International Journal of Systems Science: Operations & Logistics

ISSN: 2330-2674 (Print) 2330-2682 (Online) Journal homepage: http://www.tandfonline.com/loi/tsyb20

Stochastic single vehicle routing problem with ordered customers and partial fulfilment of demands

Epaminondas G. Kyriakidis & Theodosis D. Dimitrakos

INTERNATIONAL JOURNAL OF SYSTEMS SCIENCE: OPERATIONS & LOGISTICS, 2017
https://doi.org/10.1080/23302674.2017.1381888

Stochastic single vehicle routing problem with ordered customers and partial fulfilment of demands

Epaminondas G. Kyriakidis[a] and Theodosis D. Dimitrakos[b]

[a]Department of Statistics, Athens University of Economics and Business, Athens, Greece; [b]Department of Mathematics, University of the Aegean, Karlovassi, Samos, Greece

ABSTRACT

We consider the problem of finding the optimal routing of a single vehicle that starts its route from a depot and delivers a product to N customers that are served according to a particular order. The vehicle during its route can return to the depot for replenishment. It is assumed a stochastic demand for each customer. The actual demand of each customer becomes known upon the vehicle's arrival at the customer's site. It is permissible to satisfy fully or to satisfy partially or not to satisfy the demand of a customer. The cost structure includes travel costs between consecutive customers, travel costs between the customers and the depot and penalty costs if a customer's demand is not satisfied or if it is satisfied partially. A dynamic programming algorithm is developed for the determination of the optimal routing policy. It is shown that the optimal routing policy has a specific threshold-type structure. Furthermore, if we consider the same problem without the assumption that the customers are ordered, numerical experiments indicate that the optimal routing strategy can be computed for N smaller or equal to nine.

ARTICLE HISTORY
Received 23 January 2017
Accepted 12 September 2017

KEYWORDS
Routing problem; dynamic programming; partial service; stochastic demands

1. Introduction

One of the most widely studied problems in combinatorial optimisation is the vehicle routing problem (VRP). The objective of the VRP is to minimise the total route cost for one vehicle or for a fleet of identical vehicles that depart from one or several depots and deliver goods to N customers that are scattered in a geographical area. The vehicle may also collect expired products from the customers. The VRP can be considered as a generalisation of the classical travelling salesman problem that aims at finding an optimal route for visiting N cities and returning to the point of origin. A first version of the VRP was proposed by Dantzig, and Ramser (1959). In that paper, the authors developed a mathematical programming formulation for its solution, and described, as a realistic application, the design of optimal routing of a fleet of gasoline delivery trucks between a bulk terminal and a large number of service stations that are dispersed in a geographical area. The VRP has been extensively studied in the optimisation literature during the last 55 years. Much attention has been paid to (1) the VRP with time windows in which the delivering locations have time windows within which the deliveries of goods must be made and (2) the capacitated VRP (with or without time windows) in which the vehicles have limited carrying capacity of the goods that must be delivered. Various exact algorithms (e.g. branch-and-bound, branch-and-cut, branch-and-cut-and-price methods) that lead to the optimal routing strategy have been developed. Heuristics and metaheuristics (tabu search, simulated annealing, genetic algorithms, colony optimisation) have also been designed that lead to a 'good' solution that is not necessarily the overall optimal routing strategy. Furthermore, hybrid methods use a combination of exact algorithms, heuristics or metaheuristics to solve VRP. Surveys of results for various versions of the VRP can be found in Berhan, Beshah, Kitaw, and Abraham (2014), Eksioglu, Vural, and Reisman (2009), Kumar, and Panneerselvam (2012), Laporte (2009), Liong, Wan Rosmanira, Khairuddin, and Zirour (2008), Pillac, Gendreau, Gueret, and Megaglia (2013), Psaraftis, Wen, and Kontovas (2016), Simchi-Levi, Chen, and Bramel (2005) and Toth and Vigo (2002).

In the last 17 years, some capacitated VRPs have been studied in which a single vehicle starts its route from a depot and serves N customers according to a predefined order. This means that customer 1 is served first, then customer 2 is served, then customer 3 is served and so on. We refer to the papers by Dikas, Minis, and Mamasis (2016), Dimitrakos, and Kyriakidis (2015), Kyriakidis, and Dimitrakos (2008), Minis, and Tatarakis (2011), Pandelis, Karamatsoukis, and Kyriakidis (2013a, 2013b),

CONTACT Theodosis D. Dimitrakos dimitheo@aegean.gr

Pandelis, Kyriakidis, and Dimitrakos (2012), Tatarakis, and Minis (2009), Tsirimpas, Tatarakis, Minis, and Kyriakidis (2008), Yang, Mathur, and Ballou (2000) and Zhang, Lam, and Chen (2016). Suitable dynamic programming algorithms have been proposed for these problems. It was shown that the structure of the optimal routing strategy is of threshold-type. In all these problems, it was assumed that the demands of the customers must be satisfied completely. In the present paper, we relax this assumption by allowing the possibility to satisfy partially or not to satisfy the demands of the customers. If the demand of a customer is satisfied partially, a penalty cost is incurred. Specifically, we assume that a vehicle starts its route from a depot loaded with items of a product to its full capacity and visits N customers according to a predefined sequence $1 \to 2 \to \cdots \to N$. The demands of the customers for the product are random variables with known distributions. The actual demand of a customer becomes known only when the vehicle arrives at his/her site. The vehicle may satisfy fully the demand of a customer or satisfy some part of the demand or not satisfy the demand at all. The vehicle may interrupt its route by going to the depot for replenishment. The total cost consists of (1) travel costs between consecutive customers, (2) travel costs between customers and the depot and (3) penalty costs due to unsatisfied demands. A dynamic programming algorithm is constructed for the determination of the optimal routing strategy of the vehicle. It is shown that the optimal routing strategy is characterised for each customer by two critical numbers. This characterisation enables us to design an efficient special-purpose dynamic programming algorithm that operates over the routing strategies having this structure.

We give two realistic examples that fit to the problem that we study. The first example is the so-called ex-van sales. In ex-van sales, the driver acts as a salesman. Suppose that his customers are groceries that are supplied with a particular kind of milk according to a particular order. It is plausible to assume that the number of bottles of milk that each grocery will demand is not known in advance but it will be revealed as soon as the vehicle arrives at a grocery. Therefore, in this example the demands of the customers can be considered to be discrete random variables. A second practical example is the delivery of petrol to petrol stations in a geographical area. The petrol stations are serviced according to a particular order. It is reasonable to assume that the demand for petrol of each petrol station is stochastic, since, when the order is issued it is unknown how much petrol will be sold during the time between the order and the delivery. Therefore in this case, the demands of the customers can be considered to be continuous random variables.

Note that the problem studied in the present paper can be considered as a generalisation of the problem studied in Kyriakidis, and Dimitrakos (2013). In that work it was assumed that the vehicle visits each customer and satisfies as much demand as possible and then an action is selected that depends on the number of the remaining items of the product carried by the vehicle. A penalty cost was imposed if the demand of the customer was satisfied partially. In the present work, the action depends on the load of the vehicle when it arrives at a customer's site and the actual demand of the customer. Note also that there are studies (see e.g. pp. 179–180 in Bhusiri, Qureshi, and Taniguchi (2014); Toth and Vigo (2002)) where penalty costs are included in the cost structure in VRPs with time windows when the deliveries of goods are materialised before or after the time windows.

The rest of the paper is organised as follows. In Section 2, the problem is specified and analysed for the case of discrete stochastic demands of the customers. A dynamic programming approach is proposed for the determination of the optimal routing strategy. The structure of the optimal routing strategy is proved and it is used for the design of an efficient special-purpose dynamic programming algorithm. In Section 3, analogous results are presented for the case of continuous stochastic demands of the customers. In Section 4, our theoretical results are illustrated by numerical examples. In Section 5, we consider the same problem without the assumption that the customers are ordered. A summary of results and topics for future research are presented in Section 6. Note that some results of the present paper without mathematical details for the case of discrete demands were presented in the operations research conference OR2017 (see Kyriakidis, and Dimitrakos (2017)).

2. The problem when the demands are discrete random variables

We assume that a vehicle of capacity Q starts its route from a depot loaded with Q items of a product and visits N customers according to a predefined order $1 \to 2 \to \cdots \to N$. The demand of customer $j \in \{1, \ldots, N\}$ for a product is a discrete random variable $\xi_j \in \{0, \ldots, Q\}$ with known distribution. The actual demand of each customer becomes known only when the vehicle visits the customer's site. Let c_{j0} and c_{0j}, $j = 1, \ldots, N$, be the travel cost from customer j to the depot and the travel cost from the depot to customer j, respectively. Let also $c_{j,j+1}$, $j = 1, \ldots, N-1$, be the travel cost from customer j to customer $j+1$. These costs can be considered as the costs of the required driver's labour and of the gasoline that the vehicle needs to cover the distances between

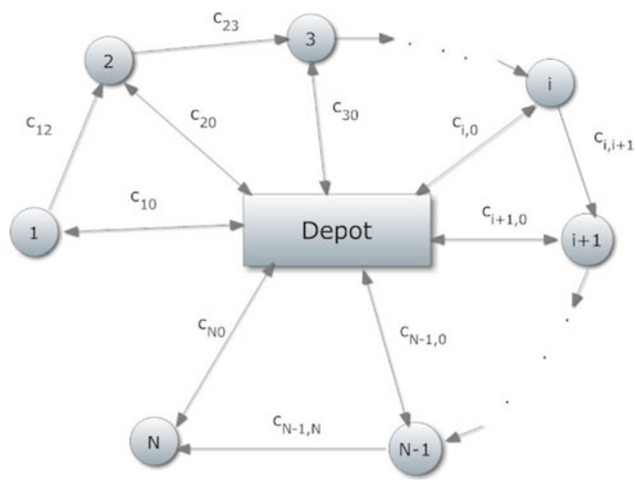

Figure 1. The road network for the problem.

consecutive customers and the distances between customers and the depot. It is plausible to assume that these costs satisfy the symmetric property and the triangle inequality, i.e. $c_{j0} = c_{0j}$, $j = 1, \ldots, N$, and $c_{j,j+1} \le c_{j0} + c_{0,j+1}$, $j = 1, \ldots, N-1$. The road network is depicted in Figure 1.

