Flexible Services and Manufacturing Journal
https://doi.org/10.1007/s10696-018-9328-8

A branch-and-price algorithm for the home-caregiver scheduling and routing problem with stochastic travel and service times

Ran Liu[1] · Biao Yuan[1] · Zhibin Jiang[1]

Abstract
This paper addresses the solution methods for the stochastic home-caregiver scheduling and routing problem which arises in many service industries such as home care and home health care. In the problem, the caregiver's travel times and service times for customers are stochastic. A chance constraint is introduced to ensure the on-time service probability for the customers. Such stochastic travelling and service time and the chance constraint further complicate the problem. In this paper, a route-based mathematical model is introduced. A branch-and-price (B&P) algorithm and a discrete approximation method are combined to solve the problem. Herein, effective label algorithms are designed to generate negative reduced cost routes. The efficiency of the algorithm are improved by employing three acceleration strategies. The experiments on test instances validate the performances of the proposed B&P algorithm and demonstrate the necessity of considering the stochasticity of travel times of home-caregiver and service times to the customers.

Keywords Home health care · Home-caregiver · Scheduling · Routing · Branch-and-price · Stochastic time

1 Introduction

In home care or home health care (HHC), caregivers travel to the locations of customers in need of medical services due to acute or terminal illnesses, temporary or permanent disabilities, or chronic diseases. Such a service-to-home mode has witnessed a steady increase over the past years. For example, according to 2017 report of Organization for Economic Co-operation and Development (OECD) (2017), over the past 10 years in response to most people's preference to receive long-term health

✉ Ran Liu
liuran2009@sjtu.edu.cn

[1] Department of Industrial Engineering and Management, Shanghai Jiao Tong University, Shanghai, China

Published online: 25 September 2018

care services at home, many OECD countries have implemented programmes and benefits to support home-based health care, in particular for older people. As the demands for long-term home health care is increasing, in most OECD countries the annual expenses for HHC services and the number of the HHC workers are also rising. For example, in 2015, there are 10 and 5 long-term home health care workers per 100 people aged 65 and over in Israel and Japan, respectively. In Norway and United States, there are about 12 home health care workers per 100 old people (aged > 65) who provide the care to people limited in their daily activities at home or in institutions, excluding hospitals (OECD 2017).

The planning of caregivers is vital in the service-to-home industry. However, this work may be challenging and computationally difficult due to three practical reasons. First, each caregiver's route has its own start- and end-location nodes (Trautsamwieser and Hirsch 2011). Every morning a caregiver goes to the first customer's home from his or her own home and returns home after finishing the daily jobs. Second, the travel times among customers (which depends on road-traffic conditions) and service times for them (which depends on their physical states) are both stochastic. Third, due to uncertainty in the milieu, companies typically strive to improve the service quality by guaranteeing on-time service for customers. This is usually expressed as a chance constraint, i.e., the probability that a customer can be visited in a special time window is not lower than a pre-defined value. However, given the difficulty in scheduling for the caregivers due to the uncertainty, such a chance constraint further complicates the problem. In this paper, we undertake to design optimization methods to address this home-caregiver scheduling and routing problem with stochastic travel and service times.

With the rapid growing of HHC services, an increasing number of researchers study the operation management problems in HHC industry, especially the detailed operational level problem involving daily scheduling and routing of caregivers. For example, Bertels and Fahle (2006) address a nurse routing problem in HHC using a combination of linear programming, constraint programming, and metaheuristic. Eveborn et al. (2006) focus on a home care staff planning problem arising in Sweden where people receive home care from the local authorities. They formulate the problem using a set partitioning model and applying a repeated matching algorithm for solutions. Trautsamwieser et al. (2011) present a model formulation and solution approach for the daily planning of HHC services in times of natural disasters (such as flood disasters). Shao et al. (2012) and Zachariadis et al. (2015) solve the problem of constructing weekly schedules for the therapists, who offer different levels of services to meet demand spread over a large geographic area in the United States Midwest. Rasmussen et al. (2012) proposes an exact branch-and-price algorithm for the HHC caregiver scheduling and routing problem with temporal dependencies. Liu et al. (2017) study the caregiver scheduling and routing problem, considering the lunch break requirements. Yalçındağ et al. (2016) study the patients-to-caregivers assignment problem that the home health care companies must address before generating the daily routes. Since assignment decision is typically made without knowing the visiting sequence, the authors propose a data-driven method to estimate the travelling time of caregivers in the assignment problem. Yalçındağ and Matta (2017) propose a two-stage approach for the assignment, scheduling and routing problems of the HHC services. The approach in the work of Yalçındağ and Matta (2017) decomposes the whole problem into assignment, scheduling and routing via two stages where scheduling decision is incorporated into both the assignment and routing stages. In their approach, time windows are considered at assignment level using a probabilistic model without considering the routing problem. Readers are referred to the recent surveys of Cissé et al. (2017) and Fikar and Hirsch (2017) for more relative literature.

Although many precise and heuristic algorithms have been proposed for the HHC home-caregiver scheduling and routing problem, few studies focus on the stochastic version of problem. For example, Lanzarone et al. (2012) first exploit the variability of patients' demands in the *assignment problem* of HHC, i.e., the decision regarding which caregiver among the compatible ones will take care of which patients. The authors propose a set of mathematical programming models to treat the assignment problem for different types of service providers. Given the high variability of patient demands, models are developed under the assumption that patients' demands are either deterministic or stochastic. Lanzarone and Matta (2014) addresses the problem of assigning newly admitted patients to their reference nurse, while maintaining continuity of care. Specifically, this paper proposes an analytical policy for solving the nurse-to-patient assignment problem within the HC setting, taking into account the stochasticity of new patient's demand and nurses' workloads. To our knowledge, Yuan et al. (2015) first address the stochastic caregiver scheduling and routing problem. In Yuan et al. (2015), travel time between two customers is deterministic, and service time to a patient is stochastic due to their varying health conditions. Each customer has a soft time window, i.e., a customer is allowed to be visited and served by a caregiver later than the left part of the time windows. A stochastic programming model with recourse is proposed to formulate the problem in which the expected penalty for late arrival at customers is considered. The differences between Liu et al. (2017) and Yuan et al. (2015) and this study are obvious. First, in this paper both travel times and service times are stochastic. Second, in this paper each customer requires a hard service time window, which truncates the distributions of arrival time and starting service time (i.e. the closed formula for the failure cost cannot be obtained). Third, in this paper, each caregiver route starts from and ends at his/her home, i.e., a multi-depot version of stochastic caregiver scheduling and routing problem. Furthermore, since papers (Liu et al. 2017; Yuan et al. 2015) do not consider chance constraints and hard time windows simultaneously, even if we regard the travel time between two nodes is deterministic in the current paper, the approaches of Liu et al. (2017) or Yuan et al. (2015) cannot be used directly to solve the current problem.

Since the caregiver route is similar to the vehicle route, the proposed problem can be viewed as a special multi-depot vehicle routing problem with time windows (MDVRPTW) with *stochastic travel and service times.* Although the deterministic version of this problem, i.e. the deterministic MDVRPTW, has been elucidated in the literature (Desaulniers et al. 1998; Polacek et al. 2004; Vidal et al. 2013), little attention has been paid to the corresponding problem with stochastic travel and service times. Li et al. (2010) devise a tabu search (TS) heuristic with simulation to solve the one-depot VRPTW with stochastic times in which all the vehicles started from and ended at only one depot. Zhang et al. (2013) also propose an iterated TS heuristic with a discrete approximation method to address this one-depot problem. Ehmke et al. (2015) develop the LANTIME TS heuristic with an approximation approach based on the extreme value theory to solve the problem. To the best of our knowledge, in the field of vehicle routing problem, the multi-depot home-caregiver scheduling and routing problem with stochastic travel and service times has not been studied.

