Redistributing Stock in Library Systems with a Depot

van der Heide, Gerlach; Roodbergen, Kees Jan; van Foreest, Nicky

Published in:
Computers & Operations Research

DOI:
10.1016/j.cor.2017.02.005

Computers and Operations Research 83 (2017) 66–77

Contents lists available at ScienceDirect

Computers and Operations Research

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

Redistributing stock in library systems with a depot

G. Van der Heide*, K.J. Roodbergen, N.D. Van Foreest

University of Groningen, P.O. Box 800, 9700 AV Groningen, The Netherlands

ARTICLE INFO

Article history:
Received 2 June 2016
Revised 23 December 2016
Accepted 8 February 2017
Available online 9 February 2017

Keywords:
Inventory policies
Markov decision process
Heuristics
Libraries
Depot

ABSTRACT

Public library organizations often utilize depots for carrying out shipments to libraries in case of stock-outs and for storing low demand rental items at low cost. Similar systems may be employed by rental companies for other rental products such as tools, DVDs, and jewelry. Since shipments deplete the depot's inventory, stock must be taken back from the libraries in order to deal with future shipment requests. These shipment and take-back operations are carried out periodically, e.g. daily or weekly. This work focuses on optimizing the decisions for shipments and take-backs. We model the system by means of a Markov decision process and investigate its optimal policy for various problem instances. For the take-back decision, we distinguish between so-called threshold, reactive, and preventive take-backs. We use the insights from the MDP to develop a three-phase take-back heuristic. In experiments, our heuristic performs within 1% on average from the optimal solution. For settings with a large number of libraries, it is shown that an acceptable performance can be achieved by setting a base-stock level at the depot and taking back sufficient stock from the libraries to achieve this level.

1. Introduction

In recent years, the Dutch public library system is increasingly adopting concepts derived from e-commerce. As of 2016, clients can access and order items online from a nationwide catalog showing all the items and their availability at each library in the country. The fulfillment of these online orders is a challenging task, in particular when clients request items for pickup from a library that does not have the item in stock. In order to satisfy such requests, libraries often introduce a joint depot dedicated to shipment of locally unavailable items.

Currently, public libraries in many countries ship directly between libraries in response to stock-outs. However, the number of items in a shipment is typically small, leading to ineffective usage of transportation devices. This fact has become increasingly pressing, since demand for physical books has significantly decreased over time after the introduction of internet and e-books. A depot allows to consolidate these small shipment streams into larger streams. This is easier to coordinate and creates significant economies of scale, since items can be shipped using fewer transportation devices and handled in one dedicated place. In addition, a depot can serve as a low cost storage location for items that are currently not needed. For this reason library organizations are increasingly adopting a system with shipments from a depot.

The above motivation applies not only to public libraries but also to rental systems in general. For example, tool rental companies store rare and expensive tools in a depot so that they can be shared effectively between rental locations. Other possible rental products for which a depot may be utilized are jewelry and DVDs. While the focus application of this article is public libraries, insights and heuristics carry over analogously to those other rental systems.

In library systems with a depot, various operational decisions are carried out periodically. When locally unavailable items are requested at a library, these are shipped from the depot. In contrast to sales-driven companies, where stock is bought and sold, stock in library systems is often fixed and all rented items are returned by the client. The depot will thus have to be resupplied by carrying out a take-back operation of items from the individual libraries. The main difficulty lies in deciding how many items to take back in total and from which libraries. Since due to budget cuts the government funding for public libraries has significantly decreased in the last several years, it is important to carry out these operations efficiently.

An often encountered practical problem for public libraries is that a large part of the collection consists of low-demand items. Muckstadt and Thomas (1980) conclude that two-echelon systems are important for low-demand items. Hence, storage of such items in a low-cost depot may be an effective strategy to reduce holding costs and free up space at the libraries for other items or activi-

* Corresponding author.
E-mail addresses: g.van.der.heide@rug.nl (G. Van der Heide), k.j.roodbergen@rug.nl (K.J. Roodbergen), n.d.van.foreest@rug.nl (N.D. Van Foreest).

http://dx.doi.org/10.1016/j.cor.2017.02.005
0305-0548/

ties. An important problem is deciding which low-demand items to store in the depot.

In this paper we simultaneously optimize the decisions for resupplying the depot and dealing with low-demand items at the libraries. We consider a periodic review model where demands and returns at the libraries occur between reviews. At the review, stock is observed and there is an option to carry out shipments and take-backs. By first solving an MDP for a problem with a single library and single depot, we obtain the main insights for storing low-demand items at the depot. Subsequently, we solve MDPs for problems with multiple libraries. By analyzing the optimal policy for several example configurations, we obtain insights into optimal shipment and take-back operations. Based on the insights we formulate a near-optimal heuristic for larger problem instances and in various experiments we compare it to the optimal policy and several other simple heuristics.

The research on multilocation rental systems has mainly focused on vehicle rentals systems (Ernst et al., 2011; Li and Tao, 2010). For vehicle rentals, the common option for dealing with stock-outs is to provide substitute vehicles, whereas for library books shipments can be a practical option because the items are easily shipped and clients are typically willing to wait for a shipment. In vehicle rentals, shipments from another location in response to demand are typically only considered in deterministic problems. For example, Ernst et al. (2011) determine an optimal schedule for a finite planning horizon where bookings are known in advance. For a bike sharing system, Dell'Amico et al. (2016) solve a single period rebalancing problem. Such deterministic methods do not match the multiperiod stochastic setting that we consider.

