European Journal of Operational Research 219 (2012) 557–563

Contents lists available at SciVerse ScienceDirect

European Journal of Operational Research

journal homepage: www.elsevier.com/locate/ejor

Optimal patient and personnel scheduling policies for care-at-home service facilities

P.M. Koeleman, S. Bhulai *, M. van Meersbergen

VU University Amsterdam, Faculty of Sciences, De Boelelaan 1081a, 1081 HV, Amsterdam, The Netherlands

ARTICLE INFO

Article history:
Available online 9 November 2011

Keywords:
OR in health services
Care-at-home facilities
Markov decision processes
Personnel planning
Scheduling

ABSTRACT

In this paper we study the problem of personnel planning in care-at-home facilities. We model the system as a Markov decision process, which leads to a high-dimensional control problem. We study monotonicity properties of the system and derive structural results for the optimal policy. Based on these insights, we propose a trunk reservation heuristic to control the system. We provide numerical evidence that the heuristic yields close to optimal performance, and scales well for large problem instances.

1. Introduction

Advances in health care have led to an increasing number of elderly people in society who want to continue living in their own homes while needing medical care and care-at-home services (e.g., housekeeping and personal care). This trend has led to a situation in which home care providers are faced with a larger number of patients. At the same time, home care providers have to provide care-at-home services with fewer resources because of changing organization and finance structures and increased competition. Therefore, efficient workforce management is essential to provide a high quality of service against low operational costs.

In practice, efficient workforce management is hard to achieve. The care-at-home sector typically has a very unpredictable demand for service. Moreover, the duration of the service is highly volatile. This creates a tension between the size of the workforce and the operational costs. On the one hand, having a lot of home care personnel leads to a very good service quality: all demand can be satisfied directly and no patient is turned down. However, the operational costs are high and much of the personnel will have a lot of idle time and low productivity. On the other hand, having too few personnel leads to low operational costs, but also a deterioration in quality of service.

The personnel planning problem is not unique to the care-at-home sector. Many other service providers are faced with this challenging problem, e.g., call centers (Aksin et al. [1] and Gans et al. [12]), health care (Burke and Petrovic [5]), and public transport (Petrovic and Berghe [14]). Ernst et al. [9] present a comprehensive collection of some 700 papers on personnel scheduling in different application areas. The aforementioned surveys show that most of the literature on personnel planning in health care systems deal with appointment scheduling, shift scheduling, cyclic rostering, hospital admission and bed planning. This is mostly done in a deterministic setting for which mathematical and integer programming methods, set covering and partitioning, local and tabu search techniques are used. The papers that deal with stochastic health care systems (mostly, appointment scheduling and hospital planning) use local search, genetic programming, simulation techniques, Markov decision theory, and queueing theory (see, e.g., the survey paper [9] and the special issues [5,14] with the references therein).

The care-at-home sector is a rather unique part of the health care industry that has received little attention in the literature. Moreover, it faces challenges that the other industries do not. First, patients have very specialized needs for home care so that home care providers are faced with a large number of very different patient and service profiles. Second, a patient may require a number of hours home care in a week, but needs to receive that for several weeks consecutively. Hence, enough personnel capacity has to be available so that a patient will continue to receive home care once admitted. These two distinguishing features in a stochastic setting add additional complexity to the personnel planning problem which makes many of the modeling and solution techniques intractable.

The literature on workforce management in a home care personnel planning setting can be categorized into two groups: the first group describes the imminent shortage of skilled nurses and other health workers in the coming years due to the aging population, and the important factors for organizations in attracting new staff and retaining their current staff. Ellenbecker [8] mentions having a realistic workload and a stable schedule as important factors in keeping personnel retention at low levels. Flynn and Deatrick [11] also mention having a realistic workload, adequate

* Corresponding author.
E-mail addresses: paulien@few.vu.nl (P.M. Koeleman), s.bhulai@vu.nl (S. Bhulai), maarten@few.vu.nl (M. van Meersbergen).

0377-2217/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.ejor.2011.10.046

staffing levels, and scheduled days off as important factors for the nurses. The second group of articles focuses on daily scheduling and routing of nurses. Cheng and Rich [6] models this problem as a vehicle routing problem with time windows using a mixed integer program. Bertels and Fahle [3] study the rostering and routing problem simultaneously. They choose not to obtain optimal solutions, because of the large computation times. Instead, they present several good solutions, by modeling different requirements by hard and soft constraints. Eveborn et al. [10] describe a decision support system for planning home care routes. This lets the user attach priorities to different aspects of the solution, such as travel times and preferred staff members to visit certain patients. The literature on home care personnel planning is largely deterministic in nature and does not deal with the stochastic nature of the demand and service required that is perceived in practice.

In this paper, we aim to provide a model to deal with the personnel planning problem in care-at-home service facilities in a stochastic setting. We cast the problem as a Markov decision problem as this is sufficiently flexible to deal with different patient and service profiles in a decision framework while is still remains sufficiently tractable (see Powell [15] for issues on modeling and computation). Next, we study the monotonicity properties of the Markov decision model. These results are used to derive optimal patient admission policies, given the demand for service. Moreover, we study the performance of these policies so that the size of the workforce can be determined. The model provides a first step in the workforce planning process; after the capacity has been derived by the model, one needs to make rosters in which individual personnel members are assigned tasks. Moreover, one needs to make efficient routes from the care-at-home facility to the patients. These problems are not taken into account in our model, since many algorithms and software packages already exist to deal with this.