The rest of this paper is structured as follows. Section 2 introduces the route-based model for the problem and the calculation of the on-time service probability. A branch-and-price (B&P) algorithm is presented in Sect. 3 for solving the problem, of which the effectiveness is analyzed in Sect. 4. The paper ends in Sect. 5 with some conclusions and directions for future research.

2 Problem description

A route-based mathematical model is first constructed to describe the problem in this section, followed by the calculation of the on-time service probability.

2.1 Mathematical model

The model can be defined on a graph $G=(V, A)$, where V and A are the node and arc sets, respectively. Set $K=\{1, 2,\ldots, m\}$ denotes the caregiver set. The node set $V=H\cup N$, where $H=\{1, 2,\ldots, m\}$ corresponds to the caregivers' home set and $N=\{1, 2, 3,\ldots, n\}$ denotes the customer set. A travel time t_{ij} is associated with each arc $(i, j) \in A$ between nodes i and j. The travel cost per unit time is ct. Each customer i has a service time t_i, and a time window $[a_i, b_i]$ for starting services. Each customer can be visited by only a subset of caregivers because of skill constraints. Each caregiver $k\in K$ can serve a subset of customers and has a service cost per unit time cs_k. Let Ω_k represent the set of feasible routes visited by caregiver k; c_r is the operational cost of route $r\in\Omega_{k,}$ i.e., the total expected travel time and service time consumed in this route. The home-caregiver scheduling and routing problem consists of determining the set of caregiver's routes incurring the minimal overall cost in order to serve customers under the following five assumptions:

Assumption A Each caregiver $k\in K$ starts from and ends at his or her home $h\in H$ and involves a visit at each customer's location no more than once a day;
Assumption B Each customer is visited by no more than one caregiver;
Assumption C For each customer i who is to be served based on the planning, a service level β $(0<\beta<1)$ is guaranteed, i.e., the probability that such a customer i can be visited in time window $[a_i, b_i]$ is not lower than β. This is also called a *chance constraint* in the stochastic programming.
Assumption D The travel and service times are independent stochastic variables. In real-world HHC operations the distributions of travelling and service time are very

complex. But, in most scenarios the independent assumption is valid; it is adopted by our work to simplify the problem.

Assumption E Similar with the basic assumptions of Nickel et al. (2012) and Rasmussen et al. (2012), it is permissible that some customers may be not visited and serviced by any caregiver in the solution. However, to control the number of such uncovered customers, we assume that, for each customer who is not serviced, the company suffers a penalty value η in the operation cost. In practice, there may be uncovered customers due to the shortage of caregivers. In such case, usually the HHC company may ask a collaboration from one other company and pay a cost so that the unscheduled customers can be served by caregiver from other company.

To clearly describe the home-caregiver scheduling and routing problem with stochastic travel and service times, an example with 3 caregivers and 10 customers is shown in Fig. 1, where the probability that such each customer can be visited in his/her time window is shown in bracket. Suppose the minimal service level β=0.78, one customer, marked in red, cannot be planned to be serviced by any caregiver due to the possible service level for this customer cannot reach 0.78.

The mathematical formulation of the problem is presented below following the preceding definition, which is a route-based model.

Decision variables

$$\alpha_{ir} = \begin{cases} 1, & \text{if customer } i \text{ is in router } r \\ 0, & \text{otherwise} \end{cases}$$

$$y_r = \begin{cases} 1, & \text{if router } r \text{ is select in the optimal solution} \\ 0, & \text{otherwise} \end{cases}$$

$$z_i = \begin{cases} 1, & \text{if customer } i \text{ is not served the optimal solution} \\ 0, & \text{otherwise} \end{cases}$$

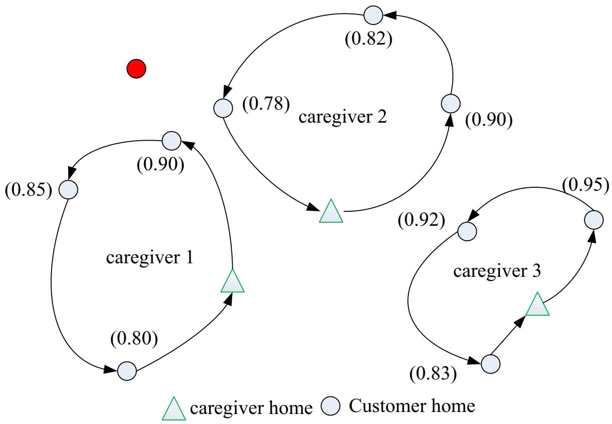

Fig. 1 A solution to the home-caregiver scheduling and routing problem

Objective function

$$\min \sum_{k\in K}\sum_{r\in\Omega_k} c_r y_r + \sum_{i\in N} \eta z_i \tag{1}$$

Subject to,

$$\sum_{k\in K}\sum_{r\in\Omega_k} \alpha_{ir} y_r + z_i = 1 \quad \forall i \in N \tag{2}$$

$$\sum_{r\in\Omega_k} y_r \le 1 \quad \forall k \in K \tag{3}$$

$$y_r = \{0,\ 1\} \quad \forall k \in K, \quad r \in \Omega_k \tag{4}$$

$$z_i = \{0, 1\} \quad \forall i \in N \tag{5}$$

Objective function (1) serves to minimize both the expected total operational cost of selected routes and the penalty value for customers who are not to be serviced according to the planning. Constraint (2) ensures that each customer is either visited by one caregiver or not covered, whereas Constraint (3) denotes that at most one route is assigned to each caregiver. Lastly, Constraints (4) and (5) impose the integrality of the decision variables.

2.2 Calculation of on-time service probability

The calculation of the on-time service probability is a key problem in our home-caregiver scheduling and routing problem with stochastic travel and service times. It may be illustrated by taking route r as an example, for which r_i is the ith customer in the route, n_r is the number of customers, r_0 and r_{n_r+1} represent the origin and destination, t_i^a is the time of arrival at the location customer i, t_i^s is the starting service time for customer i, and t_i^d is the corresponding departure time. The arrival time in the deterministic situation is recursively computed as follows.

$$t_{r_1}^a = t_{r_0 r_1} \tag{6}$$

$$t_{r_i}^s = \max\{t_{r_i}^a,\ a_{r_i}\} \quad \text{for } 1 \le i \le n_r \tag{7}$$

$$t_{r_i}^d = t_{r_i}^s + t_{r_i} \quad \text{for } 1 \le i \le n_r \tag{8}$$

$$t_{r_{i+1}}^a = t_{r_i}^d + t_{r_i r_{i+1}} \quad \text{for } 1 \le i \le n_r \tag{9}$$

Due to Eq. (7), difficulties may arise in calculating the probability in the stochastic situation. The principle of the discrete approximation method is to replace the original distribution of travel and service times by a set of sequential discrete points (Zhang et al. 2013).

For example, considering travel time $t_{r_i r_{i+1}}$ and a set of L discrete points $\{0.5\varepsilon, 1.5\varepsilon, \ldots, (L-0.5)\varepsilon\}$ in the interval [0, 1] where $\varepsilon = 1/L$, a sequence of discrete travel times $e_\upsilon^{t_{r_i r_{i+1}}}$ $(\upsilon = 1, 2, \ldots, L)$ is obtained as follows.

$$e_\upsilon^{t_{r_i r_{i+1}}} = F^{-1}(\upsilon\varepsilon - 0.5\varepsilon) \quad \text{for } 1 \le \upsilon \le L \tag{10}$$

where $F^{-1}(\bullet)$ is the inverse cumulative distribution function (CDF) of $t_{r_i r_{i+1}}$. The value of L is set to be 100 (Rasmussen et al. 2012).

