, as Q increases, the computation times for both algorithms increase non-linearly. For the special-purpose algorithm the form of the graph verifies that the complexity of the algorithm is $O(NQ^3)$. The computation time required by the special-purpose algorithm is considerably smaller than the computation time required by the initial dynamic programming algorithm especially for high values of Q.

Example 2 Suppose that $N = 10$ and $Q = 8$. The travel costs between customer i and $i+1,\ i \in \{1, \ldots, 9\}$ are given by: $c_{12} = 9,\ c_{23} = 8,\ c_{34} = 9,\ c_{45} = 7,\ c_{56} = 8, c_{67} = 10,\ c_{78} = 9,\ c_{89} = 7$ and $c_{9,10} = 8$. The travel costs between customers $i,\ i = 1, \ldots, 10,$

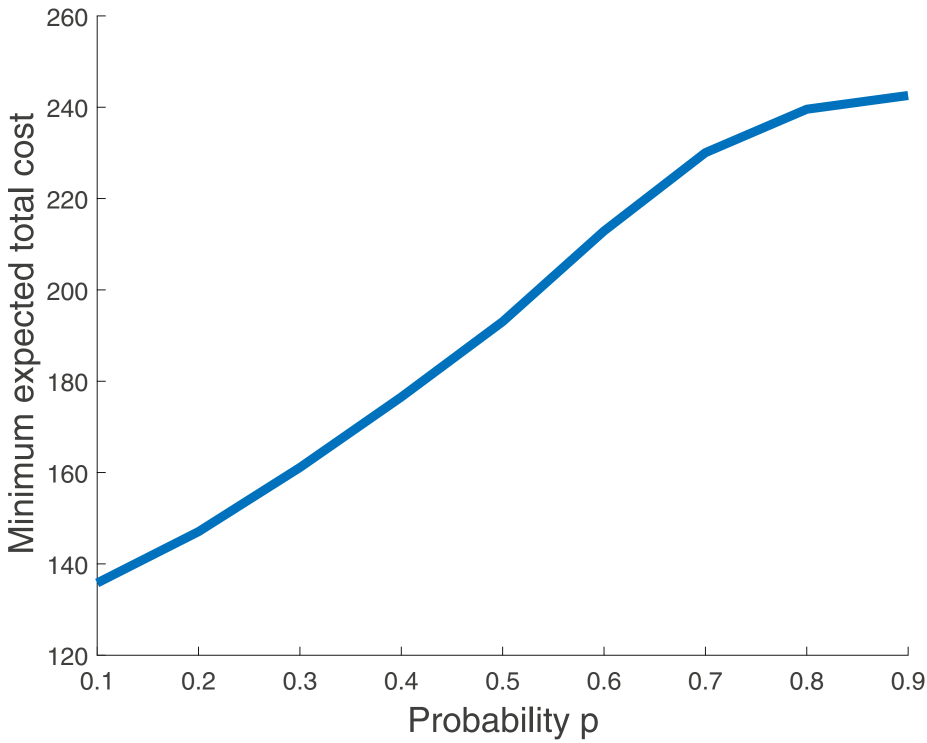

Fig. 4 The minimum total expected cost as p varies

Author's personal copy

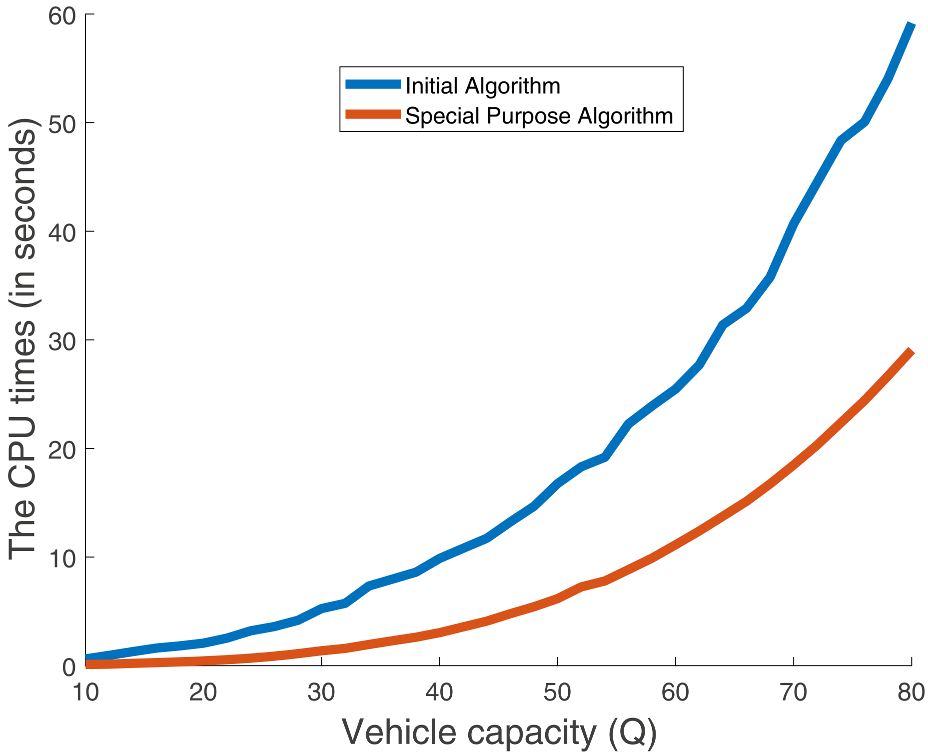

Fig. 5 The computation times of the algorithms as Q varies

and the depot are given by: $c_{10} = 7,\ c_{20} = 8,\ c_{30} = 8,\ c_{40} = 7,\ c_{50} = 6,\ c_{60} = 8,\ c_{70} = 6,\ c_{80} = 7,\ c_{90} = 7$ and $c_{10,0} = 6$.