We summarise the parameters of the model in Table 1.

Suppose that the vehicle arrives at customer's $j \in \{1, \ldots, N-1\}$ site loaded with $z \in \{0, \ldots, Q\}$ items of the product and suppose that the actual demand of customer is equal to $s \in \{0, \ldots, Q\}$. The variables of the model are presented in Table 2.

If $z \ge s$, the possible actions are Action $1_\theta, \theta \in \{0, \ldots, s\}$, and Action 2. Action 1_θ means that the vehicle delivers θ items of the product to customer j and proceeds to customer $j+1$. Action 2 means that the vehicle delivers s items of the product to customer j, it goes to the depot, restocks with load Q and then visits next customer $j+1$. If $z < s$ the possible actions are $3_\theta(\theta \in \{0, \ldots, z\}), 4, 5_\theta(\theta \in \{1, \ldots, s-z\})$ and 6. Action 3_θ means that the vehicle delivers θ items of the product to customer j and then proceeds to customer $j+1$. Action 4 means that the vehicle delivers z items of the product to customer j, it goes to the depot, restocks with load Q and then visits next customer $j+1$. Action 5_θ means that the vehicle delivers z items of the product to customer j, it goes to the depot, restocks with load Q, it returns to customer j to deliver $\theta \in \{1, \ldots, s-z\}$ owed items and then proceeds to customer $j+1$. Action 6 means that the vehicle delivers z items of the product to customer j, it goes to the depot to restock with the owed $s-z$ items of the product, it returns to customer j to deliver the owed $s-z$ items of the product, it makes a second trip to the depot to restock with Q items of the product and then goes to next customer $j+1$. Note that it is assumed that, if Action 5_θ or Action 6 is selected, there is no extra demand when the vehicle returns to customer j, i.e. ξ_j remains unaltered.

Table 1. Model parameters.

N	Total number of customers
Q	Capacity of the vehicle
c_{j0}	Travel cost from customer j to the depot
c_{0j}	Travel cost from the depot to customer j
$c_{j,j+1}$	Travel cost from customer j to customer $j+1$

Table 2. Model variables.

j	Customer's index, $j \in \{1, \ldots, N\}$
ξ_j	The demand of customer j
z	The product load of the vehicle, $z \in \{0, \ldots, Q\}$
s	The actual demand of a customer, $s \in \{0, \ldots, Q\}$

Suppose that the vehicle arrives at customer's N site and its load is greater or equal to the actual demand of customer N. In this case, it satisfies fully the demand and terminates its route by returning to the depot. If the load of the vehicle is less than the actual demand of customer N, the possible actions are Action 7 and Action 8. Action 7 means that the vehicle delivers its load to customer N and terminates its route by returning to the depot. Action 8 means that the vehicle delivers its load to customer N, it goes to the depot to restock with the owed quantity, it returns to customer N to deliver the owed quantity and then it terminates its route by returning to the depot.

Note that, when Actions $1_\theta(\theta \in \{0, \ldots, s-1\}), 3_\theta(\theta \in \{0, \ldots, z\}), 4, 5_\theta(\theta \in \{1, \ldots, s-z-1\}), 7$ are selected then some part or the whole of the demand of customer j is not satisfied. In this case, a penalty cost is incurred that is equal to π_j per item that is not delivered. In Figure 2, a flowchart is provided with all possible actions for the routing of the vehicle in each case.

Our goal is to determine the optimal routing strategy of the vehicle that serves all customers fully or partially. This routing strategy minimises the expected total cost from the beginning of the route until its end. The total cost consists of travel costs and penalty costs. Note that a practical example with discrete demands of the customers is the delivery of bottles of milk to a sequence of customers (see Section 1).

Let $f_j(z,s), z, s = 0, \ldots, Q$, denote the minimum expected future cost if the number of items carried by the vehicle when it arrives at customer's $j \in \{1, \ldots, N\}$ site is equal to z and s is the number of items that customer j demands. For $j \in \{1, \ldots, N-1\}$, this quantity satisfies the following dynamic programming Equations (1) and (4) (see Chapter I in Ross (1983)).

If $z \ge s$ then

$$f_j(z,s) = \min\{H_j(z,s), R_j\}, \qquad (1)$$

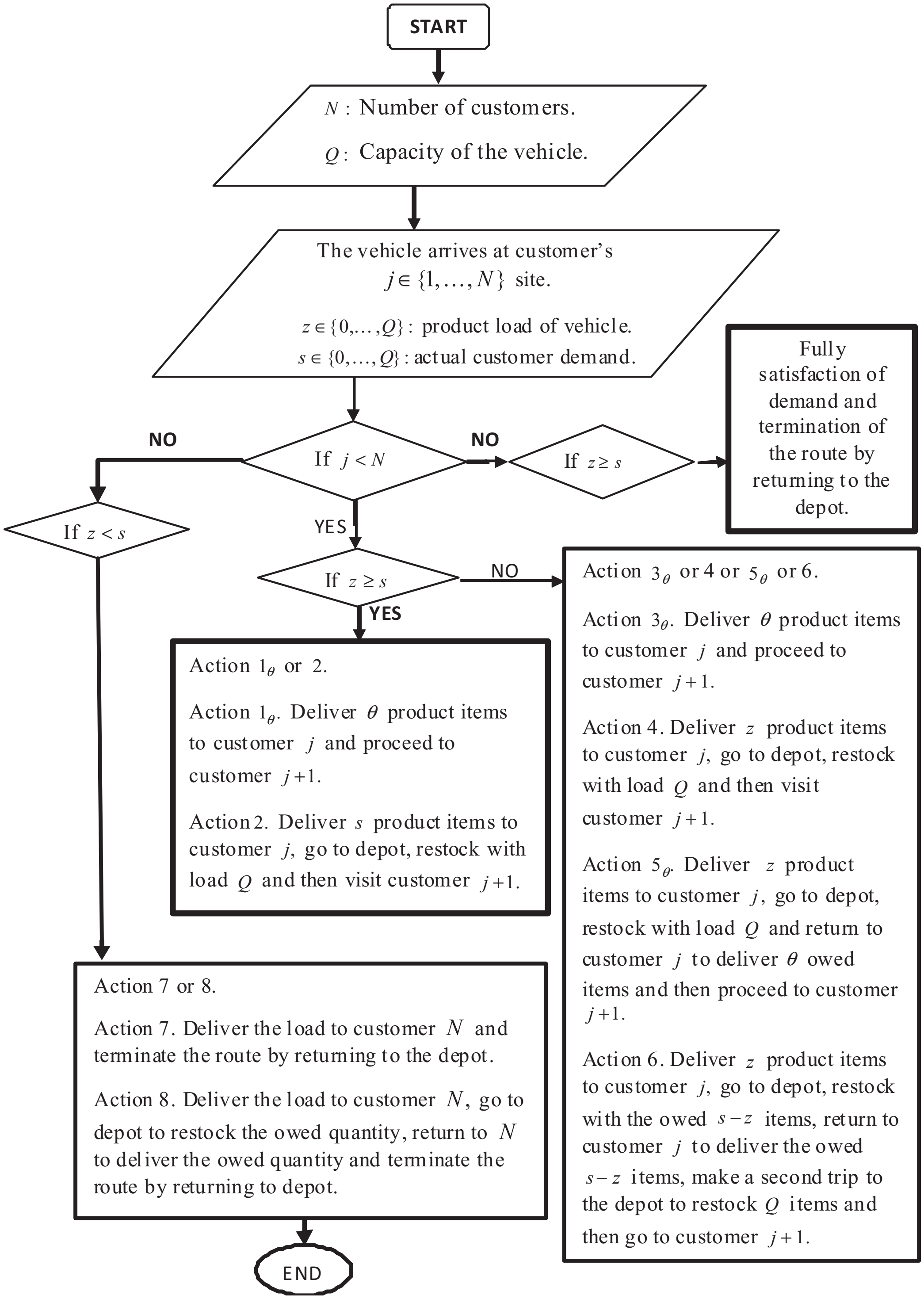

Figure 2. A flowchart for all possible actions.