The set of sequential discrete travel times approximates the inverse CDF of $t_{r_i r_{i+1}}$ as follows.

$$\kappa^{t_{r_i r_{i+1}}}(\upsilon\varepsilon) = e_\upsilon^{t_{r_i r_{i+1}}} \quad \text{for } 1 \le \upsilon \le L \tag{11}$$

where $\kappa^{t_{r_i r_{i+1}}}$ links the relationship between probability $\upsilon\varepsilon$ and discrete point $e_\upsilon^{t_{r_i r_{i+1}}}$. The distributions of other travel and service times are approximated in the same rule.

The arrival time distribution of customer r_1 is approximated as

$$\kappa^{t_{r_1}^a}(\upsilon\varepsilon) = e_\upsilon^{t_{0 r_1}} \quad \text{for } 1 \le \upsilon \le L \tag{12}$$

The starting service time distribution of customer r_i is approximated as

$$\kappa^{t_{r_i}^s}(\upsilon\varepsilon) = \max\left\{ e_\upsilon^{t_{r_i}^a},\ a_{r_i} \right\} \quad \text{for } 1 \le \upsilon \le L \tag{13}$$

The departure time distribution of customer r_i is calculated: a set of L^2 discrete points $e_\upsilon{}'$ $(\upsilon = 1, 2, \ldots, L^2)$ is firstly generated by Eq. (14), then all L^2 elements of $e_\upsilon{}'$ are sorted in an ascending order as represented by a new set of discrete values e_υ, and finally the approximate departure time distribution is obtained by Eq. (15).

$$e'_\upsilon = \kappa^{t_{r_i}^s}(\upsilon_1\varepsilon) + \kappa^{t_{r_i}}(\upsilon_2\varepsilon) \quad \text{for } 1 \le \upsilon_1 \le L,\ 1 \le \upsilon_2 \le L,\ 1 \le \upsilon \le L^2 \tag{14}$$

$$\kappa^{t_{r_i}^d}(\upsilon\varepsilon) = \frac{1}{L} \sum_{\upsilon_2=1}^{L} e_{(\upsilon-1)L+\upsilon_2} \quad \text{for } 1 \le \upsilon \le L \tag{15}$$

The arrival time distribution of other customers is calculated in the same manner as the departure time distribution.

The on-time service probability at customer r_i is computed as

$$P(t^{a}_{r_i} \leq b_{r_i}) = \begin{cases} 0, & \text{if } b_{r_i} < e_1^{t^a_{r_i}} \\ \upsilon_{r_i}\varepsilon, & \text{if } e_{\upsilon_{r_i}}^{t^a_{r_i}} \leq b_{r_i} < e_{\upsilon_{r_i}+1}^{t^a_{r_i}}, \ 1 \leq \upsilon_{r_i} < L \\ 1, & \text{if } b_{r_i} \geq e_L^{t^a_{r_i}} \end{cases} \tag{16}$$

3 Solution approach

Given the considerable cardinality of Ω_k in model (1) to (5), this model cannot be directly solved with an optimization solver. For such a route-based model, the B&P algorithm is usually adopted (Dayarian et al. 2015a, b). It relies on the branch-and-bound framework in which the lower bound of each node in the search tree is calculated by the column generation algorithm (Feillet 2010). In this paper, the frameworks of the proposed B&P and column generation algorithm are adapted from our previous study (Liu et al. 2017). However, while the stochastic travel times, service times, and chance constraints are not considered in Liu et al. (2017), the components of algorithm such as pricing algorithm and acceleration strategies have to be designed according to the characteristics of the problem in this study.

The original problem contains a master problem and several pricing sub-problems. In each iteration of the column generation, the restricted master problem (RMP) is solved upon the relaxation of the integrity of the decision variables and inclusion of a subset of routes generated by solving the sub-problems.

In the search tree, the root node requires several columns to initialize the RMP. At other nodes, complying with branching decisions, the RMP inherits a subset of columns from the parent node. When the RMP at a node is solved by the column generation algorithm, if the solution is integral, the smaller one between it and the current upper bound is chosen as the new incumbent upper bound; in such cases, branching is not performed from the current node. Otherwise, branching rules are used to prune fractional solutions. The iterative process stops until the tree is explored completely.

3.1 RMP and pricing sub-problem

The RMP including a subset of columns $\Omega_k{}' \subseteq \Omega_k$ is as follows.

Objective function

$$\min \sum_{k\in K} \sum_{r\in\Omega'_k} c_r y_r + \sum_{i\in N} \eta z_i \tag{17}$$

Subject to,

$$\sum_{k\in K} \sum_{r\in\Omega'_k} \alpha_{ir} y_r + z_i = 1 \quad \forall i \in N \tag{18}$$

$$\sum_{r\in\Omega_k'} y_r \leq 1 \quad \forall k \in K \tag{19}$$

$$y_r \geq 0 \quad \forall k \in K,\ r \in \Omega_k' \tag{20}$$

$$z_i \geq 0 \quad \forall i \in N \tag{21}$$

Constraints (18) and (19) are identical to Constraints (2) and (3). The set Ω_k' is continuously expanded from the solutions of pricing sub-problems. In the column generation algorithm, the pricing process contains m sub-problems, each of which is an elementary shortest-path problem with resource constraints corresponding to a caregiver. Taking the dual solution of the RMP as the input, the sub-problem is used to identify one or more negative reduced cost columns (Athanasopoulos and Minis 2013).

Let λ_i and μ_k be the dual variables associated with Constraints (18) and (19), respectively. The objective function of the pricing sub-problem (or the reduced cost of route r) associated with caregiver k is computed as

$$c_r - \sum_{i\in N} \alpha_{ir}\lambda_i - \mu_k \tag{22}$$

3.2 Pricing algorithm

To demonstrate the optimality of solutions to the RMP, the label algorithm has to be used at least once. Given the typically time-consuming acquisition of the optimal route, it is common practice to generate columns heuristically to accelerate the iterative process (Liberatore et al. 2011). Thus a bi-level pricing procedure is devised in which the variable neighborhood descent (VND) heuristic is first adopted to generate promising columns; if it fails, the label algorithm is used.

3.2.1 VND heuristic

For each route (corresponding to a basic variable of the RMP solution), the VND heuristic uses *removal*, *swap*, and *insertion* operators to find new columns. The removal operator generates new routes by removing one of visited customers in the route with a neighboring size O(n) where n is the number of customers. The swap operator acquires new routes by swapping a visited customer in the route with another unvisited one with a size O(n^2); and the insertion operator obtains new routes by inserting one of unvisited customers into the route with a neighboring size O(n). The above operators are sequentially performed. If an operator finds new columns, they are returned to the RMP; otherwise, the next operator is employed. We also test other operators such as 2-opt which shows no obvious algorithm performance improvement. To control the computational time, we only keep three basic operators as mentioned above. When the VND heuristic fails to identify one new column, the label algorithm is invoked.

3.2.2 Label algorithm

The label algorithm starts with an initial label at the origin and extends the labels along the arcs using extension functions. To avoid producing an exponential number of labels, only non-dominated labels are kept at each node according to the dominance rules as described later.