Note that these costs satisfy the triangle inequality. We assume that the penalty costs $\pi_1, \ldots, \pi_{10}$ incurred if a unit of Material 1 is loaded in Compartment 2 or if a unit of Material 2 is loaded in Compartment 1 are elements of the row vector $\pi = (1.1, 1.3, 1.4, 1.2, 1, 1.4, 1.1, 1.2, 1.4, 1)$. We further assume that the quantity ξ_i that is collected from each customer $i \in \{1, \ldots, 10\}$ is a continuous random variable which follows the Normal distribution truncated in the interval $[0, Q]$. The probability density functions $\phi_i(x)$ are given by:

$$\phi_i(x) = [F(Q) - F(0))]^{-1} \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{(x-\mu)^2}{2\sigma^2}\right\},\ \ x \in [0, Q],$$

where, $\mu \in \mathbb{R},\ \sigma > 0$ and $F(x)$ is the cumulative distribution of the Normal distribution with parameters $\mu \in \mathbb{R}$ and $\sigma^2 > 0$. We choose $\mu = 3$ and $\sigma = 2$. We assume that the probabilities $p_1, \ldots, p_{10}$ that Material 1 will be collected from customers $1, \ldots, 10$ are elements of the row vector $p = (0.4, 0.6, 0.3, 0.5, 0.6, 0.3, 0.7, 0.8, 0.6, 0.5)$. In Figs. 6 and 7, we present the optimal decisions for customers 6 and 9. If $z_1 \in [0, Q],\ z_2 \in [0, Q]$, the action of proceeding directly to next customer (Action 1) is colored by green and