The paper is structured as follows. In Section 2 we provide a rigorous mathematical description of the problem. In Section 3 we analyze the case both with and without waiting, and derive (nearly) optimal policies for them. We then illustrate the results through numerical experiments and assess the quality of the policies in Section 4. Finally, we conclude the paper with Section 5 in which we discuss the conclusions and topics for further research.

2. Problem formulation

We consider a care-at-home facility at which patients arrive for home care services according to a Poisson process with rate λ. There are N home care employees that are available for a number of hours per week (depending on their contract) to provide home care to the patients. The home care service that each patient requires is for a duration of several hours a week for several weeks consecutively. To turn this requirement into a tractable model formulation, we represent the workforce not as the number of home care employees, but as the number of service hours S available per week determined by the N employees. Then, the service duration of a patient can be modeled by assuming that there are K classes of patients, such that an arriving patient belongs to class $k \in K$ with probability p_k and needs home care for c_k hours per week for a duration of weeks that is distributed according to an exponential distribution with parameter μ_k.

When a request for home care arrives at the care-at-home facility, the facility knows (e.g., based on the patient's medical records) to which class the patient belongs. Denote by $\vec{x} = (x_1, \ldots, x_K)$ the state of the home care employees, i.e., x_k is the number of patients of class k in service for $k = 1,\ldots,K$. When the state vector $\vec{x}$ is given, then the spare capacity in the system is given by $\text{cap}(\vec{x}) = S - \sum_{k=1}^{K} c_k \cdot x_k$. Now, let us suppose that a patient from class k arrives. Then the facility has several options to deal with this request. First, consider the scenario in which there is not sufficient service capacity available (thus, $\text{cap}(\vec{x}) < c_k$). Then, the facility can reject the request, but can also decide to admit the patient so that the patient is put on a waiting list for home care. For this purpose, let $\vec{q} = (q_1, \ldots, q_K)$ denote the number of patients of each class that are on the waiting list. In case there is sufficient capacity (i.e., $\text{cap}(\vec{x}) \geqslant c_k$), the facility has three options. First, it can again reject the request, because it expects other arrivals of patients that might conflict with the current request (this can happen, especially, when c_k is large), it can put the patient on the waiting list, or it can admit the patient immediately so that its service can start without delay. We assume that the waiting list can only hold B patients. If a patient is required to wait while the waiting list already has B patients, then the patient is rejected anyway.

We assume that the system is subject to rejection costs r_k for request k with $k = 1,\ldots,K$, and that there are costs for having a patient in the system (either waiting or in service). We are interested in finding a policy that balances the rejection costs and the average number of patients in the system by minimizing the joint cost function. To this purpose, we cast the problem of optimal decision making in this system in the framework of Markov decision theory. To this purpose, we denote the state of the care-at-home facility by $(\vec{x}, \vec{q})$ with state space $\mathcal{S} = \{(\vec{x}, \vec{q}) | \sum_{k=1}^{K} c_k \cdot x_k \leqslant S, \sum_{k=1}^{K} q_k \leqslant B\}$. Let $V(\vec{x}, \vec{q})$ be a real-valued function defined on the state space $\mathcal{S}$. This function will play the role of the relative value function, i.e., the asymptotic difference in total costs that results from starting the process in state $(\vec{x}, \vec{q})$ instead of some reference state. From this point on, we will take without loss of generality the empty system, i.e., $(\vec{x}, \vec{q}) = (\vec{0}, \vec{0})$, as reference state. The relative value function is also important for deriving the optimal actions: when a patient of class k arrives, the optimal action can be determined by the minimizing action in H_a given by:

$$H_a(\vec{x}, \vec{q}, k) = \begin{cases} V(\vec{x}, \vec{q}) + r_k, & \text{if } \text{cap}(\vec{x}) < c_k, \sum_{k=1}^{K} q_k = B, \\ \min\{V(\vec{x}, \vec{q}) + r_k, V(\vec{x}, \vec{q} + e_k)\}, & \text{if } \text{cap}(\vec{x}) < c_k, \sum_{k=1}^{K} q_k < B, \\ \min\{V(\vec{x}, \vec{q}) + r_k, V(\vec{x} + e_k, \vec{q})\}, & \text{if } \text{cap}(\vec{x}) \geqslant c_k, \sum_{k=1}^{K} q_k = B, \\ \min\{V(\vec{x}, \vec{q}) + r_k, V(\vec{x}, \vec{q} + e_k), V(\vec{x} + e_k, \vec{q})\}, & \text{otherwise}, \end{cases}$$

with e_k the vector with zeros and a one at the k-th entry. The terms $V(\vec{x}, \vec{q}) + r_k, V(\vec{x}, \vec{q} + e_k)$, and $V(\vec{x} + e_k, \vec{q})$ represent the value of rejecting, delaying, and admitting a patient, respectively. A similar result holds when a patient no longer requires service so that home care capacity becomes available. In that case, a patient that is delayed can be taken into service. We denote by H_d the term that deals with the actions after a departure of a patient given by:

$$H_d(\vec{x}, \vec{q}) = \min_{(\vec{x}', \vec{q}') \in \mathcal{S}} \left\{ V(\vec{x}', \vec{q}') | x'_k \geqslant x_k \text{ for } k = 1, \ldots, K, \sum_{k=1}^{K} (x_k + q_k) = \sum_{k=1}^{K} (x'_k + q'_k) \right\}.$$

Note the H_d allows multiple patients to be admitted into the system, since it could happen that a patient with a large service utilization has left. This event could free up capacity for multiple patients with a relatively small demand for services. The first condition between the brackets ensures that *new* patients are admitted, whereas the second condition makes sure that the total number of patients in both situations are equal so that no patients are rejected.