In stochastic settings, the main focus in vehicle rental literature is on optimizing the fleet size and the fleet redistribution policy, often by applying queuing theory (George and Xia, 2011). Li and Tao (2010) use dynamic programming to optimize the redistribution policy. The authors consider a two-location system where vehicles rented from one location can be returned to the other location in the same period. While we do not consider returns to other locations, we add a depot, more than two locations, and consider stochastic rental times. Our MDP therefore has a significantly larger state space than the dynamic program in Li and Tao (2010) and requires an efficient implementation. For a multilocation library system with lateral transshipments, Van der Heide and Roodbergen (2013) apply dynamic programming to optimize lateral shipments and stock redistribution policies. They show that a dynamic redistribution policy, accounting for current on-hand and rented stock at each library, significantly outperforms the standard policy in practice of sending back each item to its owner location. To the best of our knowledge, no authors have considered the redistribution of stock in a library system with a depot.

Shipments from a depot in response to stock-outs have been studied recently in spare parts inventory control. In a case study for the spare parts division of a car manufacturer, Axsäter et al. (2013) demonstrate that significant cost savings can be achieved by introducing the shipment option. Van Wijk et al. (2013) derive structure results for the optimal operational decisions of assigning shipment requests from the depot to local stock points. In the above papers, stock is transferred from the depot to local stock points, but no attention is paid to the transfer of returned stock from local stock points to the depot, which is an important feature of library systems.

Hub-and-spoke systems are also characterized by exchanges of vehicles between locations and a depot. In a hub-and-spoke system, vehicles rented at the hub return at the spoke, and vice versa, while in the library system items are typically rented from and returned to the same location. These essentially different dynamics demand different strategies for repositioning stock. Köchel (2007) and Song and Carter (2008) consider repositioning of empty cars in hub-and-spoke systems. Both authors start by considering repositioning policies for systems with a hub and a single spoke. The resulting policies are used to formulate heuristics for systems with multiple spokes. We follow a similar approach by basing part of our heuristic on the optimal policy of the single library problem.

The outline of the article is as follows. Section 2 introduces the model for the library system with a depot. In Section 3 MDPs are solved for base scenarios with one, two, and three libraries to gain insight into the relevant trade-offs. In Sections 4 and 5 shipment and take-back heuristics are developed, which are compared to the optimal policy and to each other in Section 6. Finally, Section 7 concludes.

2. Problem formulation

In this section we formulate the problem of shipping and taking back stock for a library system with n libraries and one depot. The depot is indexed by $i = 0$ and the libraries by $i = 1, \ldots, n$. The system is depicted schematically in Fig. 1. A downstream movement of stock from the depot to the libraries is called a *shipment*. An upstream movement from a library to the depot is called a *take-back*.

We consider the inventory control for a single item type, e.g., a specific book title. It seems reasonable to assume that in settings with low demand and quick shipments of back-up stock from a depot, the effect of substitution in case of stock-outs is negligible. Therefore, we can repeat our analysis for every item type in case there are multiple item types. It is straightforward to extend the mathematical model with substitution by including multiple item types, however, given that the problem without substitution is already of significant interest, we believe such an extension is beyond the scope of this paper.

The total number of copies of this item is fixed and given by K. Libraries in practice typically allow a limited number of backorders per library in order to reduce administrative inconvenience and waiting times. We let $B > 0$ be the maximum number of backorders per library; any additional demand is lost. In case there is full backordering, which could be the case for other rental companies, we could set B large enough to approximate full backordering situations. In Fig. 1, $x_{0t} \geq 0$ is the on-hand inventory at the depot in period t. The on-hand inventory at library $i, i = 1, \ldots, n$ is given by $x_{it} \geq -B$. Similarly, $y_{it} \geq 0$ is the number of items rented from library $i, i = 1, \ldots, n$. The state S_t of the system in period t is represented as $S_t = (x_{0t}, x_t, y_t)$, with $x_t = (x_{1t}, \ldots, x_{nt})$ and $y_t = (y_{1t}, \ldots, y_{nt})$.

The library system employs a periodic review policy. In typical library systems, these reviews are executed on a daily, biweekly, or weekly basis. The time line of events is summarized in Fig. 2, where the state after demands/returns, shipments, and take-backs are indexed with zero, one, and two primes, respectively. For example, the on-hand inventory levels after these respective phases are given by x_t, x'_t and x''_t.

Period t starts with clients demanding D_t new items and returning R_t previously rented items at the libraries. The demand and return processes are as follows. Library i faces demand D_{it} during period t. We use the common Poisson process for modeling customer arrivals, hence $D_{it} \sim$ Poisson(λ_i) with λ_i a library spe-

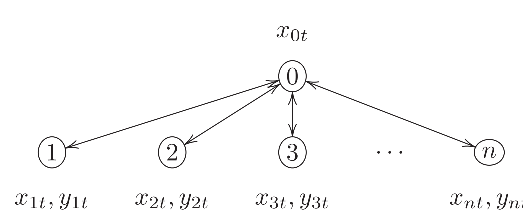

Fig. 1. A library system with n libraries and depot.