Author's personal copy

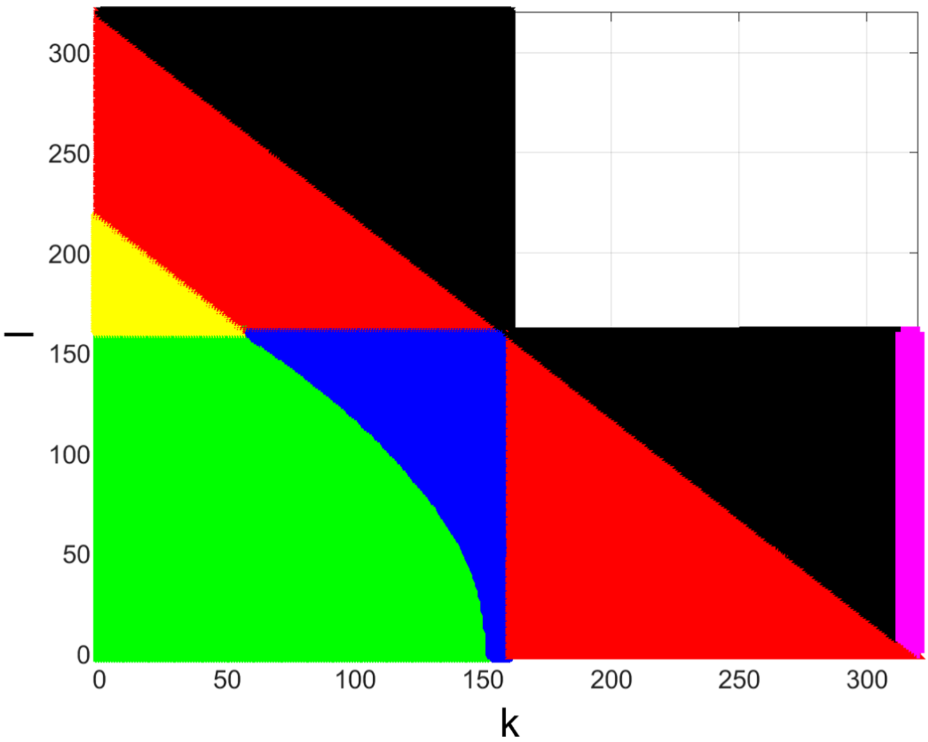

Fig. 6 The optimal decisions for customer 6

the action of going to the depot for unloading and then going to the next customer (Action 2) is colored by blue. If $z_1 \in (Q, 2Q]$, $z_2 \in [0, Q]$, or if $z_1 \in [0, Q]$, $z_2 \in (Q, 2Q]$, Action 3 that corresponds to quantity $C_i(z_1, z_2)$ is colored by yellow, Action 4 that corresponds to quantity $D_i(z_1)$ (or to quantity $D_i(z_2)$) is colored by red, Action 5_θ, $\theta \in [0, z_1 - Q)$ that corresponds to quantity $E_i(z_1)$ (or Action 5_θ, $\theta \in [0, z_2 - Q)$ that corresponds to quantity $E_i(z_2)$) is colored by cyan, Action 6 that corresponds to quantity F_i is colored by magenta and Action 7_θ that corresponds to quantity $G_i(z_1, z_2)$ is colored by black. We choose $\rho = 0.05$ so that the discretized state space $\tilde{S}$ for each customer $i \in \{1, \ldots, 10\}$ is the set:

$$\{(k \cdot 0.05, l \cdot 0.05) : k, l = 0, \ldots, 160\} \cup \{(k \cdot 0.05, l \cdot 0.05) : k = 161, \ldots, 320,\ l = 0, \ldots, 160\} \cup \{(k \cdot 0.05, l \cdot 0.05) : k = 0, \ldots, 160,\ l = 161, \ldots, 320\}.$$

The value of the minimum total expected cost f_0 is found to be approximately equal to 103.45. The computation time of the special-purpose dynamic programming algorithm for the calculation of f_0 is approximately equal to 388 seconds. It is considerably smaller than the corresponding computation time of the initial dynamic programming algorithm which is approximately equal to 852 seconds.