Thus, to fully control the system one needs a patient admission policy and a personnel scheduling policy. These policies can be determined if the relative value function is known. Let g denote the long-term average cost in the system. For simplicity we assume that $\lambda + S \max_{\{k=1,\dots,K\}}\{\mu_k\} < 1$; without loss of generality, we can always achieve this by scaling. Uniformizing is equivalent to adding dummy transitions (from a state to itself) such that the rate out of each state is equal to 1; then we can consider the arrival and service rates to be transition probabilities. By doing so, the relative value function can be determined by solving the optimality equation (in vector notation) $g + V = TV$, where T is the dynamic programming operator acting on $V(\vec{x},\vec{q})$ defined as follows

$$TV(\vec{x},\vec{q}) = \sum_{k=1}^{K}(x_k + q_k) + \sum_{k=1}^{K}\lambda p_k H_a(\vec{x},\vec{q},k) + \sum_{k=1}^{K} x_k\mu_k H_d(\vec{x} - e_k,\vec{q}) + \left(1 - \lambda - \sum_{k=1}^{K} x_k\mu_k\right)V(\vec{x},\vec{q}).$$

The first term in the equality counts the number of patients in service and on the waiting list. The second term models the arrivals of patients and the optimal admission policy. The third term deals with the situation in which a home care service ends and a patient departs the system. The last term is the dummy term that follows from uniformization of the system.

The optimality equation $g + V = TV$ is hard to solve analytically in practice. Alternatively, the optimal actions can also be obtained by recursively defining $V_{i+1} = TV_i$, for $i = 0,\dots$ and arbitrary V_0. For $i \to \infty$, the maximizing actions converge to the optimal ones (for existence and convergence of solutions and optimal policies we refer to Puterman [16]). Consequently, when V is known, we can restrict our attention to the function H_a and H_d to obtain the optimal actions. In Section 3 we shall adopt this approach to prove monotonicity results of the value function, and in Section 4 we use this approach to numerically compute optimal policies.

3. Optimal scheduling policies

In this section we will study the optimal scheduling policies for the care-at-home problem. We will distinguish between two cases: one with no waiting room for arriving patients, and one with waiting room. The case without waiting room has been studied in literature before in the setting of bandwidth allocation in telecommunication systems. We will provide an overview of the results in the literature for this case. The case with waiting room is inherently complex and has been given little attention in the literature. We will provide monotonicity results for this case and characterize part of the optimal policy.

3.1. The case with no waiting room

In this section we analyze the care-at-home model that has been described in the previous section. However, before studying the general model, we first study the case with no waiting, i.e., $B = 0$. The optimality equations then reduce to

$$g + V(\vec{x}) = \sum_{k=1}^{K} x_k + \sum_{k=1}^{K}\lambda p_k\left[\mathbb{1}_{\{\mathrm{cap}(\vec{x})<c_k\}}[V(\vec{x}) + r_k] + \mathbb{1}_{\{\ \mathrm{cap}(\vec{x})\geqslant c_k\}}\min\{V(\vec{x}) + r_k, V(\vec{x} + e_k)\}\right] + \sum_{k=1}^{K} x_k\mu_k V(\vec{x} - e_k) + \left(1 - \lambda - \sum_{k=1}^{K} x_k\mu_k\right)V(\vec{x}).$$

In case the policy is always to accept when possible, then the system reduces to the multi-rate blocking model. This is an extension of the Erlang blocking model and has been well-studied in a telecommunications setting. In this setting there is a certain amount of bandwidth (home care service capacity) available that arriving Internet requests (patients) can use. The Internet requests require part of the bandwidth (the number of hours per week) for a certain duration (the number of weeks consecutively). The model under this policy has a product-form solution for its steady-state distribution. Thus, let $\rho_k = \lambda p_k/\mu_k$ for $k = 1,\dots,K$. Then the probability of being in state $\vec{x} \in \mathcal{S}$ is given by $\pi(\vec{x})$ and has the form

$$\pi(\vec{x}) = \pi(x_1,\dots,x_K) = \frac{1}{G}\prod_{k=1}^{K}\frac{\rho_k^{x_k}}{x_k!} \quad \text{with} \quad G = \sum_{\vec{x}\in\mathcal{S}}\prod_{k=1}^{K}\frac{\rho_k^{x_k}}{x_k!}.$$

Let S_k denote the subset of states in which a call of class k is admitted to the system, i.e., $S_k = \{\vec{x} \in \mathcal{X} | \mathrm{cap}(\vec{x}) \geqslant c_k\}$. Then the blocking probability of a call of class k is given by $B_k = 1 - \sum_{\vec{x}\in S_k}\pi(\vec{x})$. However, the numerical evaluation can be a problem and the Kaufman–Robert recursion alleviates this problem. The long-term average cost g can then be efficiently calculated by $g = \sum_{k=1}^{K} B_k r_k$. The model with $B = 0$ that we study is an extension of the multi-rate blocking model. Blocking a patient of class k brings with it a cost of r_k that can be different for each class. The optimal policy with these different cost rates is usually different than the policy used in the multi-rate blocking model.