where

$$H_j(z,s) = c_{j,j+1} + \min_{\theta\in\{0,\dots,s\}}\{\pi_j(s-\theta) + E[f_{j+1}(z-\theta,\xi_{j+1})]\}, \qquad (2)$$

and

$$R_j = c_{0j} + c_{j+1,0} + E[f_{j+1}(Q,\xi_{j+1})]. \qquad (3)$$

If $z < s$ then

$$f_j(z,s) = \min\{A_j(z,s), B_j(z,s), C_j(z,s), D_j\}, \qquad (4)$$

where

$$A_j(z,s) = c_{j,j+1} + \min_{\theta\in\{0,\dots,z\}}\{\pi_j(s-\theta) + E[f_{j+1}(z-\theta,\xi_{j+1})]\}, \quad (5)$$

$$B_j(z,s) = c_{0j} + c_{0,j+1} + \pi_j(s-z) + E[f_{j+1}(Q,\xi_{j+1})], \quad (6)$$

$$C_j(z,s) = 2c_{0,j} + c_{j,j+1} + \min_{\theta\in\{1,\dots,s-z\}}\{\pi_j(s-z-\theta) + E[f_{j+1}(Q-\theta,\xi_{j+1})]\}, \quad (7)$$

$$D_j = 3c_{0j} + c_{0,j+1} + E[f_{j+1}(Q,\xi_{j+1})]. \quad (8)$$

The boundary conditions are

$$f_N(z,s) = c_{0N},\ z \ge s, \quad (9)$$

and

$$f_N(z,s) = c_{0N} + \min\{F_N(z,s), G_N\},\ \ z < s, \quad (10)$$

where

$$F_N(z,s) = \pi_N(s-z),$$

$$G_N = 2c_{N0}.$$

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

$$f_0 = c_{01} + E[f_1(Q,\xi_1)].$$

In the above equations, the expected values $E[f_j(z,s)]$, $j = 1,\dots,N$, $z \in \{0,\dots,Q\}$, $s \in \{0,\dots,Q\}$, are taken with respect to the random variables ξ_j, $j = 1,\dots,N$. Since the demand of each customer $j \in \{1,\dots,N\}$ for the product is a discrete random variable $\xi_j \in \{0,\dots,Q\}$ with known distribution, the expected values $E[f_{j+1}(z,\xi_{j+1})]$ are computed in terms of sums. For example, the quantity $A_j(z,s)$, $z < s$, in Equation (5), is computed as follows:

$$A_j(z,s) = c_{j,j+1} + \min_{\theta\in\{0,\dots,z\}}\left\{\pi_j(s-\theta) + \sum_{x=0}^{Q} f_{j+1}(z-\theta,x)P(\xi_{j+1}=x)\right\},$$

where $P(\xi_{j+1}=x)$ is the probability mass function of the demand ξ_{j+1} of customer $j+1$. The terms $H_j(z,s)$ and R_j in the right-hand-side of Equation (1) correspond to Actions $1_\theta(\theta \in \{0,\dots,s\})$ and to Action 2, respectively. The terms $A_j(z,s), B_j(z,s), C_j(z,s), D_j$ in the right-hand-side of Equation (4) correspond to Actions $3_\theta(\theta \in \{0,\dots,z\})$, Action 4, Action $5_\theta(\theta \in \{1,\dots,s-z\})$and Action 6, respectively. The terms $F_N(z,s)$ and G_N in the right-hand-side of Equation (10) correspond to Action 7 and Action 8, respectively. Lemma 2.1 and Lemma 2.2 below will be used in the proof of Theorem 2.1 that describes the structure of the optimal routing strategy. The proofs of Lemmas 1, 2 and the proof of Theorem 2.1 are given in the Appendix.

Lemma 2.1: *For $j = 1,\dots,N$, for fixed $s \in \{0,\dots,Q\}$, $f_j(z,s)$ is non-increasing with respect to $z \in \{0,\dots,Q\}$.*

Lemma 2.2: *For $j = 1,\dots,N-1$, for fixed $z \in \{0,\dots,Q\}$, $H_j(z,s)$ is non-decreasing in $s \in \{0,\dots,z\}$ and $A_j(z,s), B_j(z,s), C_j(z,s)$ are non-decreasing in $s \in \{z+1,\dots,Q\}$.*

Theorem 2.1: *For each customer $j \in \{1,\dots,N-1\}$ and each demand $s \in \{0,\dots,Q\}$, there exist two critical integers $h_1(j,s), h_2(j,s)$ $(h_2(j,s) < s \le h_1(j,s))$ such that it is optimal to select (1) Action 1_θ for some $\theta \in \{0,\dots,s\}$ if $z \in \{h_1(j,s),\dots,Q\}$, (2) Action 2 if $z \in \{s,\dots,h_1(j,s)-1\}$, (3) Action 3_θ for some $\theta \in \{0,\dots,z\}$ or Action 4 or Action 5_θ for some $\theta \in \{1,\dots,s-z\}$ if $z \in \{h_2(j,s)+1,\dots,s-1\}$ and (4) Action 6 if $z \in \{0,\dots,h_2(j,s)\}$. The critical integers $h_1(j,s), h_2(j,s)$ are non-decreasing in s.*

Remark 2.1: Suppose that we restrict the action set when $z < s$ by assuming that the actions $3_\theta, \theta \in \{0,\dots,z-1\}$, are not possible. Then

$$A_j(z,s) = c_{j,j+1} + \pi_j(s-z) + E[f_{j+1}(0,\xi_{j+1})]. \quad (11)$$

Suppose that

$$c_{j,j+1} + E[f_{j+1}(0,\xi_{j+1})] \le c_{0j} + E[f_{j+1}(Q,\xi_{j+1})] \quad (12)$$

From (6), (11) and (12) we deduce that $A_j(z,s) \le B_j(z,s), 0 \le z \le s-1$. A consequence of this inequality and the fact that $A_j(z,s), A_j(z,s) - C_j(z,s)$ are non-increasing in z is the existence of a third critical number $\tilde{h}_2(j,s)(h_2(j,s) \le \tilde{h}_2(j,s) < s)$ such that it is optimal to select Action 3_z if $\tilde{h}_2(j,s) \le z < s$, and Action 3_θ, for some $\theta \in \{0,\dots,z\}$, if $h_2(j,s) < z < \tilde{h}_2(j,s)$. If (12) does not hold, then the optimal action i $\tilde{h}_2(j,s) \le z < s$ is Action 4 instead of Action 3_z.

In view of Theorem 2.1, the optimal policy, i.e. the critical numbers $h_1(j,s), h_2(j,s)$ for $j = 1,\dots,N-1$ and $s \in \{0,\dots,Q\}$, can be found by the following special-purpose dynamic programming algorithm:

Algorithm for the determination of the critical integers $h_1(j,s)$,$h_2(j,s)$, $s \in \{0,\dots,Q\}$ for customers $j = 1,\dots,N-1$

Step 0. Set $f_N(z,s) = c_{0N}$ for $z, s \in \{0,\dots,Q\}$ such that $z \geq s$ and
$f_N(z,s) = c_{0N} + \min\{2c_{N0}, \pi_N(s-z)\}$ for $z, s \in \{0,\dots,Q\}$ such that $z < s$.
Set $j = N-1$.

Step 1. (Determination of critical integers $h_1(j,s)$, $0 \leq s \leq Q$)
Compute R_j from (3).
For $s = 0,\dots,Q$ do the following:
For $z = Q, Q-1,\dots$ compute $H_j(z,s)$ until $H_j(z,s) > R_j$ or $z = s$.
If $z > s$ set $h_1(j,s) = z+1$.
If $z = s$ and $H_j(s,s) > R_j$ set $h_1(j,s) = z+1$.
If $z = s$ and $H_j(s,s) \leq R_j$ set $h_1(j,s) = z$.
Set $f_j(z,s) = H_j(z,s)$, $h_1(j,s) \leq z \leq Q$ and $f_j(z,s) = R_j$, $s \leq z < h_1(j,s)$.

Step 2. (Determination of critical integers $h_2(j,s)$, $0 \leq s \leq Q$)
Compute D_j from (8).
For $s = 0,\dots,Q$ do the following:
For $z = s-1, s-2,\dots$ compute $A_j(z,s), B_j(z,s), C_j(z,s)$ until $A_j(z,s), B_j(z,s), C_j(z,s) \geq D_j$ or $z = 0$.
If $z > 0$ set $h_2(j,s) = z$.
If $z = 0$ and $A_j(0,s), B_j(0,s), C_j(0,s) \geq D_j$ set $h_2(j,s) = 0$; otherwise, set $h_2(j,s) = -1$.
Set $f_j(z,s) = \min\{A_j(z,s), B_j(z,s), C_j(z,s), D_j\}$, $h_2(j,s) < z < s$ and $f_j(z,s) = D_j$, $0 \leq z \leq h_2(j,s)$.

Step 3. Set $j = j-1$. If $j \geq 1$ go to Step 1. Otherwise stop.

3. The problem when the demands are continuous random variables

We modify the problem that we introduced in Section 2 by assuming that the demands ξ_j, $j = 1,\dots,N$, of the customers are continuous random variables and take values in the interval $[0,Q]$ with probability density function $\phi_j(x)$. A practical example with continuous demands could be the delivery of petrol to a sequence of petrol stations that we described in Section 1. Let $z, s \in [0,Q]$ be the quantity of the product carried by the vehicle when it arrives at customer's $j \in \{1,\dots,N\}$ site and the quantity of the product that customer j demands, respectively. The Actions $1_\theta (0 \leq \theta \leq s)$, 2 when $z \geq s$ and the Actions $3_\theta (0 < \theta \leq z)$, 4, $5_\theta (0 \leq \theta \leq s-z)$, 6 when $z < s$ are the same as those defined in Section 2. The minimum expected future cost $f_j(z,s)$, $z, s \in [0,Q]$, for $j = 1,\dots,N$, satisfies the dynamic programming Equations (1) and (4) and the boundary conditions (9) and (10). Since, in this case, the demand of each customer $j \in \{1,\dots,N\}$ for the product is a continuous random variable $\xi_j \in [0,Q]$ with known distribution, the expected values $E[f_{j+1}(z,\xi_{j+1})]$ are computed in terms of integrals. For example, the quantity $B_j(z,s)$, $z < s$, in Equation (6), is computed, in this case, as follows:

$$B_j(z,s) = c_{0j} + c_{0,j+1} + \pi_j(s-z) + \int_0^Q f_{j+1}(Q,x)\phi_{j+1}(x)dx,$$

where $\phi_{j+1}(x)$ is the probability density function of the demand ξ_{j+1} of customer $j+1$. The structure of the optimal routing strategy is the same as in the case of discrete demands and is given in the theorem below.