Both algorithms enable us to determine the optimal values of θ when the optimal actions are the actions 5_θ and 7_θ. For example, for customer 9, if the state is $(z_1, z_2) = (15.6, 0.4)$,

Author's personal copy

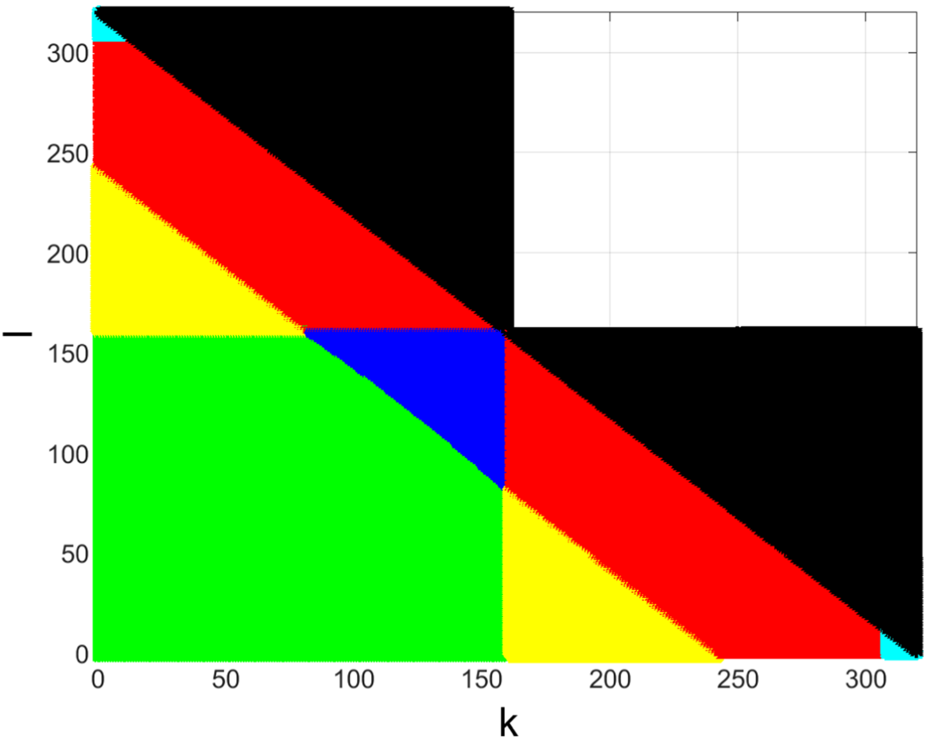

Fig. 7 The optimal decisions for customer 9

then the optimal action for the vehicle is the action 5_θ with $\theta = 0$. According to this action, the vehicle does not load any quantity of Material 1 in Compartment 2, goes to the depot to unload, returns to customer 9 to load quantity equal to $z_1 - Q - \theta = 15.6 - 8 - 0 = 7.6$ units of Material 1 in Compartment 1 and then goes to customer 10.

For customer 6, if the state is $(z_1, z_2) = (7.8, 14.8)$ then the optimal action for the vehicle is action 7_θ with $\theta = 0.15$. According to this action, the vehicle loads a quantity equal to $\theta = 0.15$ units of Material 2 in Compartment 1, goes to the depot to unload, returns to customer 6 to load a quantity equal to $z_2 - Q - \theta = 14.8 - 8 - 0.15 = 6.65$ units of Material 2 in Compartment 2 and proceeds to customer 7.

In Table 3, for customer 9 and for some states (z_1, z_2) for which the optimal action is the action 5_θ, the optimal values of θ are presented.

Table 3 The optimal values of θ for customer 9 when action 5_θ is optimal

States (z_1, z_2)	Optimal values of θ
(15.4,0.6)	0.2
(0.35,15.65)	0
(0.55,15.45)	0

Author's personal copy

Table 4 The optimal values of θ for customer 6 when action 7_θ is optimal

States (z_1, z_2)	Optimal values of θ
(7.75,13.55)	0.15
(8,15.9)	0
(7.8,10.05)	0.1
(7.9,12.5)	0
(7.85,12.15)	0.05

In Table 4, for customer 6 and for some states (z_1, z_2) for which the optimal action is the action 7_θ, the optimal values of θ are presented.