A label is represented by $l=(i, \bar{c}, t_i^a, U)$ where i is the last visited node, $\bar{c}$ the reduced cost corresponding to Eq. (22), t_i^a the arrival time at node i, and U the set of nodes visited along the route. The initial label associated with the origin is $l_0=\{0, 0, 0, \varnothing\}$. The components of label l are denoted by $i(l)$, $\bar{c}(l)$, $t_i^a(l)$ and $U(l)$.

Given that label l represents a route ending at node i, i.e., $i(l)=i$, the extension of this label along arc (i, j) generates a new label l' using extension functions:

$$i(l') = j \tag{23}$$

$$\bar{c}(l') = \bar{c}(l) + \ ct \cdot E(t_{ij}) + cs \cdot E(t_j) - \lambda_i/2 - \lambda_j/2 \tag{24}$$

$$t_j^a(l') = max(t_i^a(l),\ \ a_i) + t_i + t_{ij} \tag{25}$$

$$U(l') = U(l) \cup \{j\} \tag{26}$$

For ease of description, Eq. (24) calculates the reduced cost of the route ignoring the k index, and Eq. (25) computes the arrival time distribution at customer j in accordance to the steps in Sect. 2.2. Label l' is deemed feasible if $j \notin U(l)$ and $P(t_j^a(l') \le b_j) \ge \beta$.

To find non-dominated labels, dominance rules are devised. For two labels l_1 and l_2 ending at node i, label l_1 dominates label l_2 if the following conditions are satisfied:

(a) $\bar{c}(l_1) \le \bar{c}(l_2)$,
(b) $t_i^a(l_1) \le t_i^a(l_2)$,
(c) $U(l_1) \subseteq U(l_2)$.

For Condition (b), the comparison of $t_i^a(l_1) \le t_i^a(l_2)$ means that $e_\upsilon^{t_i^a(l_1)}$ is not larger than $e_\upsilon^{t_i^a(l_2)}$ for $\upsilon = 1, 2,\ldots, L$.

3.2.3 Acceleration strategies of label algorithm

The following strategies are used to accelerate the basic label algorithm.

(1) Inclusion of unreachable nodes

In terms of label l, $U(l)$ can be augmented to contain customers who are unreachable from the extension of label l due to the resource constraints and have not been visited along the route, i.e., $j \notin U(l)$ and $P(t_j^a(l) \le b_j) < \beta$. Let $\hat{U}$ replace the component U of the label; accordingly, Condition (c) in the dominance rules is modified by (d) $\hat{U}(l_1) \subseteq \hat{U}(l_2)$.

(2) Decremental state-space relaxation

The decremental state-space relaxation (DSSR) technique allows the generation of routes with cycles (i.e., relaxing constraints of elementarily) (Boland et al. 2006). The relaxation is iteratively tightened by adding some nodes into the critical node set $E \subseteq N$ for which the elementarity is currently required. At the end of the label algorithm, if the optimal route contains no cycles, the procedure stops; otherwise, the state-space is augmented by forbidding multiple visits to newly selected critical nodes and the procedure restarts. The augmentation policy selects the nodes that are visited more than one time in the optimal route as new critical nodes.

When extending label l along arc (i, j) to create label l', the extension function (26) is replaced by

$$U(l') = (U(l) \cup \{j\}) \cap E \tag{27}$$

In the context of column generation, the label algorithm with the DSSR stops if one of the following criteria is satisfied: a negative reduced cost elementary route is obtained, or no negative reduced cost (cyclic) route is found.

(3) *ng*-route relaxation

The *ng*-route relaxation is designed for the capacitated VRP and VRP with time windows (Baldacci et al. 2011). Let N_r represent the set of customers visited by a partial route r. For each customer $i \in N$, let $\Gamma_i \subseteq N$ be the original neighborhood of customer i which contains the Δ_{ng} closest customers (including customer i himself or herself). For label l associated with a given partial route $r = \{r_0, r_1, r_2, \dots, r_i\}$, the set $\Pi(l) \subseteq N_r$ of prohibited extensions of route r is defined as follows.

$$\Pi(l) = \left\{ r_j \in N_r \,:\, r_j \in \bigcap_{s=j+1}^{i} \Gamma_{r_s}, j = 1, \dots, i-1 \right\} \cup \{r_i\} \tag{28}$$

The set U is no longer used to check the feasibility of label extensions. It is replaced by the set Π. In the label l_0 at the origin, $\Pi(l_0)$ is set to be $\varnothing$. Label l can be extended along arc (i, j) to create label l' if $j \notin \Pi(l)$ and $P(t^a(l') \le b_j) \ge \beta$, the extension function (26) is replaced by

$$\Pi(l') = (\Pi(l) \cap \Gamma_j) \cup \{j\}. \tag{29}$$

Condition (c) in the dominance rules is replaced by

$$\text{(e)} \quad \Pi(l_1) \subseteq \Pi(l_2).$$

The aforementioned strategies are incorporated into the label algorithm, as represented by an Unreachable-*ng*-DSSR. This Unreachable-*ng*-DSSR contains the *ng*-route relaxation procedure in which, at the beginning, the empty set $\bar{\Gamma}_i \subseteq \Gamma_i$ (named the applied neighborhood) is employed instead of the original neighborhood Γ_i. In the search tree, $\bar{\Gamma}_\mathrm{i}$ is set to be $\varnothing$ at the root node and then is used for other nodes.

As the rationale of DSSR, when the label algorithm terminates, all negative reduced cost routes with cycles and without cycles satisfying the *ng*-relaxation rule in regard to the original neighborhoods (called the *ng*-feasible routes) are inserted into the RMP; if no such route exists but the optimal route with the negative reduced cost is not *ng*-feasible, some of the applied neighborhoods are augmented and the label algorithm restarts with the new applied neighborhoods. The augmentation policy enlarges the applied neighborhoods $\bar{\Gamma}_j$ of those nodes forming a cycle beginning and ending at node i when the *ng*-relaxation rule is violated in the optimal route, and appends node i into those neighborhoods (i.e., $\bar{\Gamma}_j = \bar{\Gamma}_j \cup \{i\}$). The inclusion of unreachable nodes limited to the applied neighborhoods is also adopted in the extension functions to further accelerate the label algorithm.

Since the label algorithm with the Unreachable-*ng*-DSSR yield *ng*-feasible columns which may contain cycles, α_{ir} in Constraints (18) represents the number of times that customer i is visited in route r, and the "=" is replaced by "≥".

3.3 Other algorithmic components

In this section, we explain some other components of the B&P algorithm, such as the initial columns and the branching strategy.

3.3.1 Initial columns and upper bound

The RMP at the root node requires initialization by inserting several columns from a feasible solution. The following heuristic is used to generate the initial columns. First, customers are sorted in the ascending order of magnitude of the angle that they form with the center of the region. Subsequently customer j is inserted into the route of caregiver k who can serve him or her to minimize the total cost. The insertion of customer j is performed between two successive customers in the route. If the customer cannot be assigned to any caregiver, he or she is not visited in the solution.

The initial upper bound is obtained by solving the models (1) to (5), including all columns generated during the process of solving the RMP at the root node.

3.3.2 Branching strategy

The hierarchical branching scheme is devised to remove fractional solutions, as detailed below:

(1) *Branching on uncovered customers.* The decision on whether a customer is covered in the optimal solution depends on variables z_i. If some of them are fractional, the variable z_i farthest to an integer is selected to branch. Two child

nodes are created with the addition of constraints $z_i = 1$ and $z_i = 0$ into the RMP, respectively. For the child node with $z_i = 1$, the columns including customer i, inherited from the current node, should be removed, and no columns involving customer i can be generated. For the other child node with $z_i = 0$, no modification is required for the column set of RMP and pricing process.