In case $c_k \equiv 1$ and $\mu_k \equiv \mu$, it is known that trunk reservation is optimal (Miller [13]). However, in a more general setting, Ross and Tsang [17] showed that the trunk reservation policy is not optimal anymore. Under the assumption that $c_k \leqslant c_{k+1}$ and $\mu_k \geqslant \mu_{k+1}$, Altman et al. [2] derived a stochastic ordering of the patient classes such that priority is given to the patient class with the smallest index. If in addition the assumption that $r_k \leqslant R_{k+1}$ is made, then it can be shown that the trunk reservation policy is optimal again. The more general case with no assumptions could only be studied in a fluid model, in which the authors showed the optimality of trunk reservation.

3.2. The general case with a waiting room

In this section we treat the care-at-home model in which patients are allowed to wait as well. This case is significantly more difficult than the case with no waiting room. Unlike the case with no waiting room, there is very limited literature available on the care-at-home model with waiting room. Therefore, we start with some structural properties of the relative value function V, which gives us insight into the structure of the optimal policy. We start by showing that V is an increasing function in all of its components. This is formalized by the following lemma.

Lemma 3.1 (*increasingness*). *For all $(\vec{x},\vec{q}) \in \mathcal{S}$ and $(\vec{x} + e_k,\vec{q}) \in \mathcal{S}$ we have*

$$V(\vec{x} + e_k,\vec{q}) \geqslant V(\vec{x},\vec{q}),$$

for k = 1,...,K. Similarly, for all $(\vec{x},\vec{q}) \in \mathcal{S}$ and $(\vec{x},\vec{q} + e_k) \in \mathcal{S}$ we have

$$V(\vec{x},\vec{q} + e_k) \geqslant V(\vec{x},\vec{q}),$$

for k = 1,...,K.

Proof. The proof is by induction on n in V_n. Define $V_0(\vec{x},\vec{q}) = 0$ for all states $(\vec{x},\vec{q}) \in \mathcal{S}$. Then, clearly, $V_0(\vec{x},\vec{q})$ is increasing in all components of $\vec{x}$ and $\vec{q}$. Now, assume that the statement of the lemma holds for V_n for some $n \in \mathbb{N}$. Now, we prove that $V_{n+1}(\vec{x},\vec{q})$ satisfies the increasingness property as well. Therefore, fix $k \in \{1,\dots,K\}$ and assume that $(\vec{x} + e_k,\vec{q}) \in \mathcal{S}$, then

$$\begin{aligned}V_{n+1}(\vec{x}+e_k,\vec{q})-V_{n+1}(\vec{x},\vec{q}) = 1 &+ \sum_{j=1}^{K}\lambda p_j[H_a(\vec{x}+e_k,\vec{q},j)-H_a(\vec{x},\vec{q},j)]\\ &+\sum_{j=1}^{K}x_j\mu_j[H_d(\vec{x}+e_k-e_j,\vec{q})\\ &-H_d(\vec{x}-e_j,\vec{q})]+\mu_k H_d(\vec{x},\vec{q})\\ &+\left(1-\lambda-\sum_{j=1}^{K}(x_j+\mathbb{1}_{\{j=k\}})\mu_j\right)\\ &\times[V_n(\vec{x}+e_k,\vec{q})-V_n(\vec{x},\vec{q})]-\mu_k V_n(\vec{x},\vec{q}).\end{aligned}$$

The first term of the righthand-side equals 1, since that is exactly the difference in the number of patients in both systems. The second term deals with the arrivals. Note that the optimal action in $H_a(\vec{x}+e_k,\vec{q},j)$ can be used in $H_a(\vec{x},\vec{q},j)$ as well (possibly as a suboptimal action). In doing so, we get that $H_a(\vec{x}+e_k,\vec{q},j)-H_a(\vec{x},\vec{q},j)\geqslant 0$ due to the induction hypothesis. The same goes for the term dealing with actions after departures. The next term cancels the sixth term. For the fifth term the induction hypothesis directly applies.

Now assume that $(\vec{x},\vec{q}+e_k)\in\mathcal{S}$, then

$$\begin{aligned}V_{n+1}(\vec{x},\vec{q}+e_k)-V_{n+1}(\vec{x},\vec{q}) = 1 &+ \sum_{j=1}^{K}\lambda p_j[H_a(\vec{x},\vec{q}+e_k,j)-H_a(\vec{x},\vec{q},j)]\\ &+\sum_{j=1}^{K}x_j\mu_j[H_d(\vec{x}-e_j,\vec{q}+e_k)\\ &-H_d(\vec{x}-e_j,\vec{q})]+\left(1-\lambda-\sum_{j=1}^{K}x_j\mu_j\right)\\ &\times[V_n(\vec{x},\vec{q}+e_k)-V_n(\vec{x},\vec{q})].\end{aligned}$$