Theorem 3.1: *For each customer $j \in \{1,\dots,N-1\}$ and each demand $s \in [0,Q]$ there exist two critical numbers $h_1(j,s), h_2(j,s)$ $(h_2(j,s) < s \leq h_1(j,s))$ such that it is optimal to select (1) Action 1_θ for some $\theta \in [0,s]$ if $z \in [h_1(j,s),Q]$, (2) Action 2 if $z \in [s,h_1(j,s))$, (3) Action 3_θ for some $\theta \in [0,z]$ or Action 4 or Action 5_θ for some $\theta \in (0,s-z]$ if $z \in (h_2(j,s),s)$ and (4) Action 6 if $z \in [0,h_2(j,s)]$. The critical numbers $h_1(j,s), h_2(j,s)$ are non-decreasing in s.*

The state space after the first visit of the vehicle to customer $j \in \{1,\dots,N\}$ is the set $S = \{(z,s) : 0 \leq z, s \leq Q\}$. A discretisation of the state space is necessary for the implementation of the dynamic programming algorithm. Let ρ be a relatively small number (e.g. $\rho = 0.05$ or $\rho = 0.01$). We discretise S by restricting our attention only to its points $(k\rho, l\rho)$, $k, l = 0,\dots,Q/\rho-1, Q/\rho$. The minimum expected cost $f_N(k\rho, l\rho)$, $k, l = 0,\dots,Q/\rho$ is found by using (9) and (10) with $z = k\rho$, $s = l\rho$. The minimum expected cost $f_j(k\rho, l\rho)$, $k, l = 0,\dots,Q/\rho$, and the corresponding optimal decision are found, recursively, for $j = N-1, N-2,\dots,1$ by using the dynamic programming Equations (1) and (4) with $z = k\rho$ and $s = l\rho$. The parameter θ in (2), (5) and (7) takes values in the sets $\{u\rho : u = 0,\dots,l\}$, $\{u\rho : u = 0,\dots,k\}$, $\{u\rho : u = 1,\dots,l-k\}$, respectively. The expectations in (2), (3), (5), (6), (7) and (8) are computed approximately. For example, the quantity $C_j(k\rho, l\rho)$, $k < l$, is computed as follows:

$$C_j(k\rho, l\rho) = 2c_{0j} + c_{j,j+1} + \min_{\theta \in \{\rho, 2\rho, \dots, (l-k)\rho\}} \times \left\{ \pi_j(l\rho - k\rho - \theta) + \sum_{x=0}^{Q/\rho - 1} f_{j+1}(Q - \theta, x\rho)\phi_{j+1}(x\rho)\rho \right\}.$$

In view of Theorem 2.1, the optimal policy, i.e. the critical numbers $h_1(j, l\rho), h_2(j, l\rho),\ l = 0, \ldots, Q/\rho$, can be found by the following special-purpose dynamic programming algorithm:

Algorithm for the determination of the critical numbers $h_1(j, l\rho), h_2(j, l\rho), l = 0, \ldots, Q/\rho$, for customers $j = 1, \ldots, N-1$

Step 0. Set $f_N(k\rho, l\rho) = c_{0N}$ for $k, l \in \{0, \ldots, Q/\rho\}$ such that $k \geq l$ and
$f_N(k\rho, l\rho) = c_{0N} + \min\{2c_{N0}, \pi_N(k\rho - l\rho)\}$ for $k, l \in \{0, \ldots, Q/\rho\}$ such that $k < l$.
Set $j = N - 1$.

Step 1. (Determination of critical numbers $h_1(j, l\rho), l = 0, \ldots, Q/\rho$)
Compute R_j from (3).
For $l = 0, \ldots, Q/\rho$ do the following:
For $k = Q/\rho, Q/\rho - 1, \ldots$ compute $H_j(k\rho, l\rho)$ until $H_j(k\rho, l\rho) > R_j$ or $k = l$.
If $k > l$ set $h_1(j, l\rho) = k\rho + \rho$.
If $k = l$ and $H_j(l\rho, l\rho) > R_j$ set $h_1(j, l\rho) = k\rho + \rho$.
If $k = l$ and $H_j(l\rho, l\rho) \leq R_j$ set $h_1(j, l\rho) = k\rho$.
Set $f_j(k\rho, l\rho) = H_j(k\rho, l\rho), h_1(j, l\rho)/\rho \leq k \leq Q/\rho$
and $f_j(k\rho, l\rho) = R_j, l \leq k < h_1(j, l\rho)/\rho$.

Step 2. (Determination of critical numbers $h_2(j, l\rho), 0 \leq l \leq Q/\rho$)
Compute D_j from (8).
For $l = 0, \ldots, Q/\rho$ do the following:
For $k = l - 1, l - 2, \ldots$ compute $A_j(k\rho, l\rho), B_j(k\rho, l\rho), C_j(k\rho, l\rho)$ until $A_j(k\rho, l\rho), B_j(k\rho, l\rho), C_j(k\rho, l\rho) \geq D_j$ or $k = 0$.
If $k > 0$ set $h_2(j, l\rho) = k\rho$.
If $k = 0$ and $A_j(0, l\rho), B_j(0, l\rho), C_j(0, l\rho) \geq D_j$ set $h_2(j, l\rho) = 0$; otherwise,
set $h_2(j, l\rho) = -\rho$.
Set $f_j(k\rho, l\rho) = \min\{A_j(k\rho, l\rho), B_j(k\rho, l\rho), C_j(k\rho, l\rho), D_j\}, h_2(j, l\rho)/\rho < k < l$
and $f_j(k\rho, l\rho) = D_j, 0 \leq k \leq h_2(j, l\rho)/\rho$.

Step 3. Set $j = j - 1$. If $j \geq 1$ go to Step 1. Otherwise stop.

4. Numerical examples

In the following numerical examples, we implemented the initial dynamic programming algorithm and the special-purpose dynamic programming algorithm by running the corresponding Matlab program on a personal computer equipped with an Intel Core i5-3230 M, 2.6 GHz processor and 4 GB of RAM. In Examples 1–5, the demands of the customers for the product are discrete random variables and in Examples 6–8 the demands of the customers for the product are continuous random variables.

Table 3. The values of model parameters.

Parameters	Values
N	12
Q	20
$c_{j,j+1},\ j \in \{1, \ldots, 11\}$	10, 12, 10, 14, 12, 15, 12, 13, 10, 9, 12
$c_{j0},\ j \in \{1, \ldots, 12\}$	12, 11, 9, 10, 13, 11, 14, 10, 12, 13, 10, 13
$\pi_j,\ j \in \{1, \ldots, 12\}$	2.5, for each customer $j \in \{1, \ldots, 12\}$

Example 4.1: We suppose that the parameters of the model take the values given in Table 3.

Note that the travel costs $c_{j,j+1},\ j = 1, \ldots, 11$, from customer j to customer $j+1$ and the travel costs c_{j0} from customer j to the depot satisfy the triangle inequality. Suppose that for each customer $j \in \{1, \ldots, 12\}$, the demand ξ_j for the product follows the binomial distribution $\mathrm{Bin}(Q,\ p)$, i.e.

$$\Pr(\xi_j = x) = \binom{Q}{x} p^x (1-p)^{Q-x}, x = 0, \ldots, Q.$$

Note that the binomial distribution had been used to model demands in VRPs. For example, we refer to the papers of Golden, and Yee (1979) and Haugland, Ho, and Laporte (2007). For $p = 0.35$, in Figures 3 and 4 below, we present the optimal decisions for customers 3 and 7. It can be seen from Figures 3 and 4 that the structure of the optimal routing strategy is, as expected, of threshold-type described in Theorem 2.1. For example, $h_1(3,\ 10) = 16$, $h_2(3,\ 17) = 2$, $h_3(7,\ 6) = 10$, $h_2(7,\ 18) = 1$.

Note that in Figures 3 and 4, action 3_θ does not appear as optimal action. We point out that in this numerical example action 3_θ does not appear as optimal action for any customer $j = 1, \ldots, 11$. This is due to the cost structure chosen in this numerical example. The value

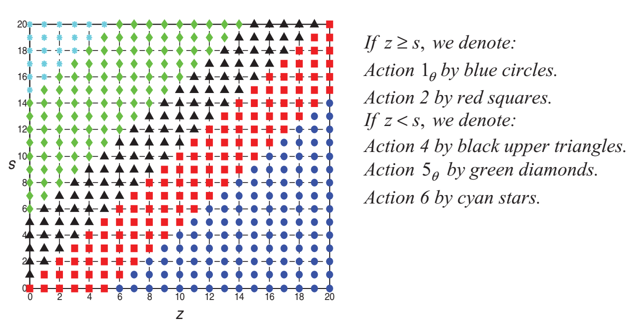

Figure 3. The optimal decisions for customer 3.

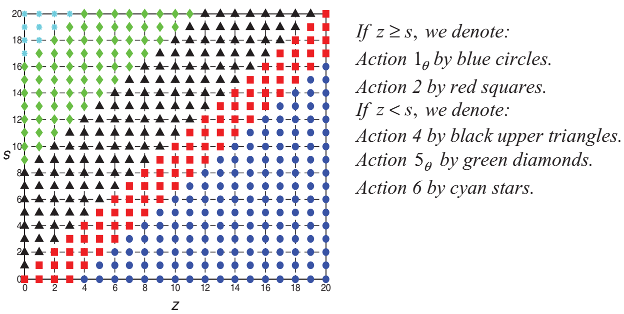

Figure 4. The optimal decisions for customer 7.

Table 4. The values of model parameters.

Parameters	Values
N	12
Q	20
$c_{j,j+1}$, $j \in \{1, \ldots, 11\}$	20, 24, 20, 28, 24, 30, 24, 26, 20, 18, 24
c_{j0}, $j \in \{1, \ldots, 12\}$	24, 22, 18, 20, 26, 22, 28, 20, 24, 26, 20, 26
π_j, $j \in \{1, \ldots, 12\}$	5, for each customer $j \in \{1, \ldots, 12\}$

of the minimum total expected cost f_0 is found to be approximately equal to 199.96. The computation time of the special-purpose dynamic programming algorithm is 3.4 s. It is considerably smaller than the computation time of the initial dynamic programming algorithm which is 5.57 s.