(2) *Branching on the assignment of caregivers to customers.* When each value of z_i is integral, the assignment variables of caregivers to customers, calculated as $\theta_{ik} = \sum_{r \in \Omega_k} \alpha_{ir} y_r$ are used. If the values of θ_{ik} are fractional, the one farthest away from an integer is selected as the branching variable. For the child node whose $\theta_{ik} = 1$, the columns with customer i should be removed from the column set of other caregiver $k' \neq k$ in the RMP and not be generated when solving the corresponding pricing sub-problems. For the child node whose $\theta_{ik} = 0$, the columns with customer i should be deleted from the column set of caregiver k in the RMP and not be produced during the process of solving the pricing sub-problem associated with caregiver k.

3.3.3 Search strategy

In the algorithm the best-first strategy is used for the nodal search. When choosing a node in the search tree to be computed next, that with the smallest lower bound is selected.

4 Computational experiments

In this section, we examine the algorithmic performance through numerous computational experiments. First, we generate the test instances. Next, we determine the values of parameters of the proposed B&P algorithm and analyze the acceleration strategies of the label algorithm. Finally, we report all computational results. The algorithms are coded in C++. The RMPs are solved by IBM Cplex 12.5. The experiments are performed on Xeon E5-2650 2.6 GHz processor and 64 GB RAM. The B&P algorithm terminates until the optimal solution is found.

4.1 Test instances

Given the lack of existing research on the proposed problem, the test instances are generated to evaluate the performance of the algorithm. For each test instance, a predefined number of customers and caregivers are spread in a square with its length and width both measuring 100, according to a discrete uniform distribution. The mean travelling time equals the Euclidean distance between two nodes whereas the mean service time for each customer is a uniform random number in the interval (15, 30). The standard deviations of the travellling and service time equals the corresponding mean value multiplied by a parameter γ, where γ is a uniform number in the interval (0.2, 0.25).

The time windows of customers are generated as follows. First, we let the width of the time window t_l to be uniformly distributed in the discrete interval [60, 120]. Then we give a starting point $a_0 = 0$ and an end point $b_0 = 480$. For each customer we let t_a and t_b represent the distance from this customer to node (50, 50) and node (0, 0), respectively. Then, we give a random number t_m which is uniformly distributed in the discrete interval $[t_a, b_0 - t_b - t_s]$ where t_s is the mean value of service time. Next, we generate the time windows for each customer as: $[t_m - 0.5t_l, t_m + 0.5t_l]$.

Three categories of instances are generated, including 30 customers and 5 caregivers (denoted by R3), 40 customers and 7 caregivers (R4), and 50 customers and 9 caregivers (R5). Each category includes 10 instances. For each test instance, the travel cost per unit time ct is 1.0, service cost per unit time cs_k is randomly chosen from {1.0, 0.8, 0.6}, and the penalty for one uncovered customer η is 1000.

All test instances in computational experiments are available on the website:

https://drive.google.com/file/d/1ppfHc0AZfb2HpJmvCEsxRpwEiLNE4f5b/view?usp=sharingz

4.2 Analysis of label algorithm

The label algorithm integrates the strategies of the inclusion of unreachable nodes, DSSR, and *ng*-route relaxation, represented by LA + Unreachable-*ng*-DSSR. The LA + Unreachable-*ng*-DSSR obtains *ng*-feasible columns which may contain cycles. Therefore, its lower bound is worse than that by the label algorithm with the inclusion of the unreachable nodes and DSSR (called the LA + Unreachable-DSSR).

To demonstrate its performance, the LA + Unreachable-*ng*-DSSR is compared with the LA + Unreachable-DSSR by testing instances at service levels of 75% and 95%. The runtime at the root node and total runtime of the proposed algorithm are used as indicators for evaluation. The size of original neighborhoods Δ_{ng} is determined among 3, 5, and 7. Tables 1 and 2 outline the average runtime (in s) at the root node (Column "Root") and total runtime (Column "Total") by the LA + Unreachable-DSSR and LA + Unreachable-*ng*-DSSR with different values of Δ_{ng}. The smallest values among different algorithms are bold in each row of the tables. As shown in Tables 1 and 2, except for the mean runtimes at the root node in the instances of R5 when β is 75%, the algorithm with the *ng*-route relaxation

Table 1 Summary of average runtimes of instances by all algorithms when β is 75.0%

Instance	LA + unreachable-DSSR		LA + unreachable-*ng*-DSSR					
			$\Delta_{ng} = 3$		$\Delta_{ng} = 5$		$\Delta_{ng} = 7$	
	Root	Total	Root	Total	Root	Total	Root	Total
R3	180.91	1003.02	175.85	862.05	179.57	**832.04**	**168.03**	843.64
R4	245.91	1310.04	233.70	1064.44	**231.54**	**1000.78**	235.55	1030.11
R5	**298.39**	4417.90	302.32	4278.25	299.07	**4085.72**	312.26	4134.12
Ave.	241.74	2243.65	237.29	2068.24	**236.73**	**1972.85**	238.61	2002.62

Table 2 Summary of average runtimes of instances by all algorithms when β is 95.0%

Instance	LA + unreachable-DSSR		LA + unreachable-*ng*-DSSR					
			$\Delta_{ng}=3$		$\Delta_{ng}=5$		$\Delta_{ng}=7$	
	Root	Total	Root	Total	Root	Total	Root	Total
R3	109.22	386.83	**107.22**	**338.94**	121.60	370.18	116.70	370.14
R4	196.07	1319.62	169.52	1219.00	**159.68**	**1175.05**	163.93	1218.75
R5	212.56	5578.31	211.61	5141.05	**200.71**	**4673.70**	212.79	4855.28
Ave.	172.62	2428.25	162.78	2233.00	**160.67**	**2072.98**	164.47	2148.06

(i.e., the LA + Unreachable-*ng*-DSSR) outperforms its counterpart without such relaxation (i.e., the LA + Unreachable-DSSR) in other test instances. The size of the original neighborhood Δ_{ng} is the only parameter of the proposed algorithm. When Δ_{ng} is set to different values, the algorithm with *ng*-route relaxation behaves differently. In terms of average runtimes at the root node and total runtimes, the absolute deviations among different values of Δ_{ng} are small and the algorithm performs better when Δ_{ng} equals five. Therefore, Δ_{ng} is set to five in the proposed algorithm.

4.3 Results of instances

Tables 3 and 4 demonstrate the results with the service levels of 75.0% and 95.0%, including the upper bound (RUB), lower bound (RLB), relative gap (RGap) between RUB and RLB, and runtime at the root node. The details of the optimal solution (i.e., the objective value, operational cost including expected travel cost and service cost, and number of uncovered customers), and the runtime in the search tree of instances are also presented in these two tables. The RUB is obtained as described in Sect. 3.3.1, and the RGap equals (RUB-RLB)/RUB.

Table 3 presents the results of instances when the service level of customers is 75.0%. For the instances with 30 customers, the proposed algorithm solves the RMP at the root node within 179.27 s and obtains the optimal solution in 832.04 s. For the instances with 40 customers, the algorithm spends about 231.54 and 1000.78 s on average to obtain the lower bound at the root node and the optimal solution, respectively. As the size of the instances increases, the runtime increases dramatically. For the largest instances with 50 customers, the average runtimes for optimally solving the RMP at the root node and the proposed problem are 299.07 and 4085.72 s, respectively. The results of the instances when the service level of customers is 95.0% are outlined in Table 4. These results lend evidence to the favorable performance of the proposed B&P algorithm.