The first term of the righthand-side equals 1, since that is exactly the difference in the number of patients in both systems. The second term deals with the arrivals. Note that the optimal action in $H_a(\vec{x},\vec{q}+e_k,j)$ can be used in $H_a(\vec{x},\vec{q},j)$ as well (possibly as a suboptimal action). In doing so, we get that $H_a(\vec{x},\vec{q}+e_k,j)-H_a(\vec{x},\vec{q},j)\geqslant 0$ due to the induction hypothesis. A similar remark holds for the next term with only a slight difference. In case the optimal action in $H_d(\vec{x}-e_j,\vec{q}+e_k)$ serves $q_k + e_k$ patients of type k, then the (sub) optimal action in $H_d(\vec{x}-e_j,\vec{q})$ should serve q_k patients of type k. The resulting state will differ by 1 for x_k while q_k is equal and so the induction hypothesis will still apply. For the last term the induction hypothesis directly applies.

We conclude, by taking the limit of $n\to\infty$, that $V(\vec{x},\vec{q})$ is increasing in x_k and q_k for all $k = 1,\ldots,K$. □

For the special case with all service needs of all classes of patients equal, we can give a very simple rule that gives optimal results. This rule states that after a departure the patient with the smallest expected service time is taken into service. This minimizes the number of patients in the system, and thus also the total rejection and holding costs.

Theorem 3.1. *Assume that $c_k = 1$ for all $k = 1,\ldots, K$. Suppose that $\mu_k > \mu_{k+1}$. Then, the optimal scheduling policy schedules people with the smallest index first.*

Proof. Because the theorem only concerns actions taken just after a departure, only H_d needs to be taken into account. Also, because the service requirements are equal for all types of patients, only one new patient will be taken into service at the same time after a departure and no room needs to be saved for patients with large service requirements. So what we need to prove is that $V(\vec{x}+e_i,\vec{q}-e_i)\leqslant V(\vec{x}+e_j,\vec{q}-e_j)$ if $i < j$.

The proof is by induction on n in V_n. Define $V_0(\vec{x},\vec{q})=0$ for all states $(\vec{x},\vec{q})\in\mathcal{S}$. Then, clearly, $V(\vec{x}+e_i,\vec{q}-e_i)\leqslant V(\vec{x}+e_j,\vec{q}-e_j)$ holds for all values of i and j. Now, assume that the statement of the theorem holds for V_n for some $n\in\mathbb{N}$. Now, we prove that $V_{n+1}(\vec{x},\vec{q})$ satisfies the theorem as well. Therefore assume that $(\vec{x}+e_i,\vec{q}-e_i),(\vec{x}+e_j,\vec{q}-e_j)\in\mathcal{S}$, then

$$\begin{aligned}&V_{n+1}(\vec{x}+e_i,\vec{q}-e_i)-V_{n+1}(\vec{x}+e_j,\vec{q}-e_j)\\ &\quad=0+\sum_{k=1}^{K}\lambda p_k(H_a(\vec{x}+e_i,\vec{q}-e_i,k)-H_a(\vec{x}+e_j,\vec{q}-e_j,k))\\ &\qquad+\sum_{k=1}^{K}\vec{x}_k\mu_k(H_d(\vec{x}+e_i-e_k,\vec{q}-e_i)-H_d(\vec{x}+e_j-e_k,\vec{q}-e_j))\\ &\qquad+\mu_i H_d(\vec{x},\vec{q}-e_i)-\mu_j H_d(\vec{x},\vec{q}-e_j)\\ &\qquad+(1-\lambda-\sum_{k=1}^{K}\vec{x}_k\mu_k)(V_n(\vec{x}+e_i,\vec{q})-V_n(\vec{x}+e_j,\vec{q}-e_j))\\ &\qquad-\mu_i V_n(\vec{x}+e_i,\vec{q}-e_i)+\mu_j V_n(\vec{x}+e_j,\vec{q}-e_j).\end{aligned}$$

The first term of the right-hand side is 0 since the total number of patients present in the system is equal in both cases. The second term deals with arrivals. The optimal action that is chosen in $H_a(\vec{x}+e_i,\vec{q}-e_i,k)$ can also be chosen in $H_a(\vec{x}+e_j,\vec{q}-e_j,k)$, although it is not necessarily optimal. Then, by the induction hypothesis, $H_a(\vec{x}+e_i,\vec{q}-e_i,k)-H_a(\vec{x}+e_j,\vec{q}-e_j,k)\geqslant 0$. The same goes for the third term dealing with departures, with the added remark that in the case when in $H_d(\vec{x}+e_i-e_k,\vec{q}-e_i)$ chooses to take q_i patients of type i in service, then in the other case $q_i - 1$ patients of that type should be taken into service. Then the induction hypothesis can be applied. To the sixth term the hypothesis can be applied directly. For the fourth, fifth, seventh and eighth term some rewriting is needed:

$$\begin{aligned}&\mu_i H_d(\vec{x},\vec{q}-e_i)-\mu_j H_d(\vec{x},\vec{q}-e_j)-\mu_i V_n(\vec{x}+e_i,\vec{q}-e_i)+\mu_j V_n(\vec{x}+e_j,\vec{q}-e_j)\\ &\quad=-\mu_i(V_n(\vec{x}+e_i,\vec{q}-e_i)-H_d(\vec{x},\vec{q}-e_i))+\mu_j(V_n(\vec{x}+e_j,\vec{q}-e_j)\\ &\qquad-H_d(\vec{x},\vec{q}-e_i)).\end{aligned}$$

We know that $\mu_i > \mu_j$. Then this is smaller than 0 because the difference between $V_n(\vec{x}+e_i,\vec{q}-e_i)$ and $V_n(\vec{x}+e_j,\vec{q}-e_j)$ or between taking a type i or type j patient into service, is smaller than the difference between $H_d(\vec{x},\vec{q}-e_i)$ and $H_d(\vec{x},\vec{q}-e_j)$, the optimal action on departure of a type i or type j patient respectively.