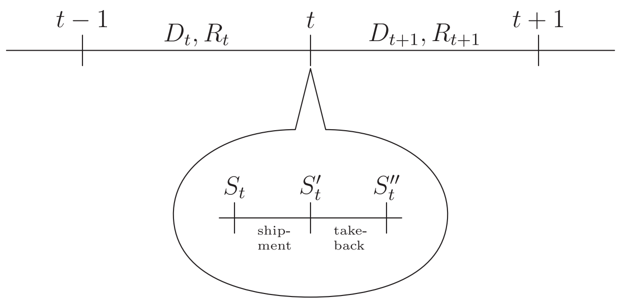

Fig. 2. Time line of events. The cloud shows the actions at the review of period t and the states after carrying out the actions.

cific demand rate. Library i faces R_{it} returns during period t. While in principle any return distribution can be modeled by keeping track of the rental time of each rented item in the state variable, we focus on a geometric return distribution for tractability and ease of exposition. The rental time of each rented item is Geometric(p) distributed. With y_{it} items rented at library i, we have that $R_{it} \sim$ Binomial(y_{it}, p). For related library systems, Van der Heide (2015) shows that policies obtained under geometric returns typically perform well even if the actual returns are empirically distributed.

At the end of period t, the inventory levels at the libraries and the depot are reviewed. Accordingly, shipment and take-back actions are carried out. Both actions are carried out at the same time because the transportation device of the depot visits the libraries once per period. The lead time for shipments and take-backs is assumed to be negligible. The rationale is similar to Axsäter et al. (2013): these actions can be carried out overnight while the libraries are closed, so the actions will be executed before any following events can occur. In accordance with current practice, stock taken back from one library cannot be shipped to another library in the same period, however, it is possible to store stock at the depot for one period and ship it in the next period.

After the shipment and take-back actions in period t, inventory costs are incurred and a new period starts. We have the following cost parameters. The depot has holding cost h_0 and the libraries holding cost h per unit of on-hand stock per period. We assume $h_0 < h$ since the depot is dedicated to storage whereas libraries are dedicated to demand fulfillment. The cost per backorder per time unit is b and the cost per lost demand is ℓ. The lost demand cost could measure losses in municipal subsidies and subscription fees due to customers canceling subscriptions or the dissatisfaction from not meeting the goal of providing maximum service. Since shipments and take-backs require manual picking of books from shelves, there is a handling cost c for each unit shipped and each unit taken back. To avoid trivial situations where all stock is parked at the depot at all times, we assume $2c > h - h_0$.

In the remainder of this section we describe, by means of several recursive equations, the dynamics of our model for its three phases: demands and returns, the shipment decision, and the take-back decision. After that we formulate the costs and a Markov Decision Process (MDP) for optimizing the shipment and take-back decisions.

2.1. Demands and returns

Suppose that the state is $S''_{t-1} = (x''_{0,t-1}, x''_{t-1}, y''_{t-1})$ after the review of previous period $t-1$. The depot faces no demand, hence

$$x_{0t} = x''_{0,t-1}.$$

Let D_t and R_t be the vectors of demands and returns. At the libraries returning items are added to the stock and demanded items are taken by clients. Backorders that exceed B are cut off. Hence, we take the pairwise maximum of the new stock after demands/returns and $-B$, i.e.,

$$x_t = \max\{x''_{t-1} + R_t - D_t, -B\}. \quad (1)$$

For the rented items, note that the difference in on-hand inventory before and after demands and returns is given by $(x''_{t-1})^+ - (x_t)^+$, with $(x)^+ = \max\{x, 0\}$. If the on-hand inventory decreases, clients rented $(x''_{t-1})^+ - (x_t)^+$ more items than they returned and if the on-hand inventory increases, clients returned $(x''_{t-1})^+ - (x_t)^+$ more items than they rented. The rented items after demands and returns are therefore

$$y_t = y''_{t-1} + (x''_{t-1})^+ - (x_t)^+.$$

The new state after demands and returns is $S_t = (x_{0t}, x_t, y_t)$.

2.2. Shipment decision

At the end of period t the stock levels of state S_t are reviewed and a decision has to be made regarding how many items to ship. Since libraries aim to maximize service to their clients, the depot always ships an item from its on-hand stock if there are unmet demands at some library. Proactive shipments from the depot to the libraries are not used since service is considered adequate if an item is shipped in the same period as it is ordered.

Let the decision variable $z'_t \geq 0$ be the vector with amounts shipped from the depot to the libraries. For example, if $n = 3$ and $z'_t = (1, 0, 0)$, one unit is shipped to the first library. The backordered demand in the system at time t is given by $\sum_{i=1}^{n} (x_{it})^-$ with $(x_{it})^- = -\min\{x_{it}, 0\}$. If the backordered demand exceeds the stock at the depot, whatever stock available at the depot is shipped downstream. Otherwise, all backordered demand is shipped. Hence, the number of shipped items is

$$Z'_t = \min\left\{x_{0t}, \sum_{i=1}^{n} (x_{it})^-\right\}.$$

The following restrictions on z'_t ensure that the Z'_t items in total are shipped only to libraries with backorders:

$$0 \leq z'_t \leq (x_t)^- \quad (2)$$

$$\sum_{i=1}^{n} z'_{it} = Z'_t.$$

In case $Z'_t = \sum_{i=1}^{n} (x_{it})^-$, it is easy to see that the only feasible choice is $z'_t = (x_t)^-$, which means all backorders are dealt with. In case $Z'_t < \sum_{i=1}^{n} (x_{it})^-$, some libraries cannot receive shipments, requiring a choice between libraries. This choice influences not only backorders in the current period, but also on-hand stock at the library several periods later due to the future return of the item.

After shipments at time t, the depot's inventory has decreased by Z'_t:

$$x'_{0t} = x_{0t} - Z'_t.$$

Since libraries with backorders receive z'_t shipments, their number of backorders decreases by z'_t:

$$x'_t = x_t + z'_t.$$

Finally, all shipped items are given to waiting clients. Hence, the new number of rented items is

$$y'_t = y_t + z'_t.$$

The updated state after the shipment decisions is given by $S'_t = (x'_{0t}, x'_t, y'_t)$.