We assume that $Q = 4$ and that the number of customers N takes values in the set $\{5, 6, \ldots, 20\}$. For each value of N, let $c_{i,i+1} = 14$, $i \in \{1, \ldots, N-1\}$, $c_{i0} = 12$ if i is odd and $c_{i0} = 10$, if i is even. For each customer $i \in \{1, \ldots, N\}$, we assume that the penalty cost π_i incurred if a unit of Material 1 is loaded in Compartment 2 or if a unit of Material 2 is loaded in Compartment 1 is equal to 2 and the probability that Material 1 will be collected from each customer is equal to 0.5.

In Fig. 8, we present graphs that show, as N varies in the set $\{5, 6, \ldots, 20\}$ the variation in the computation times, expressed in seconds, required by the initial dynamic programming algorithm and by the special-purpose dynamic programming algorithm.

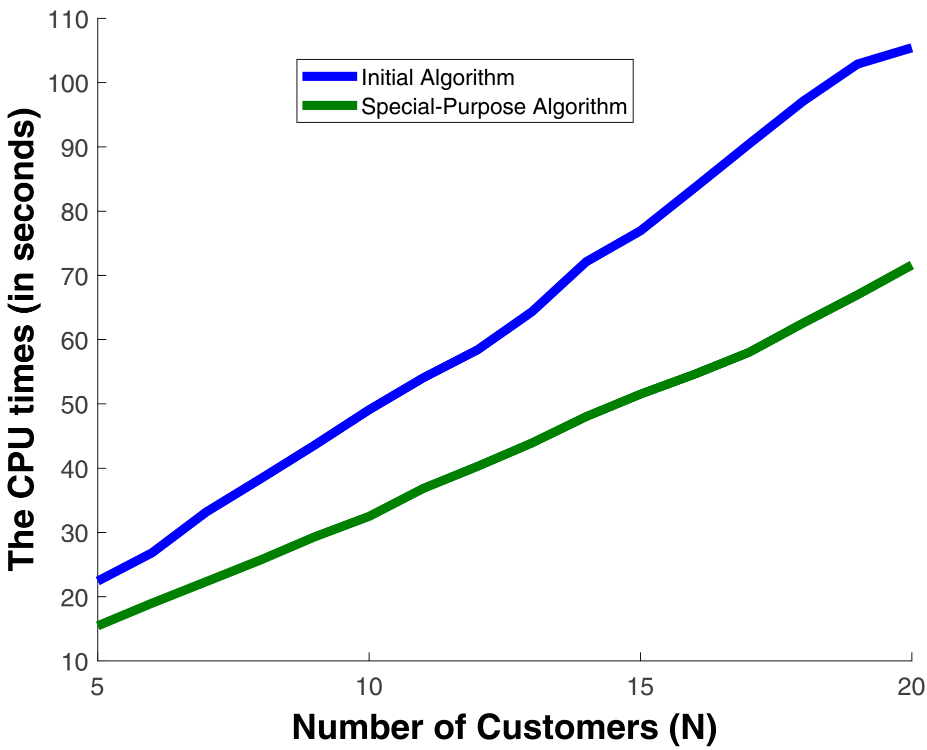

Fig. 8 The computation of the algorithms as N varies

Author's personal copy

We observe that, as N increases, the computation times for both algorithms increase approximately linearly. The form of the graph confirms that the complexity of the special-purpose algorithm ($O(N[Q/\rho]^3)$) is a linear function with respect to N. The computation time required by the special-purpose algorithm is considerably smaller than the computation time required by the initial dynamic programming algorithm for all values of N. The difference between the computation times increases as N increases.

5 The Problem when the Customers are not Ordered

We modify the problem that we introduced in Section 2 by assuming that the customers are not serviced according to a predefined sequence. In this case there are $N!$ different customer sequences that the vehicle may follow. For each sequence using the dynamic programming algorithm we can find the optimal routing strategy and the corresponding minimum total expected cost, and then by comparing these minimum costs we can determine the optimal customer sequence that achieves the overall minimum cost. Numerical experiments indicate that, if the demands of the customers are discrete random variables, it is possible to find the optimal customer sequence for values of N up to 9. As illustration we give below a numerical example.