Then, by taking the limit of $n\to\infty$, $V(\vec{x}+e_i,\vec{q}-e_i)\leqslant V(\vec{x}+e_j,\vec{q}-e_j)$ if $\mu_i > \mu_j$. □

The next theorem shows that if the service requirements are equal for all types of patients, it is never optimal to leave a patient in the queue while there are servers available. So in this case the optimal policy will be a work-conserving policy. This also follows from intuition, because to minimize the number of patients present in the system you will serve them as quickly as possible once they have been admitted.

Theorem 3.2. *Assume that $c_k = 1$ for all $k = 1,\ldots,K$. Then an optimal policy will schedule patients in service while there are idle servers available.*

Proof. We need to prove that upon arrival, if a server is available, the patient will use this server. This means that we need that $V(\vec{x}+e_k,\vec{q})\leqslant V(\vec{x},\vec{q}+e_k)$.

Again this proof is by induction on n in V_n. Define $V_0(\vec{x},\vec{q})=0$ for all states $(\vec{x},\vec{q})\in\mathcal{S}$. Then of course $V_0(\vec{x}+e_k,\vec{q})\leqslant V_0(\vec{x},\vec{q}+e_k)$ holds. Now assume the proposition holds for some $n\in\mathbb{N}$. Now we prove that for $n + 1$ the statement holds as well. For $n + 1$ we have

$$
\begin{aligned}
&V_{n+1}(\vec{x},\vec{q}+e_k) - V_{n+1}(\vec{x}+e_k,\vec{q}) \\
&\quad = \sum_{j=1}^{K} \lambda p_j (H_a(\vec{x},\vec{q}+e_k,j) - H_a(\vec{x}+e_k,\vec{q},j)) \\
&\qquad + \sum_{j=1}^{K} x_j \mu_j (H_d(\vec{x}-e_j,\vec{q}+e_k) - H_d(\vec{x}+e_k-e_j,\vec{q})) - \mu_k H_d(\vec{x},\vec{q}) \\
&\qquad + \left(1 - \lambda - \sum_{j=1}^{K}\right)(V_n(\vec{x},\vec{q}+e_k) - V_n(\vec{x}+e_k,\vec{q})) + \mu_k V_n(\vec{x}+e_k,\vec{q}).
\end{aligned}
$$

The first term concerns the arrivals. It is easily seen that when taking the same action in the second part as is optimal in the first part, the term satisfies the statement using the induction hypothesis. The second term deals with actions after departures. Because of the induction hypothesis, as many patients as possible are taken into service. This means that the resulting state will be the same in both cases, and the second term is equal to zero. To the fourth term the induction hypothesis can be applied directly. The third term is smaller than the fifth term as a result of Lemma 3.1. This means that $V_{n+1}(\vec{x},\vec{q}+e_k) - V_{n+1}(\vec{x}+e_k,\vec{q}) \geqslant 0$ and the proof is complete. □

4. Numerical experiments

In this section we will perform numerical experiments to illustrate the results of the previous sections. We will discuss how the optimality equations have been solved efficiently to derive the optimal policies numerically. We first start with a discussion on parallelization of the implementation that has been crucial to the computations.

4.1. Parallelization

In order to accommodate for the heavy use of memory when using value iteration, the dynamic program has been parallelized with MPI so that it can be run on the DAS-3 cluster computer [7]. However, this is not a straightforward procedure. In the following paragraphs, we will explain how this was done and how parallelization yields a gain in performance.

The calculation steps in value iteration are very similar to the steps used in Successive Overrelaxation (SOR). The SOR algorithm is used for calculating the value of a stable state in a two-dimensional matrix, in which the edges are given an initial value, and the other points are calculated using their neighboring values. The total workload is divided blockwise over the available processors of the cluster computer, limiting the inter-processor communication to the neighboring rows of these blocks. Using this scheme, each processor only needs a fraction of the data. The main difference between the SOR algorithm and our value iteration algorithm is that the matrix in our problem is not two-dimensional. This shifts the focus from computation speed to memory use. In fact, for $K = 4$ and $B = 13$, the memory requirements are about 45 Gigabytes (assuming that the relative value function evaluated in a particular state requires a double to store its value).