2.3. Take-back decision

The second part of the review at time t concerns take-backs of stock from the libraries to the depot. The state after shipments, S'_t, is first observed. Then a decision is made regarding which items, if any, are taken back from the libraries. These items are stored at the depot and can be used for shipments in the next period.

Let the decision variable $z''_t \geq 0$ be the vector with the amounts taken back from libraries to the depot. Clearly, the following restriction on z''_t holds:

$$0 \leq z''_t \leq (x_t)^+, \tag{3}$$

i.e., no more items can be taken back than are in stock. Note that (2) and (3) imply that the sets of libraries which are eligible for shipments and for take-backs are mutually exclusive, since by definition $(x_{it})^+ \cdot (x_{it})^- = 0$.

The state after take-backs updates straightforwardly. The depot receives $\sum_{i=1}^{n} z''_{it}$ units, hence

$$x''_{0t} = x'_{0t} + \sum_{i=1}^{n} z''_{it}.$$

The local stock levels reduce by the amounts taken back

$$x''_t = x'_t - z''_t.$$

The rented items remain at the clients:

$$y''_t = y'_t.$$

The new state is then $S''_t = (x''_{0t}, x''_t, y''_t)$. Any remaining unmet demand carries over as a backorder into the next period.

2.4. Costs

The costs are incurred after carrying out shipments and take-backs. Let $z_t = (z'_t, z''_t)$ be the decisions in period t. Excluding lost demand costs, the costs in period t with starting state S_t and decisions z_t are given by

$$C_t(S_t, z_t) = h_0 x''_{0t} + \sum_{i=1}^{n} \left(h(x''_{it})^+ + b(x''_{it})^- + cz'_{it} + cz''_{it} \right). \tag{4}$$

Here, the first part gives holding cost at the depot after the periodic review. The summation includes the holding, backorder, shipment, and take-back costs for each library.

In order to calculate the lost demand costs we need to know the number of items cut off in (1). Keeping track of this information in the state variable comes at a significant computational cost. Therefore, we use the following insight. Given the state S_t and decision z_t, we can calculate the expected lost demands in the next demand/returns phase. The expected lost demand costs are then simply the unit lost demand cost multiplied by the total expected number of lost demands, i.e.,

$$L_t(S_t, z_t) = \ell \sum_{i=1}^{n} E[(x''_{it} + R_{i,t+1} - D_{i,t+1} + B)^-]. \tag{5}$$

The insight is that we can include these expected costs already in the current period. The costs for state S_t and decision z_t then become $C_t(S_t, z_t) + L_t(S_t, z_t)$. Under an average cost criterion, these modified costs do not change the average cost nor the optimal policy.

2.5. Markov decision process

Now we can model a Markov Decision Process with an average cost criterion. Let $\mathcal{S}$ be the state space and let $V_0(S) = 0$ be the terminal costs for all $S \in \mathcal{S}$. The value function in state $S \in \mathcal{S}$ is then

$$V_t(S) = \min_{z_t} \{C_t(S, z_t) + L_t(S, z_t) + E[V_{t-1}(S)]\}. \tag{6}$$

This is simply the sum of the direct costs for decision z_t, the expected lost demand costs made during the transition after decision z_t, and the expected value of the states reached after demands and returns.

Under an average cost criterion we want to minimize for each $S \in \mathcal{S}$

$$\lim_{t\to\infty} \frac{1}{t} V_t(S).$$

We use value iteration to obtain an ε-optimal policy, with ε a small positive number. Letting $M_t = \max_{S\in\mathcal{S}} \{V_t(S) - V_{t-1}(S)\}$ and $m_t = \min_{S\in\mathcal{S}} \{V_t(S) - V_{t-1}(S)\}$, we apply the convergence criterion

$$\frac{M_t - m_t}{m_t} < \varepsilon. \tag{7}$$

For the optimal average cost g it is known that $m_t < g < M_t$ (Tijms, 2003). Since we consider average cost optimal policies, we drop the time index t in the remainder of the paper.

3. Examples of optimal shipment and take-back decisions

We solve the MDP for a base scenario in order to obtain insights into optimal shipment and take-back decisions. We use the value iteration algorithm from Section 2.5 to obtain the optimal policies. While this is not a conclusive analysis, we use these insights to develop our heuristic. Considering the very good performance of the heuristics (see Section 6) we deem the examples as presented in this section to be representative of common behaviour of the problem at hand.

We consider a base scenario with up to $n = 3$ libraries. We calculated the optimal policy for multitudes of other scenarios and we believe that this base scenario is representative. The base scenario has the following parameter setting: $K = 4, B = 2, p = 0.3, h_0 = 0.7, h = 1, c = 5, b = 10, \ell = 20.$ The demand rates of library 1, 2, and 3 are $\lambda_1 = 0.3, \lambda_2 = 0.2,$ and $\lambda_3 = 0.1.$

We start with take-back decisions for $n = 1,$ $n = 2,$ and $n = 3$ libraries. For $n = 1,$ the only reason to carry out take-backs is to reduce holding costs. For $n = 2,$ take-backs from a library can also be carried out to deal with current or future stock-outs at the other library. In the $n = 2$ case only one library can supply items in case of a backorder, hence we also consider the choice of supplying libraries in the $n = 3$ case. Finally, we pay attention to the shipment decisions by showing an example for the $n = 3$ case.