Example 4.2: We suppose that the parameters of the model take the values given in Table 4.

We assume that the probability mass function of the demand ξ_j, $j = 1, \ldots, 12$, is given by

$$\Pr(\xi_j = x) = \left(\sum_{i=0}^{Q} e^{-\lambda} \frac{\lambda^i}{i!} \right)^{-1} e^{-\lambda} \frac{\lambda^x}{x!}, \qquad x = 0, \ldots, Q, \tag{13}$$

with parameter $\lambda = 2$. Note that, in Beckmann, and Srinivasan (1987), a Poisson demand distribution was assumed in a stochastic inventory system with exponentially distributed delivery time. In Figure 5 below, we present graphs that show, as Q varies, the variation in the computation times, expressed in seconds, required by the initial dynamic programming algorithm and by the special-purpose dynamic programming algorithm.

We observe that, as Q increases, the computation times for both algorithms increase non-linearly. The computation time required by the special-purpose algorithm is smaller than the computation time required by the initial dynamic programming algorithm, especially for high values of Q.

Example 4.3: We assume that the parameters of the model take the values given in Table 5.

Table 5. The values of model parameters.

Parameters	Values
N	Takes values in the set $\{5, 10, \ldots, 195, 200\}$
Q	50
$c_{j,j+1}$, $j \in \{1, \ldots, N-1\}$	20
c_{j0}, $j \in \{1, \ldots, N\}$	24, if j is odd and 22 if j is even
π_j, $j \in \{1, \ldots, N\}$	5, for each customer $j \in \{1, \ldots, 12\}$

We also assume that, for each customer $j \in \{1, \ldots, N\}$, the probability mass function of the demand ξ_j is given by (13). In Figure 6 below, we present graphs that show, as N varies, the variation in the computation times, expressed in seconds, required by the initial dynamic programming algorithm and by the special-purpose dynamic programming algorithm.

We observe that, as N increases, the computation times for both algorithms increase approximately linearly. For $N \geq 60$, the computation time required by the special-purpose algorithm is considerably smaller than the computation time required by the initial dynamic programming algorithm.

Example 4.4: We assume that the parameters of the model take the values presented in Table 4. For

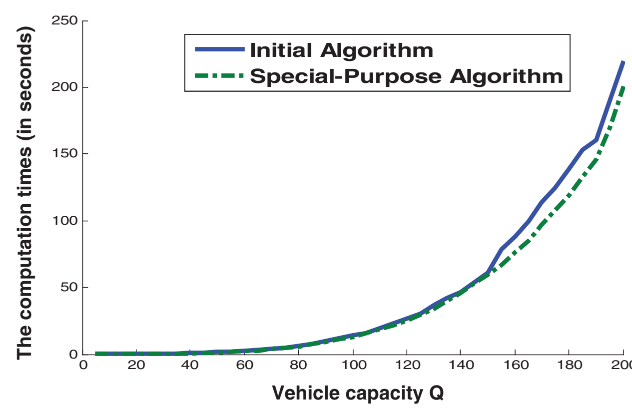

Figure 5. The computation times of the algorithms as Q varies.

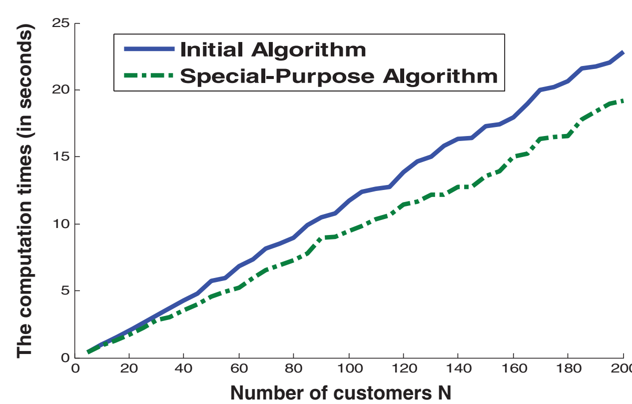

Figure 6. The computation times of the algorithms as N varies.

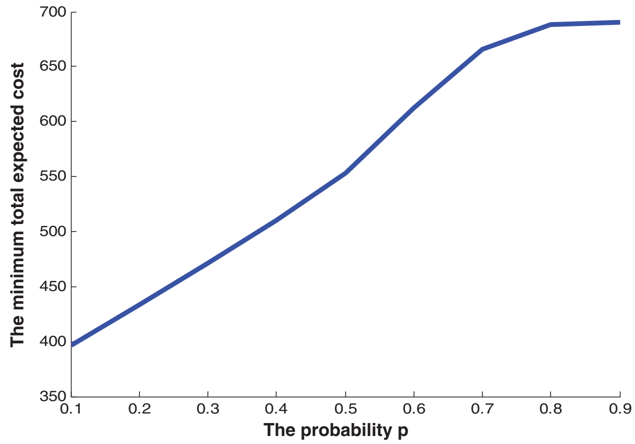

Figure 7. The minimum total expected cost as p varies.

binomial demand distributions (i.e. $\xi_j \sim \text{Bin}(Q, p)$, $j = 1, \ldots, 12$) in Figure 7, we present a graph that shows the variation in the minimum total expected cost f_0 as probability p varies.

We see that as p takes values in the set $\{0.1, \ldots, 0.6\}$, the minimum total expected cost increases rather quickly and approximately linearly. When p takes values in the set $\{0.7, 0.8, 0.9\}$, the minimum total expected cost increases rather slowly.

Example 4.5: We suppose that the parameters of the model take the values given in Table 6.

Note that the travel costs $c_{j,j+1}$, $j = 1, \ldots, 5$, from customer j to customer $j+1$ and the travel costs c_{j0} from customer j to the depot satisfy the triangle inequality. We also assume that for each customer $j \in \{1, \ldots, 6\}$, the demands ξ_j of the customers for the product follow the discrete uniform distribution, i.e. $\Pr(\xi_j = x) = (Q+1)^{-1}$, $x = 0, \ldots, Q$. In Figure 8 below, we present the optimal decisions for customer 1.

For customer 1, the action 1_θ is optimal in the states $(z, 2)$, $z \in \{5, \ldots, 10\}$. The optimal values of θ are presented in Table 7.

For customer 1, the action 5_θ is optimal in the states $(z, 5)$, $z \in \{0, \ldots, 4\}$. The optimal values of θ are presented in Table 8.

Table 6. The values of model parameters.

Parameters	Values
N	6
Q	10
$c_{j,j+1}$, $j \in \{1, \ldots, 5\}$	2, 3, 3, 2, 4
c_{j0}, $j \in \{1, \ldots, 6\}$	3, 5, 4, 4, 5, 4
π_j, $j \in \{1, \ldots, 6\}$	1.2, 1.3, 1.6, 1.9, 2, 1.1

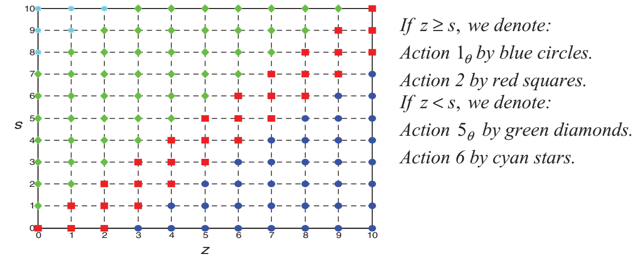

Figure 8. The optimal decisions for customer 1.

Table 7. The optimal values of θ when action 1_θ is optimal.

States (z, s)	Owed quantity s	Optimal value of θ
(5,2)	2	0
(6,2)	2	0
(7,2)	2	1
(8,2)	2	1
(9,2)	2	2
(10,2)	2	2

Table 8. The optimal values of θ when action 5_θ is optimal.

States (z, s)	Owed quantity $s - z$	Optimal value of θ
(0,5)	5	4
(1,5)	4	3
(2,5)	3	2
(3,5)	2	2
(4,5)	1	1

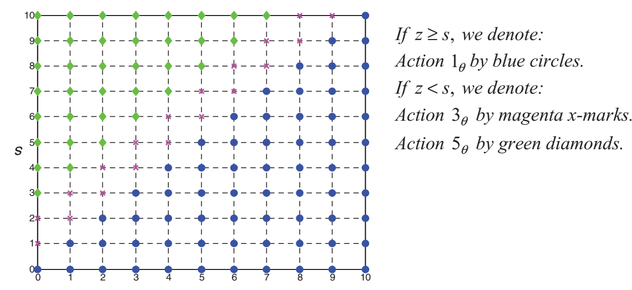

Figure 9. The optimal decisions for customer 5.

In Figure 9 below, we present the optimal decisions for customer 5. As it can be seen from Figure 9, action 3_θ appears as optimal action.

For customer 5, the action 3_θ is optimal in the states $(z, 7)$, $z \in \{5, 6\}$. The optimal values of θ are presented in Table 9.

Table 9. The optimal values of θ when action 3_θ is optimal.

States (z, s)	Owed quantity s	Optimal value of θ
(5,7)	7	3
(6,7)	7	5

Table 10. The values of model parameters.

Parameters	Values
N	10
Q	10
$c_{j,j+1}$, $j \in \{1, \ldots, 9\}$	27, 18, 27, 22, 24, 25, 23, 22, 25
c_{j0}, $j \in \{1, \ldots, 10\}$	24, 22, 23, 25, 22, 20, 21, 20, 19, 24
π_j, $j \in \{1, \ldots, 10\}$	6, for each customer $j \in \{1, \ldots, 10\}$

The value of the minimum total expected cost f_0 is found to be approximately equal to 37.37.

Example 4.6: We suppose that the parameters of the model take the values given in Table 10.