Example 3 Suppose that $Q = 7$. We assume that the number of customers N takes values in the set $\{3, 4, \ldots, 9\}$. The travel costs c_{ij} between customers $i, j \in \{1, \ldots, 9\}$ and the travel costs c_{i0} between each customer $i \in \{1, \ldots, 9\}$ and the depot are given by the following symmetric matrix $C = (c_{ij}),\ i, j = 0, \ldots, 9$.

$$C = \begin{bmatrix} 0 & 23 & 25 & 18 & 17 & 22 & 20 & 19 & 21 & 24 \\ 23 & 0 & 18 & 16 & 15 & 16 & 17 & 15 & 13 & 17 \\ 25 & 18 & 0 & 19 & 13 & 15 & 18 & 16 & 12 & 15 \\ 18 & 16 & 19 & 0 & 14 & 16 & 12 & 15 & 13 & 17 \\ 17 & 15 & 13 & 14 & 0 & 12 & 16 & 13 & 14 & 15 \\ 22 & 16 & 15 & 16 & 12 & 0 & 18 & 16 & 14 & 15 \\ 20 & 17 & 18 & 12 & 16 & 18 & 0 & 15 & 17 & 18 \\ 19 & 15 & 16 & 15 & 13 & 16 & 15 & 0 & 15 & 16 \\ 21 & 13 & 12 & 13 & 14 & 14 & 17 & 15 & 0 & 16 \\ 24 & 17 & 15 & 17 & 15 & 15 & 18 & 16 & 16 & 0 \end{bmatrix}$$

These costs satisfy the triangle inequality. We assume that the penalty costs $\pi_1, \ldots, \pi_9$ incurred if an item of Material 1 is loaded in Compartment 2 or if an item of Material 2 is loaded in Compartment 1 are elements of the row vector $\pi = (4, 3, 2, 5, 3, 4, 1, 4, 2)$. We further assume that the quantity ξ_i that is collected from each customer $i \in \{1, \ldots, 9\}$ is a discrete random variable which follows the discrete uniform distribution, i.e. $Pr(\xi_i = x) = (Q + 1)^{-1}, \quad x = 0, \ldots, Q$ and that the probabilities $p_1, \ldots, p_9$ that Material 1 will be collected from customers $1, \ldots, 9$ are elements of the row vector $p = (0.2, 0.4, 0.1, 0.3, 0.6, 0.5, 0.8, 0.4, 0.7)$. For $N \in \{3, \ldots, 9\}$ we consider the network consisting of customers $1, \ldots, N$. In Table 5 we present for $N \in \{3, \ldots, 9\}$ the number $N!$ of all possible customer sequences, the minimum expected cost among all customer sequences, the optimal customer sequence, the required computation time in seconds (Time 1) if the initial dynamic programming algorithm is used and the required computation time in seconds (Time 2) if the special-purpose dynamic programming algorithm is used.

Author's personal copy

Table 5 The optimal customer sequence for $N = 3, 4, 5, 6, 7, 8, 9$

N	$N!$	Minimum Cost	Optimal Sequence	Time 1	Time 2
3	6	86.05	1,2,3	0.069	0.025
4	24	105.51	4,2,1,3	0.2875	0.1252
5	120	129.17	4,2,5,1,3	1.5294	1.1113
6	720	151.96	4,2,5,1,6,3	11.9384	6.0243
7	5040	172.53	4,2,5,1,7,6,3	108.2632	68.2031
8	40320	194.96	4,5,2,8,1,7,6,3	953.5481	625.9688
9	362880	220.25	4,5,9,2,8,1,7,6,3	8971.1	2117.7

In Fig. 9, we present the graphs that show, as takes values in the set $\{3, \ldots, 9\}$ the variation in required computation times, expressed in seconds, if the initial dynamic programming algorithm and if the special-purpose dynamic programming algorithm are used.

We observe that, as N increases, both computation times seem to increase exponentially. The required computation time if the special-purpose dynamic programming algorithm is used is considerably smaller than the required computation time if the initial dynamic programming algorithm is used.