3.1. Take-backs in the single library case

Now we consider the base scenario with one library and one depot. All shipment decisions are fixed in this setting, because the depot ships its available stock whenever the library has a backorder. Therefore, we only have to consider the take-back policy.

Fig. 3 graphically shows the take-back policy with the base scenario in the left graph. For each combination of on-hand stock x_1 and rented stock y_1, the graphs shows the optimal stock levels after the take-back. The solid line can be regarded as a threshold on the on-hand stock. If, for a given value of y_1, x_1 exceeds the threshold, then all items above the threshold are taken back to the depot. If, for a given value of y_1, x_1 is below the threshold, then the optimal decision is to do nothing. The right graph shows the threshold line for higher values of y_1, obtained by setting $K = 9$. The threshold decreases in y_1 and ultimately becomes 0.

The structure in Fig. 3 seems intuitive. Taking back an item incurs a cost c, and shipping it when it is demanded, incurs an additional cost c. As long as an item remains at the depot after a

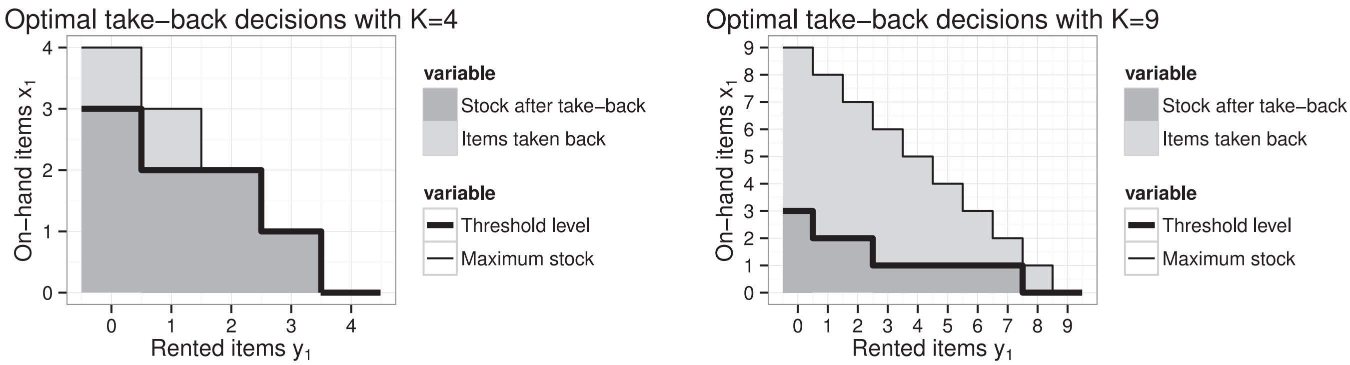

Fig. 3. Optimal stock levels after the take-back in a single library problem, for the base scenario and a modified scenario with $K = 9$.