The parallel program has been engineered to make optimal use of the memory while retaining most of the computational speed. In the value iteration algorithm, new values of the relative value function are communicated and stored directly into the neighboring blocks, without the use of additional communication buffers. Overall, the parallel version of value iteration uses significantly more memory than the sequential version, because some values in a

CPUs	problem size (×100,000)	U_{seq} (in Gb)	U_{par} (in Gb)
4	0.65	0.002	0.001
5	3.91	0.012	0.004
10	1,000.00	3.050	0.610
12	4,300.00	13.120	2.190
13	8,157.00	24.894	3.830
14	14,758.00	45.040	6.430

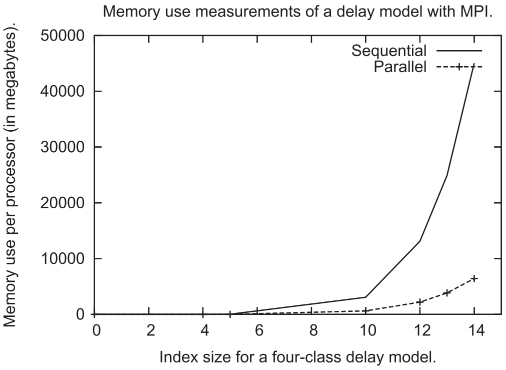

Fig. 1. Memory requirement for various eight-dimensional problems.

CPUs	time	speedup
1	64.17	0.00
2	57.81	1.11
4	32.29	1.99
8	18.60	3.45
16	11.07	5.79
32	5.87	10.94

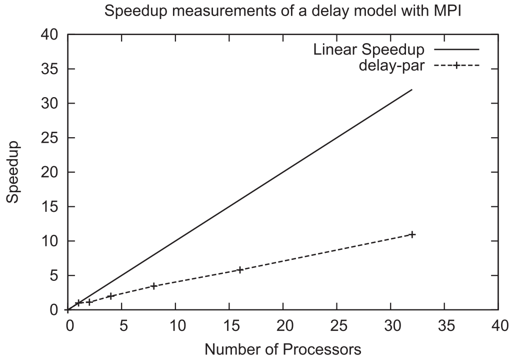

Fig. 2. Results of the speedup for a four-dimensional problem, $S = 32$.

block need to be replicated. The sequential program has a memory usage U_{seq} that is roughly of the size

$$U_{\text{seq}} = 2 \times 16 \times (\max\{B,S\}+1)^{2K} \text{ bytes}. \qquad (1)$$

The parallel version needs additional $2 \times 16 \times (\max B,S)^{2K-1}$ bytes per processor for communication. While this increases the memory usage a bit, the parallel version allows us to divide this memory in blocks over the available processors. The memory requirements U_{par} of the parallel version with p processors is then given by

$$U_{\text{par}} = \frac{U_{\text{seq}} + 2 \times 16 \times p \times (\max\{B,S\}+1)^{2K-1}}{p} \text{ bytes}. \qquad (2)$$

Fig. 1 illustrates the memory requirements for various problems with a state space that has eight dimensions. Similarly, Fig. 2 shows the speedup in computation time as more processors are utilized in the cluster computer.

In this section we describe some experiments we did to show how the model performs. For both the case with and without a waiting room we have performed experiments with the same parameters with regard to demand and number of servers available, so as to make the two cases comparable.

4.2. Scenario analysis

In this subsection, we describe several scenarios and compare the results and analyze the structure of the optimal policy. The scenarios we consider are given in Table 1. We keep the number of arriving patients and the total workload offered (almost) the same, but increase the number of different classes of patients, with the number of servers requested for each class equal to the class number. This approach will enable us to study the influence of the variability of patient demand on the performance of the system. The scenarios also reflect the fact seen in practice that the patients requiring more time per week tend both to be rarer and to have a longer service time. As these patients are also generally more urgent, they also have higher rejection costs.

Table 1
Scenarios for experiments with $\lambda = 5$.

Scenario	p_k	c_k	$\beta_k = 1/\mu_k$	r_k
1	0.7	1	22	1
	0.3	2	28	2
2	0.5	1	10	1
	0.3	2	20	2
	0.2	3	25	3
3	0.3	1	5	1
	0.2	2	10	2
	0.2	3	12	3
	0.2	4	15	4
	0.1	5	15	5
4	0.2	1	2	1
	0.2	2	3	2
	0.1	3	5	3
	0.1	4	5	4
	0.1	5	5	5
	0.1	6	5	6
	0.05	7	10	7
	0.05	8	10	8
	0.05	9	15	9
	0.05	10	15	10
5	0.2	1	1	1
	0.2	2	2	2
	0.1	3	2	3
	0.1	4	2	4
	0.05	5	5	5
	0.05	6	5	6
	0.05	7	5	7
	0.05	8	5	8
	0.05	9	5	9
	0.03	10	5	10
	0.03	11	10	11
	0.03	12	10	12
	0.03	13	10	13
	0.03	14	15	14
	0.03	15	15	15

Table 2
Results for the different scenarios.

Scenario	Optimal	Heuristic (%)
1	27.611	0.52
2	31.408	0.78
3	31.593	0.83
4	37.884	1.30
5	38.778	1.82

In the results we can see that the average cost when using the optimal policy increases with the variation in service requirements of the patients. This is of course to be expected, because reservation of larger numbers of servers is necessary for patients with a large service requirement, and this causes other patients to wait for a longer period of time. However, the increase in cost is not as high as might have been expected.