The problem with service level 75.0% can be regarded as the relaxation of that with service level 95.0%. Therefore, the results of the instances with different service levels are compared to observe the algorithmic and model behaviors. In terms of the algorithmic behavior, the prompt solving of an instance at one service level by the algorithm may not guarantee the same speed at another service level. Taking R502 as an example, when the service level is 95.0%, the algorithm solves it without

Table 3 Results of instances when β is 75.0%

Instance	Root node				Search tree			
					Optimal solution			CPU(s)
	RUB	RLB	RGap%	CPU(s)	Objective value	Operational cost	Uncovered customers	
R301	4615.20	3798.00	17.71	171.99	3852.00	1852.00	2	567.17
R302	3737.60	3151.60	15.68	334.69	3667.40	1667.40	2	1562.65
R303	5420.20	5420.20	0.00	90.60	5420.20	1420.20	4	90.60
R304	3549.60	3549.60	0.00	159.84	3549.60	1549.60	2	159.84
R305	4521.20	3741.10	17.25	176.46	3782.00	1782.00	2	549.01
R306	4496.40	4156.40	7.56	139.61	4496.40	1496.40	3	381.84
R307	5578.00	5558.53	0.35	158.02	5578.00	1578.00	4	282.19
R308	6443.60	5813.10	9.78	199.38	6443.60	1443.60	5	1892.67
R309	5590.60	5239.80	6.27	165.86	5590.60	1590.60	4	1940.13
R310	6598.80	6294.80	4.61	199.23	6598.80	1598.80	5	894.31
R401	8979.60	8188.87	8.81	181.84	8308.00	2308.00	6	1014.35
R402	4216.60	4062.05	3.67	210.92	4216.60	2216.60	2	325.00
R403	3315.60	3087.00	6.89	243.38	3315.60	2315.60	1	499.53
R404	5399.80	5012.24	7.18	364.04	5352.80	2352.80	3	1524.60
R405	6189.20	6122.60	1.08	240.36	6189.20	2189.20	4	791.82
R406	5395.00	5395.00	0.00	134.64	5395.00	2395.00	3	134.64
R407	5446.40	5072.70	6.86	197.58	5446.40	2446.40	3	1648.23
R408	5182.00	5182.00	0.00	199.44	5182.00	2182.00	3	199.44
R409	7431.20	7385.60	0.61	195.58	7425.60	2425.60	5	457.14
R410	5209.00	4463.56	14.31	347.63	5209.00	2209.00	3	3413.01
R501	6872.00	6219.77	9.49	331.46	6849.60	2849.60	4	9782.83
R502	8678.80	7387.25	14.88	361.33	7831.00	2831.00	5	3397.96
R503	5222.20	5203.80	0.35	282.36	5222.20	3222.20	2	430.17
R504	6739.00	6390.50	5.17	206.69	6739.00	2739.00	4	2130.38
R505	5011.80	4991.90	0.40	340.95	5011.80	3011.80	2	803.75
R506	7093.20	6449.93	9.07	294.77	7014.60	3014.60	4	3784.72
R507	6913.00	6536.80	5.44	363.38	6913.00	2913.00	4	6333.76
R508	8162.60	7884.04	3.41	201.55	8142.60	3142.60	5	6160.34
R509	5152.40	4885.20	5.19	270.63	5152.40	3152.40	2	647.78
R510	7851.00	6575.83	16.24	337.63	7019.60	3019.60	4	7385.51

branching in 197.16 s, whereas, when the level is 75.0%, it spends a markedly longer duration of 3397.96 s. In terms of the model behavior, the objective values of the instances with service level 95.0% are larger than or equal to those of instances with service level 75.0% shown in Tables 3 and 4. These results implicitly validate the effectiveness of the approximation approach.

Table 4 Results of instances when β is 95.0%

Instance	Root node				Search tree			
					Optimal solution			CPU (s)
	RUB	RLB	RGap%	CPU (s)	Objective value	Operational cost	Uncovered customers	
R301	5547.80	5159.40	7.00	150.10	5538.40	1538.40	4	1041.68
R302	5480.20	4516.50	17.59	236.93	4538.40	1538.40	3	446.12
R303	7297.60	7297.60	0.00	83.14	7297.60	1297.60	6	83.14
R304	5260.20	4827.10	8.23	72.57	5257.40	1257.40	4	229.84
R305	5405.40	5216.03	3.50	124.74	5405.40	1405.40	4	423.88
R306	4496.40	4496.40	0.00	73.20	4496.40	1496.40	3	73.20
R307	6636.20	6527.28	1.64	159.00	6634.00	1634.00	5	877.10
R308	7353.60	7353.60	0.00	112.15	7353.60	1353.60	6	112.15
R309	6505.80	6295.33	3.24	116.73	6505.80	1505.80	5	327.22
R310	6709.00	6709.00	0.00	87.45	6709.00	1709.00	5	87.45
R401	9036.60	9014.87	0.24	150.48	9036.60	2036.60	7	415.93
R402	5974.80	5169.02	13.49	143.45	5952.60	1952.60	4	3669.53
R403	5126.80	5126.80	0.00	146.42	5126.80	2126.80	3	146.42
R404	7038.40	6913.55	1.77	173.40	7038.40	2038.40	5	610.62
R405	7067.80	6882.55	2.62	291.91	7067.80	2067.80	5	687.66
R406	8222.20	8043.90	2.17	110.27	8222.20	2222.20	6	1624.75
R407	6264.20	6264.20	0.00	118.13	6264.20	2264.20	4	118.13
R408	6963.80	6642.40	4.62	130.16	6962.80	1962.80	5	1314.82
R409	10,043.00	9485.87	5.55	139.18	10,013.00	2013.00	8	2793.58
R410	6083.40	5841.87	3.97	193.45	6083.40	2083.40	4	369.02
R501	8725.20	8363.50	4.15	170.58	8725.20	2725.20	6	2860.40
R502	9797.40	9797.40	0.00	197.16	9797.40	2797.40	7	197.16
R503	7047.80	6899.23	2.11	229.10	7047.80	3047.80	4	1130.17
R504	8653.60	7923.40	8.44	203.88	8653.60	2653.60	6	9602.82
R505	7742.40	7226.00	6.67	202.76	7742.40	2742.40	5	6629.19
R506	9788.40	9397.40	3.99	132.44	9788.40	2788.40	7	4792.23
R507	8898.60	8235.60	7.45	215.33	8898.60	2898.60	6	5972.70
R508	10,744.00	10,363.50	3.54	197.65	10,744.00	2744.00	8	5591.22
R509	8051.00	7502.10	6.82	209.04	7912.00	2912.00	5	2271.03
R510	9669.60	9192.00	4.94	249.17	9669.60	2669.60	7	7690.10

4.4 Analysis of the effect of stochastic times on solutions

To emphasize the necessity of considering stochastic times when designing service routes, the solutions under deterministic and stochastic situations are compared. In the deterministic scenario, the service and travel times are assumed to be constant and equal to their mean value, whereas, in the stochastic scenario, they are normally distributed agreeing with the assumptions of the study outlined before. Randomly selecting three test instances, R301, R403, and R503 as examples, we obtain the solutions under the deterministic and stochastic situations by the B&P algorithm. Under the assumed distribution of the service and travelling time, the stochastic simulation is performed to calculate the actual service levels of customers, the number of simulation iterations being 100,000.

Tables 5 and 6 report the solutions in the deterministic and stochastic situations and the number of customers (not including unvisited customers) who are below the required service levels ($\beta \leq 75\%$, 80%, 85%, 90%, and 95.0%) in the stochastic simulation. The actual service levels of the customers in the deterministic and stochastic situations are compared with the maximum and minimum service levels of 75.0% and 95.0% in the stochastic simulation in Figs. 2, 3, 4. In such Tables and Figures, "Deterministic" represents the solution in the deterministic situation; "Stochastic-75" denotes the solution in the stochastic situation with the predefined service level of 75.0%, and "Service Level-75" is the value of the predefined service level. The points on the horizontal axis indicate the unvisited customers.