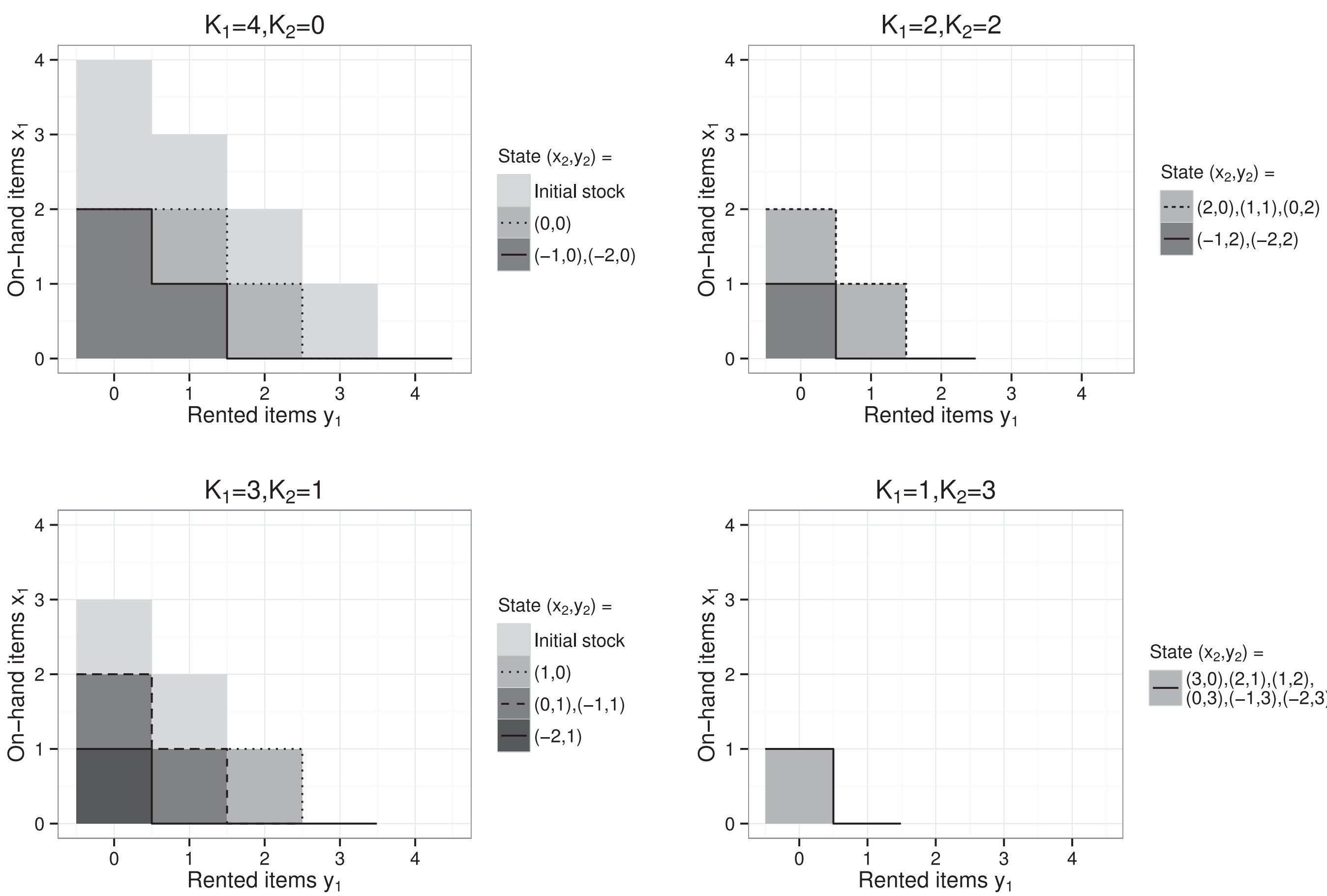

Fig. 4. Take-back policy at library 1 for the base scenario with two libraries.

take-back, the reduction in holding costs is $h - h_0$ per period. For take-backs to be economically viable, the expected number of periods that an item remains at the depot needs to be large enough to cover the transportation costs $2c$. This expected number of periods clearly increases in x_1, motivating a threshold on the on-hand stock. The threshold decreases in y_1 because rented items count as additional on-hand items when they return in the future. From now on we refer to the items taken back with the purpose of saving holding costs as *threshold take-backs*.

3.2. Take-backs in the two library case

For the two library case we also graphically show the take-back decisions in order to observe trade-offs in the optimal policy. It is useful to introduce $K_i = (x_i)^+ + y_i$ as the total number of items at library $i, i = 1, 2$. Fig. 4 depicts the optimal take-backs at library 1 for the base scenario. In order to reduce the dimensions of the graph, the figure depicts only the decisions for states with $x_0 = 0$ at the depot. Four graphs are shown with $K_1 = 4, 3, 2, 1$ items dedicated to library 1 and $K_2 = 0, 1, 2, 3$ to library 2.

For each value of y_1, the height of the bars represent the total available stock x_1 at library 1 before the take-back. The threshold lines represent the stock after the take-back, and these thresholds may vary with the state (x_2, y_2) at library 2. For example, for $K_1 = 4, K_2 = 0$ with $x_1 = 3, y_1 = 1$, we take-back 1 item if $(x_2, y_2) = (0, 0)$, but 2 items if $(x_2, y_2) = (-1, 0)$ or $(-2, 0)$. The thresholds in all graphs decrease in y_1. The thresholds also seem to decrease in the on-hand stock at library 2, so that relatively more stock from library 1 is taken back to the depot if stock at library 2 is relatively scarce.

Table 1
A selection of preventive take-back decisions for the base scenario with three libraries.

state	x_0	x	y	z''
1	0	(2,1,1)	(0,0,0)	(0,0,0)
2	0	(1,2,1)	(0,0,0)	(0,1,0)
3	0	(1,1,2)	(0,0,0)	(0,0,1)
4	0	(1,1,1)	(0,1,0)	(0,0,0)
5	0	(1,1,1)	(0,0,1)	(0,0,1)
6	0	(2,1,0)	(0,1,0)	(1,0,0)
7	0	(2,0,1)	(0,0,1)	(0,0,1)

It is interesting to compare the take-back policy of the $K_1 = 4$, $K_2 = 0$ case with the single library take-back policy for $K = 4$ in Fig. 3. For any y_1, we carry out at least the threshold take-backs from the single library case and possibly more than that. The reason for this is that, in addition to a reduction in holding costs, a take-back can prevent backorder costs at the other library.

Besides threshold take-backs, we can distinguish between two other types of take-backs. The first type is a *reactive take-back*, carried out in response to current backorders at library 2. The backorder cost b each time unit provides a clear incentive for doing this. We see for the cases $K_1 = 4$, $K_2 = 0$ and $K_1 = 3$, $K_2 = 1$ that the number of items taken back is always sufficient to meet all backorders at library 2. For the case $K_1 = 2$, $K_2 = 2$, only one item is taken-back in case $x_1 = 2$ and $x_2 = -2$. For the case $K_1 = 1$, $K_2 = 3$ take-backs from library 1 are never carried out. Since library 2 has several rented items, backorders can be dealt with by waiting for returns. Another reason not to take back the last item of library 1 are potential future backorder costs at library 1 itself.

The second type is a *preventive take-back*, carried out to prevent future backorders. In the graphs for $K_1 = 4$, $K_2 = 0$ and $K_1 = 3$, $K_2 = 1$ there are several states with $x_2 \geq 0$ for which the number of items taken back exceeds the number of threshold take-backs that we would expect from the single library case. The probability that stock-outs occur at library 2 is quite high in these states, motivating the decision to carry out preventive take-backs. For the states with $K_1 \leq 1$, no preventive take-backs are carried out, because typically the stock-out probability of library 2 is low. Moreover, taking back the last item of library 1 leads to unnecessary transportation costs when there is a significant probability of demand at library 1 in the next period. Even in states with $x_0 > 0$, not shown here, preventive take-backs are occasionally carried out.