Also Table 2 shows the results for our heuristic, which is the best trunk reservation policy. With a trunk reservation policy we denote a policy in which some patient classes are blocked as a certain number of servers are occupied. For each patient class but the highest there is such a threshold. As can be seen the relative difference in average cost between the optimal policy and the best trunk reservation policy is very small. This means that the trunk reservation policy is a very good practical alternative to the optimal policy, as it is almost as good but much easier to compute, visualize and to implement.

One last remark we want to make is a point about actions taken upon departure of a patient. If a patient who uses a high number of servers leaves the system, it is possible to take into service a number of patients with lower service requirements. This can be seen in the expression for H_d. However, from our experiments it has become clear that the average cost does not increase significantly if it is assumed that at most one new patient is taken into service after a departure. The optimal policy differs only in a very low number of cases, and then only at the boundaries of the state space. It does however speed up the computation significantly to make this assumption.

5. Conclusions and further research

In the previous sections we have studied a model for personnel scheduling in care-at-home facilities. We discuss both the model with and without waiting room. The case with a waiting room has received little attention in the literature due to the complexity of the model. We proved some monotonicity properties of the value function and the structure of the optimal policy in some special cases. Experiments were conducted, and showed that a trunk reservation heuristic provides very good results to be used in practice. For both the optimal policy and the heuristic computation times were low.

For future research there remains the question of how to include different skill levels in the model, where some staff members have a higher skill level, and patients need at least a certain level to perform their service. Two possible approaches to pursue this research direction are value function decomposition (see, e.g., Bhulai [4]) and priority-based assignment (see, e.g., Wallace and Whitt [18]). Both methods were developed for staffing problems in multi-skill call centers. In [4], staff members are grouped into levels based on their skills and costs, and the levels prioritize the use of staff members in sequence. The levels are arranged such that staff members in level $i + 1$ can serve all patients that staff members in level i can, and perhaps more. Furthermore, the levels are also arranged such that level i is more preferable than level $i + 1$. Based on this hierarchy, it is shown that the relative value function (as described in Section 2) can be decomposed into relative value functions of single-skill models so that the problem remains tractable. A different approach is adopted in [18]. The authors assign priorities (stored in a big patient-staff member matrix) to the different staff members per patient and use overflow policies for the assignment. The optimization problem is now of combinatorial nature since an optimization technique is developed to optimize over the priority matrix. For this approach to work, one needs a quick evaluation of the performance of the system for a fixed matrix. This can be done by approximating the relative value function by Approximate Dynamic Programming techniques as outlined in Powell [15]. Both techniques allow for patients changing their requirements during their service, which is not accommodated for right now and needs to be addressed in future work.

References

[1] O.Z. Aksin, M. Armony, V. Mehrotra, The modern call-center: a multi-disciplinary perspective on operations management research, Production and Operations Management 16 (6) (2007) 665–688.

[2] E. Altman, T. Jiménez, G.M. Koole, On optimal call admission control in a resource-sharing system, IEEE Transactions on Communications 49 (2001) 1659–1668.

[3] S. Bertels, T. Fahle, A hybrid setup for a hybrid scenario: combining heuristics for the home health care problem, Computers & Operations Research 33 (2006) 2866–2890.

[4] S. Bhulai, Dynamic routing policies for multi-skill call centers, Probability in the Engineering and Informational Sciences 1 (2009) 75–99.

[5] E. Burke, S. Petrovic (Eds.), European Journal of Operational Research, Special Issue on Timetabling and Rostering, Vol. 153, Elsevier, 2004.

[6] E. Cheng, J.L. Rich, A home health care routing and scheduling problem. Technical Report CAAM TR98-04, Rice University, 1998.

[7] <http://www.cs.vu.nl/das3>.

[8] C.H. Ellenbecker, A theoretical model of job retention for home health care nurses, Journal of Advanced Nursing 47 (2004) 303–310.

[9] A.T. Ernst, H. Jiang, M. Krishnamoorthy, B. Owens, D. Sier, An annotated bibliography of personnel scheduling and rostering, Annals of Operations Research 127 (2004) 21–44.

[10] P. Eveborn, P. Flisberg, M. Rönnqvist, Laps care – an operational system for staff planning of home care, European Journal of Operational Research 171 (2006) 962–976.

[11] L. Flynn, J.A. Deatrick, Home care nurses' descriptions of important agency attributes, Journal of Nursing Scholarship 35 (2003) 385–390.

[12] N. Gans, G.M. Koole, A. Mandelbaum, Telephone call centers: tutorial, review, and research prospects, Manufacturing and Service Operations Management 5 (2003) 79–141.

[13] B. Miller, A queueing reward system with several customer classes, Management Science 16 (1969) 234–245.

[14] S. Petrovic, G. Vanden Berghe (Eds.), Annals of Operations Research, Special Issue on Personnel Scheduling and Planning, Vol. 155, Springer, Netherlands, 2007.

[15] W.B. Powell, Approximate Dynamic Programming: Solving the Curses of Dimensionality, John Wiley & Sons, Inc., 2007.

[16] M.L. Puterman, Markov Decision Processes: Discrete Stochastic Dynamic Programming, John Wiley & Sons, 1994.

[17] K.W. Ross, D.H.K. Tsang, The stochastic knapsack problem, IEEE Transactions on Communications 37 (1989) 740–747.

[18] R.B. Wallace, W. Whitt, A staffing algorithm for call centers with skill-based routing, Manufacturing and Service Operations Management 7 (2005) 276–294.