The data of Tables 5 and 6 and Figs. 2, 3 and 4 shows that the objective value of the solution in the deterministic situation is much lower than that in the stochastic situations with different predefined service levels, and that the former serves more customers than the latter. These results demonstrate that, for the solutions in the stochastic situation, the customers (except for the unvisited ones) can reach the predefined service levels, demonstrating the effectiveness of the discrete approximation approach. Conversely, for the solutions in the deterministic situation, more and more customers cannot obtain the increasing predetermined service levels, demonstrating that such solutions may be infeasible in practice. Therefore, it is necessary for HHC managers to consider stochastic times when planning the routes of caregivers.

4.5 Comparing "soft time window" and "hard time window"

As stated in Sect. 1, the algorithm proposed in Yuan et al. (2015) considers the soft time window for each customer, instead of hard time windows and chance constraints. Thus, their approach cannot be applied in this paper. However, we believe it is still interesting to compare the solutions from two different approaches: one considers the stochastic travel and service times and *soft time window* for each customer, like the algorithm of Yuan et al. (2015); the other one considers the *hard time*

Table 5 Results of solutions in deterministic and stochastic situations

Instance	Deterministic						
	Objective value	Customers					
		Unvisited	$\beta \leq 75.0\%$	$\beta \leq 80.0\%$	$\beta \leq 85.0\%$	$\beta \leq 90.0\%$	$\beta \leq 95.0\%$
R301	3665.80	2	5	5	5	6	6
R403	2228.20	0	6	7	7	10	13
R503	3831.20	1	5	7	9	11	12

Table 6 Results of solutions in deterministic and stochastic situations

Instance	Stochastic—75		Stochastic—80		Stochastic—85		Stochastic—90		Stochastic—95	
	Objective value	Unvisited customers	Objective value	Unvisited customers	Objective value	Unvisited customers	Unvisited customers	Unvisited customers	Unvisited customers	Unvisited custom-ers
R301	3852.00	2	4615.20	3	4615.20	3	5517.60	4	5538.40	4
R403	3315.60	1	4338.90	3	4808.40	3	4808.40	3	5126.80	3
R503	5222.20	2	6470.40	4	6470.40	4	6470.40	4	7047.80	4

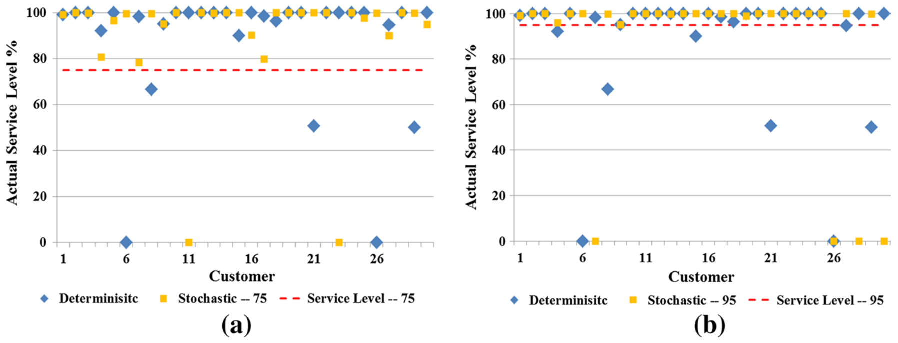

Fig. 2 Comparison of actual service levels with R301 in stochastic simulation. **a** Service level 75.0%. **b** Service level 95.0%

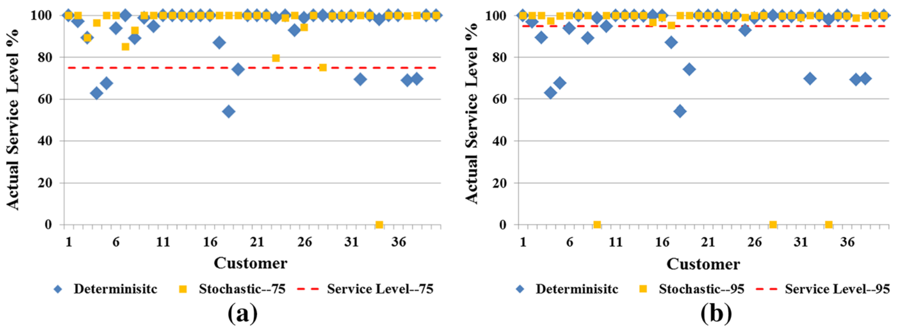

Fig. 3 Comparison of actual service levels with R403 in stochastic simulation. **a** Service level 75.0%. **b** Service level 95.0%

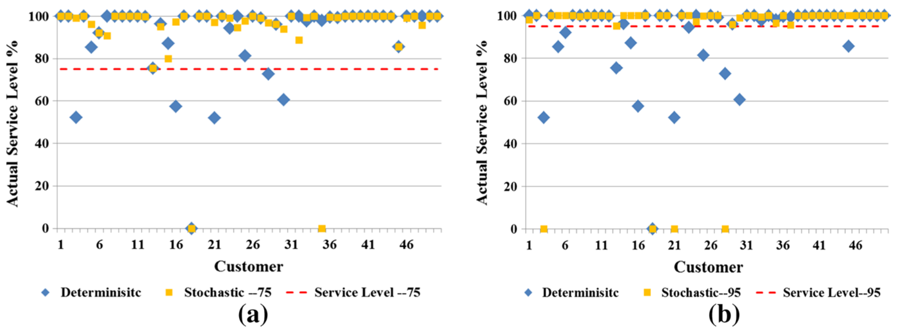

Fig. 4 Comparison of actual service levels with R503 in stochastic simulation. **a** Service level 75.0%. **b** Service level 95.0%

window just like the algorithm in this paper. Therefore, we generate the test instance as follows.

First, we generate one test instance based on one Solomon VRPTW benchmark instance, which has 30 customers and 5 caregivers. We suppose that all caregivers start from one node (company/depot) and end the route at this node. Note that a basic assumption of Yuan et al. (2015) is that the travel times are deterministic. For the purposes of comparison between two algorithms, in this test instance travel times are also set to be stochastic when it is solved by the approach of Yuan, Liu, and Jiang (Yuan et al. 2015). Then, the main differences of the two approaches are twofold, i.e., the *time windows* and *chance constraint*. Yuan et al. (2015), the soft time windows constraints are adopted and a penalty occurs for the service time later than the deadline of time window. In this paper, we use hard time windows and considers a chance constraint that each customer can be visited in her/his time window with a probability no smaller than 95%. Due to the chance constraint, some customers may not be visited resulting in a relatively large penalty cost, i.e., 1000. We obtain the solutions and total costs of these two approaches. The solutions of the two approaches are illustrated in Fig. 5. Clearly, the routes of these two solutions are different. The solution of this paper guarantees a high probability that each customer is visited in the time window except two customers (No 2 and 19). The solution of algorithm in Yuan et al. (2015) allows all the customers to be served due to the soft time windows. However, the on-time-visit to customers cannot be guaranteed in such solution. More than 40% customers cannot be serviced in their time windows.