3.3. Take-backs in the three library case

Most aspects of the optimal take-back policy have been covered in the discussion of the single and two library case. There is one interesting additional aspect with $n > 2$ libraries, namely, if a preventive or reactive take-back is required, we must decide which library supplies the item. Table 1 below shows a selection of preventive take-back decisions for several states in the base scenario. In case of reactive take-backs, the location to take back from is similar. Recall that the demand rates are $\lambda_1 = 0.3$, $\lambda_2 = 0.2$, and $\lambda_3 = 0.1$.

In states 1, 2, and 3 in Table 1, all stock is on-hand. When $x = (2, 1, 1)$ no items are taken back. However, when $x = (1, 2, 1)$ and (1,1,2), the second item at library 2 and library 3 is taken back. The take-back prevents potential backorders at library 1, because it has a higher demand rate than the other libraries. States 4 and 5 have in common that $x = (1, 1, 1)$. A take-back of the last on-hand item is carried out if library 3 has a rented item, but not when library 2 does. Library 2 has a reasonable probability that its on-hand item is required in the next period. In states 6 and 7, library 1 has stock $x_1 = 2$ and one of the other libraries has $x_i = 1$ and $y_i = 1$. If this is the case at library 2, one item is taken back from library 1. However, if this is the case at library 3, then one item is taken back from library 3 since it has the lowest demand rate. In most cases items are taken back from the library with the lowest stock-out probability in the next period.

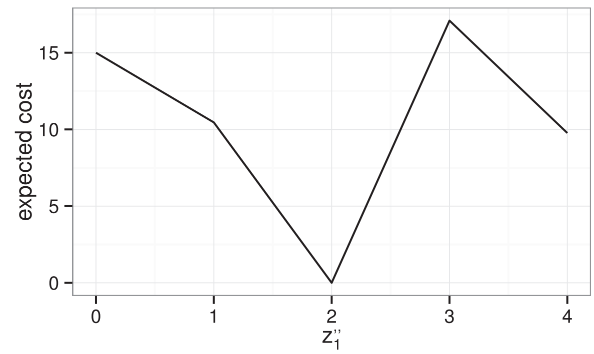

Fig. 5. Difference in expected cost for different take-back decisions.

3.4. Shape of the value function

In order to see whether the value function has usable properties, we plotted it for various scenarios. Li and Tao (2010), for example, showed that the value function for the last period is concave in the decision variable when there are backorders and nonconcave when there are lost sales. Here, we have a mixture of backorders and lost sales. While for some states the value function is unimodal in the decision variables, various situations exist where this does not hold. One such situation is shown in Fig. 5, for the same setting as in Section 3.3, except that $\lambda_1 = \lambda_2 = \lambda_3 = 1$. The state is $x_0 = 0$, $x = (4, -1, -1)$, and $y = (0, 0, 0)$. The graph shows the increase in the expected cost $C(S, z) + L(S, z) + E[V(S)]$ when taking take-back decision $z''_1 = 0, 1, \ldots, 4$ compared to the optimal decision $z''_1 = 2$. The fact that the value function is not unimodal even in one dimension emphasizes the challenge to obtain analytical results.

3.5. Shipments in the three library case

In many situations the shipment decision is easy since the depot meets all unmet demand to the extent possible. This is the case if $x_0 \geq \sum_{i=1}^{n} (x_i)^-$. A choice between libraries is required only if $x_0 < \sum_{i=1}^{n} (x_i)^-$ and $(x_i)^- > 0$ for at least two libraries.

In Table 2 we show the optimal shipment decisions in various states in the base scenario. All states are such that $x_0 = 1$, $x_1 < 0$, $x_2 < 0$, $x_3 = 0$, and $y_3 \geq 0$. We vary y_3 so that we can see decisions for all possible combinations of rented items at library 1 and 2. The reason to consider these specific states is that, since we have only one item at the depot, it is easy to see which of the two libraries receives the shipment. We prevent irregular behavior at the border of the state space, later explained, by changing the lost demand cost to $\ell = 60$.

In states 1–12 we have $y_1 < y_2$ and in states 13–24 we have $y_2 > y_1$. In all of these states, the shipment decision $z'_i = 1$ for the library with the lowest y_i. Since these libraries have the fewest future returns, their expected backorder costs will be high. In addition, by leaving backorders at the library with the highest number of rented items, we maximize the probability that backorders are met with returns, potentially reducing future shipment costs. It seems obvious to ship with priority to a library with $y_i = 0$, since its backorders can never be met by future returns.

In states 25–32 the number of rented items at both libraries is equal. The number of backorders and the height of the demand rate seem to break the tie. Here, the item is always shipped to

Table 2
Optimal shipment decisions for the base scenario with $n = 3$ libraries and lost demand cost $\ell = 60$.