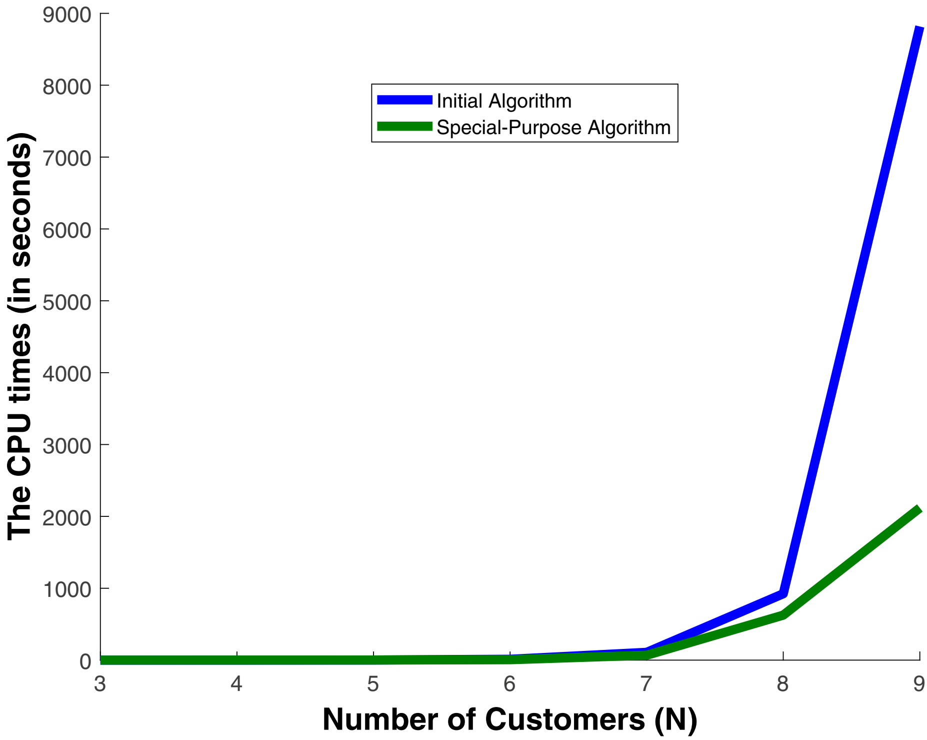

Fig. 9 The computation times of the algorithms as N varies

Author's personal copy

6 Summary of Results and a Topic for Future Research

In this paper a capacitated and compartmentalized stochastic vehicle routing problem was studied in which (i) the customers are serviced according to a particular order, (ii) the vehicle collects from the customers two similar but not identical materials, (iii) each material is loaded in the suitable compartment of the vehicle, (iv) the type of the material and the quantity that is collected from each customer are stochastic, (v) the actual material and the actual quantity that are collected from each customer become known as soon as the vehicle visits the customer. The cost structure includes travel costs between consecutive customers, travel costs between customers and the depot and penalty costs that are incurred when a material is not put in the suitable compartment of the vehicle. We choose as decision epochs for the routing of the vehicle, the epochs at which the vehicle visits for the first time each customer and the maximum possible quantity of the material that he/she possesses has been put in the suitable compartment of the vehicle. A stochastic dynamic programming algorithm is proposed for the determination of the routing strategy that minimizes the expected total costs for servicing all customers. The optimal routing strategy has a specific threshold-type structure. This result enables us to design a special-purpose dynamic programming algorithm that is considerably more efficient than the initial one. If the above Assumption (i) is not valid, it is possible to find numerically the optimal routing strategy for moderate values of the number of customers.

A possible topic for future research could be the study of a more general problem where the vehicle collects $K \geq 3$ similar but not identical materials.

Acknowledgements The authors would like to thank a reviewer for useful suggestions that improved the presentation of the paper. The author Epaminondas G. Kyriakidis has been financed by the research program EP-3042-01 (RC/AUEB).