Note that the travel costs $c_{j,j+1}$, $j = 1, \ldots, 9$, from customer j to customer $j+1$ and the travel costs c_{j0} from customer j to the depot satisfy the triangle inequality. We assume that, for each customer j, $j \in \{1, \ldots, 10\}$, the demands ξ_j of the customers for the product are continuous random variables which follow the Gamma distribution right-truncated in the interval $[0, Q]$.

The probability density function $\varphi_j(x)$ of ξ_j is given by: $\varphi_j(x) = [F(Q)]^{-1} \frac{\lambda^\alpha x^{\alpha-1}}{\Gamma(\alpha)} e^{-\lambda x}$, $x \in [0, Q]$, re, $\alpha, \lambda > 0$, $\Gamma(\alpha) = \int_0^\infty e^{-u} u^{\alpha-1} du$, $\alpha > 0$ and $F(x) = [\Gamma(\alpha)]^{-1} \int_0^{\lambda x} e^{-u} u^{\alpha-1} du$, $x \geq 0$. The Gamma distribution seems to be a reasonable choice for the demand of a product since, as mentioned in p. 442 in the book of Tijms (2003), in inventory applications the Gamma distribution is often used to model demand distributions. We choose $\alpha = 4$ and $\lambda = 2$. We also choose $\rho = 0.05$ so that the discretised state space for each customer $j \in \{1, \ldots, 10\}$ is the set $\{(k * 0.05, l * 0.05) : k, l = 0, \ldots, 200\}$. In Figures 10 and 11 below, we present the optimal decisions for customers 3 and 4. The structure of the optimal policy, as expected, is of threshold-type described in Theorem 3.1.

The value of the minimum total expected cost f_0 is found to be approximately equal to 287.59. The computation time of the special-purpose dynamic programming algorithm is 139.77 s.. It is considerably smaller than the computation time of the initial dynamic programming algorithm which is 293.65 s. Both algorithms enable

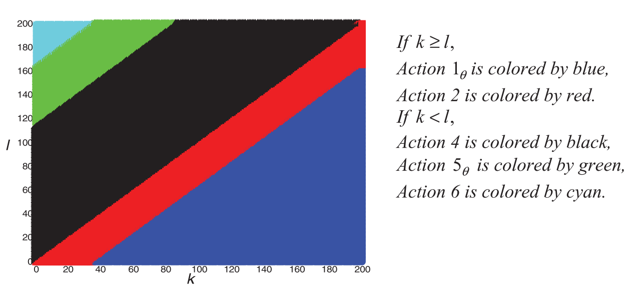

Figure 10. The optimal decisions for customer 3.

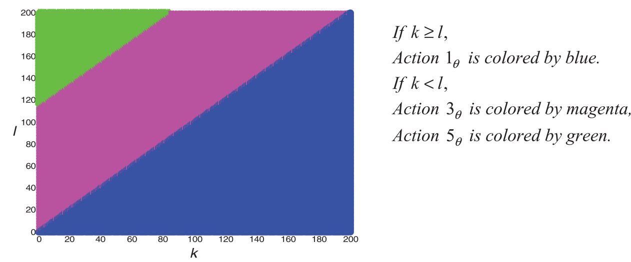

Figure 11. The optimal decisions for customer 4.

Table 11. The values of model parameters.

Parameters	Values
N	10
Q	Takes values in the set $\{3, 6, \ldots, 27, 30\}$
$c_{j,j+1}$, $j \in \{1, \ldots, 9\}$	27, 18, 27, 22, 24, 25, 23, 22, 25
c_{j0}, $j \in \{1, \ldots, 10\}$	24, 22, 23, 25, 22, 20, 21, 20, 19, 24
π_j, $j \in \{1, \ldots, 10\}$	6, for each customer $j \in \{1, \ldots, 10\}$

us to determine the optimal value of product quantity θ when the optimal actions are the actions 1_θ, 3_θ and 5_θ. For example, for customer 3, if the state is $(z, s) = (7.25, 2.4)$, then the optimal decision for the vehicle is to deliver to customer 3 product quantity θ equal to 1.9 and then proceed directly to customer 4 with a penalty cost equal to $\pi_3(s - \theta) = 3$ which is incurred due to unsatisfied demand of the product. If, again for customer 3, the state is $(z, s) = (1, 7.5)$, then the optimal decision for the vehicle is to deliver product quantity equal to 1 to customer 3, to go to the depot, to restock with load $Q = 10$, to return to customer 3 to deliver product quantity θ equal to 3.6 and then to proceed to customer 4 with the remaining product quantity equal to $Q - \theta = 6.4$ and with a penalty cost equal to $\pi_3(s - z - \theta) = 17.4$ which is incurred due to unsatisfied demand of the product. For customer 4, if the state is $(z, s) = (2.1, 4.1)$, then the optimal decision for the vehicle is to deliver product quantity θ equal to 1.6 to customer 4 and then to proceed directly to customer 5 with the remaining product quantity equal to $z - \theta = 0.5$. A penalty cost equal to $\pi_4(s - \theta) = 15$ is incurred due to unsatisfied demand.

Example 4.7: We suppose that the parameters of the model take the values given in Table 11.

We assume that for each customer j, $j \in \{1, \ldots, 10\}$, the demand ξ_j for the product is continuous random variable which follows the uniform distribution in the interval $[0, Q]$. Its probability density function $\phi_j(x)$ for each customer j, $j \in \{1, \ldots, 10\}$, is given by: $\phi_j(x) = Q^{-1}$, $x \in [0, Q]$. As in Example 4.6, we again choose $\rho = 0.05$. Note that, in Chien (1993), a problem of producing and transporting a product was studied, for which a continuous uniform demand distribution was used. In Table 12 below, we present the computation times,

Table 12. The computation times of the algorithms as Q varies.

Q	3	6	9	12	15	18	21	24	27	30
Initial algorithm	2.27	23.18	92.61	286.43	692.98	1519.5	3027.9	4693.7	7511.8	11,523
Special purpose algorithm	1.82	20.23	67.81	165.93	301.69	646.42	1111.6	1698.8	2435.6	3625.3

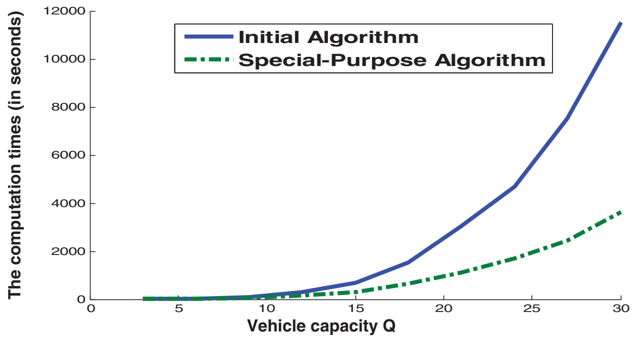

Figure 12. The computation times of the algorithms as Q varies.

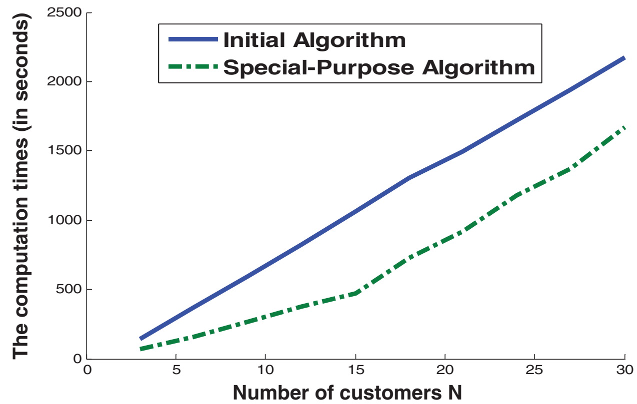

Figure 13. The computation times of the algorithms as N varies.

expressed in seconds, required by the initial dynamic programming algorithm and by the special-purpose dynamic programming algorithm.

In Figure 12 below, we also present the respective graphs that show, as Q varies, the variation in the computation times, expressed in seconds, required by the initial dynamic programming algorithm and by the special-purpose dynamic programming algorithm.

We observe that, as Q increases, the computation times for both algorithms increase non-linearly. 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 4.8: We suppose that the parameters of the model take the values given in Table 13.

For each customer $j \in \{1, \ldots, N\}$, it is assumed that ξ_j follows the continuous uniform distribution in $[0, Q]$. In Table 14 below, we present the computation times, expressed in seconds, required by the initial dynamic programming algorithm and by the special-purpose dynamic programming algorithm.

Table 13. The values of model parameters.

Parameters	Values
N	Takes values in the set $\{3, 6, \ldots, 27, 30\}$
Q	15
$c_{j,j+1}$, $j \in \{1, \ldots, N-1\}$	27, $j \in \{1, \ldots, N-1\}$
c_{j0}, $j \in \{1, \ldots, N\}$	24, if j is odd and 22, if j is even
π_j, $j \in \{1, \ldots, N\}$	6, for each customer $j \in \{1, \ldots, N\}$

In Figure 13 below, we present the respective graphs that show, as N varies, the variation in the computation times, expressed in seconds, required by the initial dynamic programming algorithm and by the special-purpose dynamic programming algorithm.

We observe that, as N increases, the computation times for both algorithms increase approximately linearly. 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.

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 expected total 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 5.1: Suppose that $Q = 10$. We assume that the number of customers N takes values in the set

Table 14. The computation times of the algorithms as N varies.