5 Conclusions and future work

This paper addresses a chance-constrained home-caregiver scheduling and routing problem with stochastic travel and service times. In this problem, each caregiver daily route starts and ends his/her home and visits each customer location at most once. Each customer has a hard time window, which truncates the distributions of arrival times and starting service times, and lets the closed formula for the recourse cost cannot be obtained. To guarantee the service quality, the chance

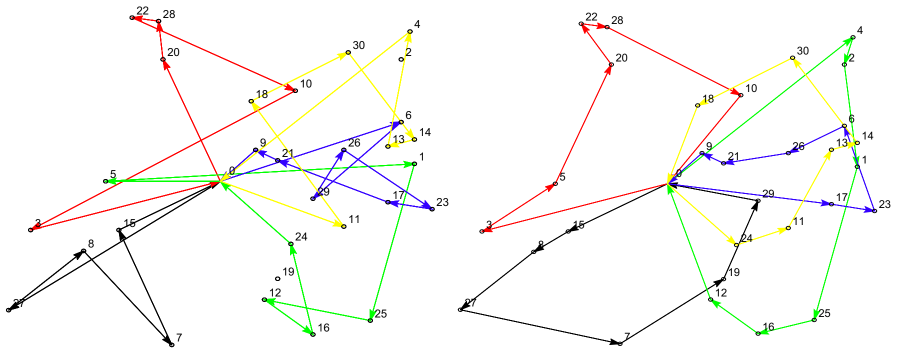

Fig. 5 Solutions with "soft time window" and "hard time window" constraints

constraint is imposed to ensure the probability that one customer can be visited in a special time window. To solve this hard problem, a branch-and-price algorithm with a discrete approximation approach is developed, in which the label algorithm with cutting-edge acceleration strategies and hierarchical branching scheme are devised. An extensive analysis using three sets of test instances of different sizes is performed to illustrate the efficiency of the proposed algorithm. Numerical results validate the effectiveness of the proposed algorithm. Meanwhile, the results highlight the importance of considering the stochastic travel times and service times in the HHC caregiver scheduling and routing problem.

In the future, meta-heuristics, such as the genetic algorithm, TS, and large neighborhood search, can be developed to solve larger instances. The algorithm proposed in this paper can also be applied in similar VRPs with chance constraints.

Acknowledgments This work is supported by National Natural Science Foundation of China (71672112).

References

Athanasopoulos T, Minis I (2013) Efficient techniques for the multi-period vehicle routing problem with time windows within a branch and price framework. Ann Oper Res 206(1):1–22

Baldacci R, Mingozzi A, Roberti R (2011) New route relaxation and pricing strategies for the vehicle routing problem. Oper Res 59(5):1269–1283

Bertels S, Fahle T (2006) A hybrid setup for a hybrid scenario: combining heuristics for the home health care problem. Comput Oper Res 33(10):2866–2890

Boland N, Dethridge J, Dumitrescu I (2006) Accelerated label setting algorithms for the elementary resource constrained shortest path problem. Oper Res Lett 34(1):58–68

Cissé M et al (2017) OR problems related to Home Health Care: a review of relevant routing and scheduling problems. Oper Res Health Care 13–14:1–22

Dayarian I et al (2015a) A branch-and-price approach for a multi-period vehicle routing problem. Comput Oper Res 55:167–184

Dayarian I et al (2015b) A column generation approach for a multi-attribute vehicle routing problem. Eur J Oper Res 241(3):888–906

Desaulniers G, Lavigne J, Soumis F (1998) Multi-depot vehicle scheduling problems with time windows and waiting costs. Eur J Oper Res 111(3):479–494

Ehmke JF, Campbell AM, Urban TL (2015) Ensuring service levels in routing problems with time windows and stochastic travel times. Eur J Oper Res 240(2):539–550

Eveborn P, Flisberg P, Rönnqvist M (2006) Laps Care—an operational system for staff planning of home care. Eur J Oper Res 171(3):962–976

Feillet D (2010) A tutorial on column generation and branch-and-price for vehicle routing problems. 4OR 8(4):407–424

Fikar C, Hirsch P (2017) Home health care routing and scheduling: a review. Comput Oper Res 77:86–95

Lanzarone E, Matta A (2014) Robust nurse-to-patient assignment in home care services to minimize overtimes under continuity of care. Oper Res Health Care 3(2):48–58

Lanzarone E, Matta A, Sahin E (2012) Operations management applied to home care services: the problem of assigning human resources to patients. IEEE Trans Syst Man Cybern Part A Syst Hum 42(6):1346–1363

Li X, Tian P, Leung SCH (2010) Vehicle routing problems with time windows and stochastic travel and service times: models and algorithm. Int J Prod Econ 125(1):137–145

Liberatore F, Righini G, Salani M (2011) A column generation algorithm for the vehicle routing problem with soft time windows. 4OR 9(1):49–82

Liu R, Yuan B, Jiang Z (2017) Mathematical model and exact algorithm for the home care worker scheduling and routing problem with lunch break requirements. Int J Prod Res 55(2):558–575

Nickel S, Schröder M, Steeg J (2012) Mid-term and short-term planning support for home health care services. Eur J Oper Res 219(3):574–587

OECD (2017) Health at a Glance 2017: OECD Indicators. OECD Publishing, Paris. https://doi.org/10.1787/health_glance-2017-en

Polacek M et al (2004) A variable neighborhood search for the multi depot vehicle routing problem with time windows. J Heurist 10(6):613–627

Rasmussen MS et al (2012) The home care crew scheduling problem: preference-based visit clustering and temporal dependencies. Eur J Oper Res 219(3):598–610

Shao Y, Bard JF, Jarrah AI (2012) The therapist routing and scheduling problem. IIE Trans 44(10):868–893

Trautsamwieser A, Hirsch P (2011) Optimization of daily scheduling for home health care services. J Appl Oper Res 3(3):124–136

Trautsamwieser A, Gronalt M, Hirsch P (2011) Securing home health care in times of natural disasters. OR Spectr 33(3):787–813

Vidal T et al (2013) A hybrid genetic algorithm with adaptive diversity management for a large class of vehicle routing problems with time-windows. Comput Oper Res 40(1):475–489

Yalçındağ S, Matta A (2017). A decomposition approach for the home health care problem with time windows. In: International conference on health care systems engineering. Springer, Berlin

Yalçındağ S et al (2016) The patient assignment problem in home health care: using a data-driven method to estimate the travel times of care givers. Flex Serv Manuf J 28(1–2):304–335

Yuan B, Liu R, Jiang Z (2015) A branch-and-price algorithm for the home health care scheduling and routing problem with stochastic service times and skill requirements. Int J Prod Res 53(24):7450–7464

Zachariadis EE, Tarantilis CD, Kiranoudis CT (2015) The load-dependent vehicle routing problem and its pick-up and delivery extension. Transp Res Part B Methodol 71:158–181

Zhang J, Lam WH, Chen BY (2013) A stochastic vehicle routing problem with travel time uncertainty: trade-off between cost and customer service. Netw Spat Econ 13(4):471–496

Ran Liu is an Associate Professor at Department of Industrial Engineering & Management, in Shanghai Jiao Tong University, Shanghai, China. He received his Ph.D. from Shanghai Jiao Tong University, Shanghai, China, in 2012. His research interests include combinational optimization, algorithm design and analysis.

Biao Yuan received his Ph.D. from Shanghai Jiao Tong University, Shanghai, China, in 2017. His research interests include combinational optimization, operation management in manufacturing and health-care systems.

Zhibin Jiang received his Ph.D. degree in manufacturing engineering and engineering management from City University of Hong Kong, Hong Kong, China, in 1999. He is Changjiang Scholar Chair Professor of MOE, China, and Distinguished Professor of Shanghai Jiao Tong University, Shanghai, China. His research interests include discrete-event modeling and simulation, operation management in manufacturing industries and health-care systems.