References

Clarke G, Wright JR (1964) Scheduling of vehicle routing problem from a central depot to a number of delivery points. Oper Res 12:568–581

Dantzig G, Ramser R (1959) The truck dispatching problem. Manage Sci 6:80–91

Dimitrakos TD, Kyriakidis EG (2015) A single vehicle routing problem with pickups and deliveries, continuous random demands and predefined customer order. Eur J Oper Res 244:990–993

Elgesem AS, Skogen ES, Wang X, Fagerholt K (2018) A traveling salesman problem with pickups and deliveries and stochastic travel times: An application from chemical shipping. Eur J Oper Res 269:844–859

Gendreau M, Laporte G, Seguin R (1996) Stochastic vehicle routing. Eur J Oper Res 88:3–12

Haugland D, Ho SC, Laporte G (2007) Designing delivery districts for the vehicle routing problem with stochastic demands. Eur J Oper Res 180:997–1010

Kyriakidis EG, Dimitrakos TD (2008) Single vehicle routing problem with a predefined customer sequence and stochastic continuous demands. Math Sci 33:148–152

Kyriakidis EG, Dimitrakos TD (2013) A vehicle routing problem with a predefined customer sequence, stochastic demands and penalties for unsatisfied demands. Proceedings of 5th International Conference on Applied Operational Research Lect Notes Manage Sci 5:10–17

Kyriakidis EG, Dimitrakos TD, Karamatsoukis CC (2019) Optimal delivery of two similar products to N ordered customers with product preferences. Int J Prod Econ 209:194–204

Markov I, Bierlaire M, Cordeau JF, Maknoon Y, Varone S (2020) Waste collection inventory routing with non-stationary stochastic demands. Comput Oper Res 113:Article number 104798

Minis I, Tatarakis A (2011) Stochastic single vehicle routing problem with delivery and pickup and a predefined customer sequence. Eur J Oper Res 213:37–51

Nguyen VA, Jiang J, Ng KM, Teo KM (2016) Satisfying measure approach for vehicle routing problem with time windows under uncertainty. Eur J Oper Res 248:404–414

Author's personal copy

Pandelis DG, Kyriakidis EG, Dimitrakos TD (2012) Single vehicle routing problems with a predefined customer sequence, compartmentalized load and stochastic demands. Eur J Oper Res 217:324–332

Pandelis DG, Karamatsoukis CC, Kyriakidis EG (2013a) Single vehicle routing problems with a predefined customer order, unified load and stochastic discrete demands. Probab Eng Inf Sci 27(1):1–23

Pandelis DG, Karamatsoukis CC, Kyriakidis EG (2013b) Finite and infinite-horizon single vehicle routing problems with a predefined customer sequence and pickup and delivery. Eur J Oper Res 231:577–586

Pillac V, Gendreau M, Gueret C, Megaglia A (2013) A review of dynamic vehicle routing problems. Eur J Oper Res 225:1–11

Psaraftis HN, Wen M, Kontovas CA (2016) Dynamic vehicle routing problems: Three decades and counting. Networks 67:3–31

Ritzinger U, Puchinger J, Richard HF (2016) A survey on dynamic and stochastic vehicle routing problems. Int J Prod Res 54(1):215–231

Sipser M (2013) Introduction to the theory of computation, 3rd edn. Cengage Learning, Boston

Tatarakis A, Minis I (2009) Stochastic single vehicle routing with a predefined customer sequence and multiple depot returns. Eur J Oper Res 197:557–571

Toth P, Vigo D (eds) (2014) The vehicle routing problem. Problems, methods and applications, 2nd edn. MOS-SIAM, Philadelphia

Tsirimpas P, Tatarakis A, Minis I, Kyriakidis EG (2008) Single vehicle routing with a predefined customer sequence and multiple depot returns. Eur J Oper Res 187:483–495

Yang W-H, Mathur K, Ballou RH (2000) Stochastic vehicle routing problem with restocking. Transport Sci 34:99–112

Zhang J, Lam WHK, Chen BY (2016) On-time delivery probabilistic models for the vehicle routing problem with stochastic demands and time windows. Eur J Oper Res 249:144–154

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