N	3	6	9	12	15	18	21	24	27	30
Initial algorithm	145.75	369.97	598.6	822.99	1063.4	1303.1	1496.8	1723.5	1938.7	2173.5
Special purpose algorithm	68.11	159.24	268.93	379.11	469.28	729.3	920.4	1181.6	1367.3	1673.9

$\{3, 4, 5, 6, 7, 8, 9\}$. 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 $C = (c_{ij}),\ i, j = 0, \ldots, 9$.

$$C = \begin{pmatrix} 0 & 24 & 22 & 23 & 25 & 22 & 20 & 21 & 20 & 19 \\ 24 & 0 & 27 & 25 & 26 & 23 & 27 & 22 & 24 & 29 \\ 22 & 27 & 0 & 18 & 20 & 22 & 19 & 23 & 25 & 24 \\ 23 & 25 & 18 & 0 & 27 & 24 & 20 & 24 & 19 & 26 \\ 25 & 26 & 20 & 27 & 0 & 22 & 18 & 25 & 21 & 22 \\ 22 & 23 & 22 & 24 & 22 & 0 & 24 & 22 & 18 & 23 \\ 20 & 27 & 19 & 20 & 18 & 24 & 0 & 25 & 26 & 24 \\ 21 & 22 & 23 & 24 & 25 & 22 & 25 & 0 & 23 & 24 \\ 20 & 24 & 25 & 19 & 21 & 18 & 26 & 23 & 0 & 21 \\ 19 & 29 & 24 & 26 & 22 & 23 & 24 & 24 & 21 & 0 \end{pmatrix},$$

These costs satisfy the triangle inequality. We assume that, for each customer $j \in \{1, \ldots, 9\}$, the penalty cost π_j incurred when an item is not delivered is equal to 4.5. We further assume that the demand ξ_j of each customer $j \in \{1, \ldots, 9\}$ is a random variable which follows the discrete uniform distribution, i.e. $\Pr(\xi_j = x) = (Q + 1)^{-1}$, $x = 0, \ldots, Q$. For $N \in \{3, \ldots, 9\}$ we consider the network consisting of customers $1, \ldots, N$. In Table 15, 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.

In Figure 14, we present the graphs that show, as N takes values in the set $\{3, \ldots, 9\}$, the variation in required computation times, expressed in seconds, if the initial dynamic programming algorithm and 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 smaller than the required computation time if the initial dynamic programming algorithm is used. The difference in computation times between the algorithms becomes significant for higher values of N.

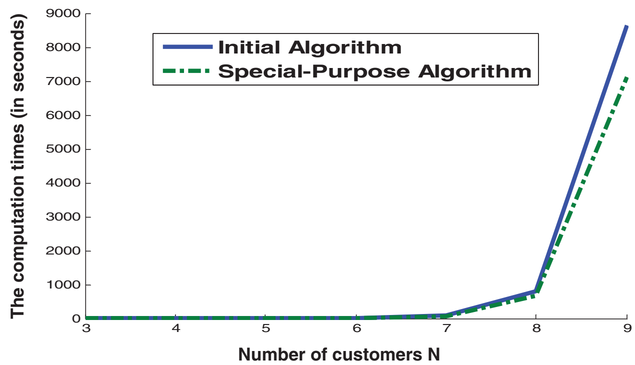

Figure 14. The computation times of the algorithms as N varies.

6. Summary of results and topics for future research

In this paper, a capacitated VRP was studied in which (1) the customers are served according to a predefined order, (2) the demands of the customers are stochastic and each customer's demand is less or equal to the vehicle capacity and (3) partial satisfaction of demand is allowed. The cost structure included travel costs between consecutive customers, travel costs between customers and the depot and penalty costs due to unsatisfied demands. We selected as decision epochs the epochs at which the vehicle visits for the first time each customer. A dynamic programming approach was proposed for the determination of the optimal routing strategy. It was proved that according to the optimal routing strategy, for a given value of a customer demand, the set of all possible loads carried by the vehicle is divided into four disjoint sets. If the load of the vehicle belongs to the first set, then the optimal action is to satisfy (fully or partially) or not to satisfy the demand of the customer and to proceed to the next customer. If it belongs to the second set, then the optimal action is to satisfy fully the demand of the customer, go to the depot for replenishment and, then proceed to the next customer. If it belongs to the third set, then the optimal action is to make one trip to the depot for restocking, return to the customer to satisfy fully or partially the demand of the customer and, then proceed to the next customer. If it belongs to the fourth set, then the optimal action is to go to the depot to restock the owed quantity, return to the customer to deliver the owed quantity, make a second trip to the depot for restocking and then go to the next

Table 15. 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	110.91	1,3,2	0.14	0.047
4	24	148.76	1,4,2,3	0.33	0.19
5	120	182.07	1,5,4,2,3	1.48	1.18
6	720	211.63	5,1,3,2,4,6	10.56	8.55
7	5040	244.54	7,1,5,3,2,4,6	86.35	71.39
8	40,320	273.08	7,1,5,8,3,2,4,6	798.02	664.95
9	362,880	305.78	9,4,6,2,3,8,5,1,7	8643.2	7112

customer. If the above Assumption (1) does not hold, it is possible to compute 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 in which (1) the vehicle delivers new products to the customers, (2) the vehicle collects expired products from the customers and (3) partial service is permissible.

Acknowledgments

The authors would like to thank two anonymous referees for useful suggestions that improved the presentation of the paper. The author Epaminondas G. Kyriakidis has been financed by the research program EP-2718-01 (RC/AUEB).

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

Epaminondas G. Kyriakidis: research program EP-2718-01 (RC/AUEB).

Notes on contributors

Epaminondas G. Kyriakidis is Professor in Applied Probability and in Stochastic Operational Research in Department of Statistics in Athens University of Economics and Business. His research interests are, among others, Markov decision processes, Control of Queues and Logistics. His work had been published, among others journals, in Journal of Applied Probability, in Statistics and Probability Letters and in Probabilty in the Engineering and Informational Science.

Theodosis D. Dimitrakos is Assistant Professor in Probability-Statistics in Department of Mathematics in University of the Aegean. His research interests are, among others, Stochastic Dynamic Programming, Optimal control of Biological Population Models and Machine Replacement Models. His work had been published, among others journals, in European Journal of Operational Research, in Methodolgy and Computing in Applied Probability and in International Journal of Production Economics.

References

Beckmann, M.J., & Srinivasan, S.K. (1987). An (*s,S*) inventory system with Poisson demands and exponential lead time. *Operations Research Spectrum, 9*, 213–217.

Berhan, E., Beshah, B., Kitaw, D., & Abraham, A. (2014). Stochastic vehicle routing problem: A literature survey. *Journal of Information and Knowledge Management, 3*, 1450022 (12 pages). doi: 10.1142/S0219649214500221

Bhusiri, N., Qureshi, A.G., & Taniguchi, E. (2014). The trade-off between fixed vehicle costs and time-dependent arrival penalties in a routing problem. *Transportation Research, Part E, 62*, 1–22.

Chien, T.W. (1993). Determining profit-maximizing production/shipping policies in a one-to-one direct shipping stochastic demand environment. *European Journal of Operational Research, 64*, 83–102.

Dantzig, G., & Ramster, R. (1959). The truck dispatching problem. *Management Science, 6*, 80–91.

Dikas, G., Minis, I., & Mamasis, K. (2016). Single vehicle routing with predefined client sequence and multiple warehouse returns: The case of two warehouses. *Central European Journal of Operations Research, 24*, 709–730.

Dimitrakos, T.D., & Kyriakidis, E.G. (2015). A single vehicle routing problem with pickups and deliveries, continuous random demands and predefined customer order. *European Journal of Operational Research, 244*, 990–993.

Eksioglu, B., Vural, A.V., & Reisman, A. (2009). The vehicle routing problem: A taxonomic review. *Computers and Industrial Engineering, 57*, 1472–1483.

Golden, B.L., & Yee, J.R. (1979). A framework for probabilistic vehicle routing. *AIIE Transactions, 11*(2), 109–112.

Haugland, D., Ho, S.C., & Laporte, G. (2007). Designing delivery districts for the vehicle routing problem with stochastic demands. *European Journal of Operational Research, 180*(3), 997–1010.

Kumar, S.N., & Panneerselvam, R. (2012). A survey on the vehicle routing problem and its variants. *Intelligent Information Management, 4*, 66–74.

Kyriakidis, E.G., & Dimitrakos, T.D. (2008). Single vehicle routing problem with a predefined customer sequence and stochastic continuous demands. *The Mathematical Scientist, 33*, 148–152.

Kyriakidis, E.G., & Dimitrakos, T.D. (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, Lecture Notes in Management Science*, *5*, 10–17.

Kyriakidis, E.G., & Dimitrakos, T.D. (2017). Single vehicle routing problem with a predefined customer sequence, stochastic demands and partial satisfaction of demands. In K.F. Doerner et al. (Ed.), *Operations Research Proceedings 2015* (157–164). Switzerland: Springer.
Laporte, G. (2009). Fifty years of vehicle routing. *Transportation Science, 43*, 408–416.
Liong, C.Y., Wan Rosmanira, I., Khairuddin, O., & Zirour, M. (2008). Vehicle routing problem: Models and solutions. *Journal of Quality Measurement and Analysis, 4*, 205–218.
Minis, I., & Tatarakis, A. (2011). Stochastic single vehicle routing problem with delivery and pickup and a predefined customer sequence. *European Journal of Operational Research, 213*, 37–51.
Pandelis, D.G., Karamatsoukis, C.C., & Kyriakidis, E.G. (2013a). Single vehicle routing problems with a predefined customer order, unified load and stochastic discrete demands. *Probability in the Engineering and Informational Sciences, 27*(1), 1–23.
Pandelis, D.G., Karamatsoukis, C.C., & Kyriakidis, E.G. (2013b). Finite and infinite-horizon single vehicle routing problems with a predefined customer sequence and pickup and delivery. *European Journal of Operational Research, 231*, 577–586.
Pandelis, D.G., Kyriakidis, E.G., & Dimitrakos, T.D. (2012). Single vehicle routing problems with a predefined customer sequence, compartmentalized load and stochastic demands. *European Journal of Operational Research, 217*, 324–332.
Pillac, V., Gendreau, M., Gueret, C., & Megaglia, A. (2013). A review of dynamic vehicle routing problems. *European Journal of Operational Research, 225*, 1–11.
Psaraftis, H.N., Wen, M., & Kontovas, C.A. (2016). Dynamic vehicle routing problems: Three decades and counting. *Networks, 67*, 3–31.
Ross, S.M. (1983). *An introduction to stochastic dynamic programming*. New York, NY: Academic Press.
Simchi-Levi, D., Chen, X., & Bramel, J. (2005). *The logic of logistics: Theory, algorithms and applications for logistics and supply chain management*. New York, NY: Springer.
Tatarakis, A., & Minis, I. (2009). Stochastic single vehicle routing with a predefined customer sequence and multiple depot returns. *European Journal of Operational Research, 197*, 557–571.
Tijms, H.C. (2003). *A first course in stochastic models*. Chichester: Wiley.
Toth, P., & Vigo, D. (Eds.), (2002). *The vehicle routing problem*. Philadelphia, PA: Siam.
Tsirimpas, P., Tatarakis, A., Minis, I., & Kyriakidis, E.G. (2008). Single vehicle routing with a predefined customer sequence and multiple depot returns. *European Journal of Operational Research, 187*, 483–495.
Yang, W.-H., Mathur, K., & Ballou, R.H. (2000). Stochastic vehicle routing problem with restocking. *Transportation Science, 34*, 99–112.
Zhang, J., Lam, W.H.K., & Chen, B.Y. (2016). On-time delivery probabilistic models for the vehicle routing problem with stochastic demands and time windows. *European Journal of Operational Research, 249*, 144–154.

Appendix

***Proof of Lemma 2.1*:** The proof is by induction on j. First, the induction basis is established by $f_N(z,s)$ being non-increasing in z (see Equations (9) and (10)). Thus, assuming that $f_{j+1}(z,s)$ is non-increasing in z, we will show that $f_j(z,s)$ is non-increasing in z. Let some fixed $s \in \{0,\dots,Q\}$. From (2) and the induction hypothesis it follows that $H_j(z,s)$ is non-increasing in $z \in \{s,\dots,Q\}$. □

Hence, in view of (1), we deduce that $f_j(z,s)$ is non-increasing in $z \in \{s,\dots,Q\}$. From (1) and (4), it follows that to prove that $f_j(s,s) \le f_j(s-1,s)$ it is enough to show that $H_j(s,s) \le A_j(s-1,s)$, $R_j \le B_j(s-1,s)$, $R_j \le C_j(s-1,s)$, $R_j \le D_j$. From (2) and (5), it follows that the inequality $H_j(s,s) \le A_j(s-1,s)$ is equivalent to

$$\min_{\theta\in\{0,\dots,s\}}\{\pi_j(s-\theta)+E[f_{j+1}(s-\theta,\xi_{j+1})]\} \le \min_{\theta\in\{0,\dots,s-1\}}\{\pi_j(s-\theta)+E[f_{j+1}(s-1-\theta,\ \xi_{j+1})]\},$$

which is valid in view of the induction hypothesis. From (3) and (6), it follows that the inequality $R_j \le B_j(s-1,s)$ is equivalent to $\pi_j \ge 0$, which clearly holds. The inequality $R_j \le C_j(s-1,s)$, in view of (3) and (7), is equivalent to

$$c_{j+1,0}+E[f_{j+1}(Q,\xi_{j+1})] \le c_{0j}+c_{j,j+1}+E[f_{j+1}(Q-1,\xi_{j+1})],$$

which is true in view of the triangle inequality and the induction hypothesis. From (3) and (8) it follows that the inequality $R_j \le D_j$ clearly holds. It remains to prove that $A_j(z,s), B_j(z,s), C_j(z,s)$ are non-increasing in $z \in \{0,\dots,s-1\}$. $B_j(z,s)$ is clearly non-increasing in $z \in \{0,\dots,s-1\}$. For $0 \le z \le s-2$, in view of induction hypothesis, we have

$$\begin{aligned} A_j(z+1,s) &= c_{j,j+1}+\min_{\theta\in\{0,\dots,z+1\}}\{\pi_j(s-\theta) \\ &\quad +E[f_{j+1}(z+1-\theta,\xi_{j+1})]\} \\ &\le c_{j,j+1}+\min_{\theta\in\{0,\dots,z\}}\{\pi_j(s-\theta) \\ &\quad +E[f_{j+1}(z-\theta,\xi_{j+1})]\} = A_j(z,s). \end{aligned}$$

For $0 \le z \le s-2$,

$$\begin{aligned} &C_j(z+1,s) \\ &= 2c_{0j}+c_{j,j+1}+\min_{\theta\in\{1,\dots,s-z-1\}}\{\pi_j(s-z-1-\theta) \\ &\quad +E[f_{j+1}(Q-\theta,\xi_{j+1})]\} \\ &= 2c_{0j}+c_{j,j+1}+\min\Big[\min_{\theta\in\{1,\dots,s-z-1\}}\{\pi_j(s-z-1-\theta) \end{aligned}$$

$$
\begin{aligned}
&\quad + E[f_{j+1}(Q-\theta, \xi_{j+1})\Big]\Big\},\ E[f_{j+1}(Q-s+z+1, \xi_{j+1})]] \\
&\leq 2c_{0j} + c_{j,j+1} + \min[\min_{\theta\in\{1,\dots,s-z-1\}}\{\pi_j(s-z-\theta) \\
&\quad + E[f_{j+1}(Q-\theta, \xi_{j+1})]\},\ E[f_{j+1}(Q-s+z, \xi_{j+1})]] \\
&= 2c_{0j} + c_{j,j+1} + \min_{\theta\in\{1,\dots,s-z\}}\{\pi_j(s-z-\theta) \\
&\quad + E[f_{j+1}(Q-\theta, \xi_{j+1})]\} = C_j(z,s),
\end{aligned} \tag{14}
$$

where (14) is a consequence of the inequality $E[f_{j+1}(Q-s+z+1, \xi_{j+1})] \leq E[f_{j+1}(Q-s+z, \xi_{j+1})]$, which follows from the induction hypothesis.

***Proof* of Lemma 2.2:** Consider some $j \in \{1, \dots, N-1\}$ and some $z \in \{0, \dots, Q\}$. For $s \in \{0, \dots, z-1\}$ we have

$$
\begin{aligned}
&\min_{\theta\in\{0,\dots,s\}}\{\pi_j(s-\theta) + E[f_{j+1}(z-\theta, \xi_{j+1})]\} \\
&\quad = \min[\min_{\theta\in\{0,\dots,s\}}\{\pi_j(s-\theta) + E[f_{j+1}(z-\theta, \xi_{j+1})]\}, \\
&\quad\quad E[f_{j+1}(z-s-1, \xi_{j+1})]] \\
&\quad = \min[\min_{\theta\in\{0,\dots,s\}}\{\pi_j(s+1-\theta) + E[f_{j+1}(z-\theta, \xi_{j+1})]\}, \\
&\quad\quad E[f_{j+1}(z-s-1, \xi_{j+1})]] \\
&\quad = \min_{\theta\in\{0,\dots,s+1\}}\{\pi_j(s+1-\theta) + E[f_{j+1}(z-\theta, \xi_{j+1})]\},
\end{aligned} \tag{15}
$$

□

where (15) follows from $E[f_{j+1}(z-s, \xi_{j+1})] \leq E[f_{j+1}(z-s-1, \xi_{j+1})]$, which holds from Lemma 2.1. Hence, $H_j(z,s)$ is non-decreasing in $s \in \{0, \dots, z\}$.

For $s \in \{z+1, \dots, Q-1\}$, we have

$$
\begin{aligned}
&\min_{\theta\in\{1,\dots,s-z\}}\{\pi_j(s-z-\theta) + E[f_{j+1}(Q-\theta, \xi_{j+1})]\} \\
&\quad = \min\Big[\min_{\theta\in\{1,\dots,s-z\}}\{\pi_j(s-z-\theta) \\
&\quad\quad + E[f_{j+1}(Q-\theta, \xi_{j+1})\Big]\Big\}, \\
&\quad\quad E[f_{j+1}(Q-s-1+z, \xi_{j+1})]] \\
&\quad = \min\Big[\min_{\theta\in\{1,\dots,s-z\}}\{\pi_j(s+1-z-\theta) \\
&\quad\quad + E[f_{j+1}(Q-\theta, \xi_{j+1})\Big]\Big\}, \\
&\quad\quad E[f_{j+1}(Q-s-1+z, \xi_{j+1})]] \\
&\quad = \min_{\theta\in\{1,\dots,s+1-z\}}\{\pi_j(s+1-z-\theta) \\
&\quad\quad + E[f_{j+1}(Q-\theta, \xi_{j+1})]\},
\end{aligned} \tag{16}
$$

where (16) follows from $E[f_{j+1}(Q-s+z, \xi_{j+1})] \leq E[f_{j+1}(Q-s-1+z, \xi_{j+1})]$, which holds from Lemma 2.1. Hence, $C_j(z,s)$ is non-decreasing in $s \in \{z+1, \dots, Q\}$. From (5), (6) it can be seen that the quantities $A_j(z,s)$ and $B_j(z,s)$ are non-decreasing in $s \in \{z+1, \dots, Q\}$.

***Proof* of Theorem 2.1:** The existence of the critical integer $h_1(j,s)$ is a direct consequence of $f_j(z,s)$ being non-increasing in z (see Lemma 2.1). The existence of $h_2(j,s)$ is a direct consequence of $A_j(z,s)$, $B_j(z,s)$, $C_j(z,s)$ being non-increasing in z. Note that the monotonicities of $A_j(z,s)$ and $C_j(z,s)$ in z have been shown in the proof of Lemma 2.1. The fact that $h_1(j,s)$ and $h_2(j,s)$ are non-decreasing in s is a direct consequence of Lemma 2.2. □