state	x_0	x_1, x_2	y_1, y_2	z'_1, z'_2	state	x_0	x_1, x_2	y_1, y_2	z'_1, z'_2
1	1	$(-1, -1)$	(0,1)	(1,0)	17	1	$(-1, -1)$	(2,0)	(0,1)
2	1	$(-1, -2)$	(0,1)	(1,0)	18	1	$(-1, -2)$	(2,0)	(0,1)
3	1	$(-2, -1)$	(0,1)	(1,0)	19	1	$(-2, -1)$	(2,0)	(0,1)
4	1	$(-2, -2)$	(0,1)	(1,0)	20	1	$(-2, -2)$	(2,0)	(0,1)
5	1	$(-1, -1)$	(0,2)	(1,0)	21	1	$(-1, -1)$	(2,1)	(0,1)
6	1	$(-1, -2)$	(0,2)	(1,0)	22	1	$(-1, -2)$	(2,1)	(0,1)
7	1	$(-2, -1)$	(0,2)	(1,0)	23	1	$(-2, -1)$	(2,1)	(0,1)
8	1	$(-2, -2)$	(0,2)	(1,0)	24	1	$(-2, -2)$	(2,1)	(0,1)
9	1	$(-1, -1)$	(1,2)	(1,0)	25	1	$(-1, -1)$	(0,0)	(1,0)
10	1	$(-1, -2)$	(1,2)	(1,0)	26	1	$(-1, -2)$	(0,0)	(0,1)
11	1	$(-2, -1)$	(1,2)	(1,0)	27	1	$(-2, -1)$	(0,0)	(1,0)
12	1	$(-2, -2)$	(1,2)	(1,0)	28	1	$(-2, -2)$	(0,0)	(1,0)
13	1	$(-1, -1)$	(1,0)	(0,1)	29	1	$(-1, -1)$	(1,1)	(1,0)
14	1	$(-1, -2)$	(1,0)	(0,1)	30	1	$(-1, -2)$	(1,1)	(0,1)
15	1	$(-2, -1)$	(1,0)	(0,1)	31	1	$(-2, -1)$	(1,1)	(1,0)
16	1	$(-2, -2)$	(1,0)	(0,1)	32	1	$(-2, -2)$	(1,1)	(1,0)

the library with the highest number of backorders. This increase the probability that an additional backorder is met by the same shipped item after the client returns it. Finally, if the number of backorders is equal, i.e., in states 25,28,29, and 32, the highest demand rate is prioritized. By shipping to the library with the highest demand rate, the item can be used later to meet future demand. In this example we can thus sort the libraries lexicographically by lowest number of rented items, highest number of backorders, and highest demand rate.

In experiments, we observed that this lexicographical ordering seems to hold in scenarios with low values of B, i.e., $B = 1$ or $B = 2$, which are typical values encountered in practice. For larger values of B, the highest number of backorders may sometimes be prioritized over the least number of rented items, so that backorders can be dealt with by future returns.

Now we explain the aforementioned irregular behavior at the border of the state space. If ℓ is low compared to b, it may be favorable to avoid shipping to libraries with $x_i = -B$ or x_i close to $-B$. Losing a demand at a one-time cost ℓ may be less costly than backordering and paying b per time unit for a considerable number of periods. This effect is most influential when K, the total number of items, is low compared to demand. In typical settings, K will be high enough to fulfill most of the demand from on-hand stock. Hence, we pay no further attention to this effect when we develop heuristics.

4. Shipment heuristics

Since the MDP can only be solved for limited size instances, we develop heuristics for large instances that are easy to apply in practice. Here, we formulate two heuristics for the shipment decision: the ordering shipment (SO) heuristic and ship-to-first (SF) heuristic. The SO heuristic assigns shipment requests to libraries with backorders by using a lexicographical ordering inspired by observations from the MDP in Section 3.5. The SF heuristic assigns the request to the library with the lowest index, which is a simple policy sometimes used in practice.

The SO and SF heuristics starts with initial shipment decision $z' = 0$. Then iteratively a new receiving library is determined from the set of libraries with backorders and its z'_i is increased by 1. This is repeated until either the depot is empty or all backorders are dealt with, i.e.,

$$x_0 - \sum_{i=1}^{n} z'_i = 0 \text{ or } \sum_{i=1}^{n} (x_i + z'_i)^- = 0.$$

At any step of the SO and SF heuristic, determine $I^- = \{i : x_i + z'_i < 0\}$ as the set of libraries with remaining backorders. The SO heuristic sorts libraries $i \in I^-$ lexicographically in descending order according to the lowest number of rented items, $y_i + z'_i$, the highest number of backorders, $(x_i + z'_i)^-$, the highest demand rate, λ_i, and the lowest index, i. The receiving library is the first library in the lexicographical order. For the SF heuristic the index of the receiving library is $\min_{i \in I^-}\{i\}$, i.e., the lowest index in the set of libraries with backorders.

5. Take-back heuristics

In this section we develop a three phase take-back heuristic (TT). This heuristic consists of three separate phases in which we deal with the threshold, reactive, and preventive take-backs as introduced in Section 3. For each of these phases we describe in Sections 5.2–5.4 how to iteratively determine a heuristic take-back decision z''. Moreover, in Section 5.5 two simpler heuristics are proposed which may be easier to coordinate in practice: the base-stock take-back (TB) heuristic and take-back all (TA) heuristic.

5.1. Expected stock-out time

It is useful to first define the expected stock-out time for a library, which will be an important criterion in the TT and TB heuristic for selecting candidate libraries for reactive and preventive take-backs. The time until stock-out for library i can be characterized as follows. Let D_{is} denote the demand s periods from now and $R_{is}(y_{i,s-1})$ the returns s periods from now, which is a function of the rented items $y_{i,s-1}$ in the preceding period $s - 1$. Letting $y_{i,0} \equiv y_i$, then the stock-out time for library i in state x_i, y_i is defined as

$$T_i(x_i, y_i) = \min\left\{ t : \sum_{s=1}^{t} D_{is} \geq x_i + \sum_{s=1}^{t} R_{is}(y_{i,s-1}) \right\}, \tag{8}$$

i.e., the first moment that the total demand exceeds the on-hand stock plus the total returns up to time t. Since $T_i(x_i, y_i)$ is a hitting time of a finite Markov chain, its expectation $E[T_i(x_i, y_i)]$ can readily be obtained (Kemeny and Snell, 1976).

5.2. Phase 1: threshold take-backs

In the example in Section 3.2 the number of items taken back in the two library problem is the same or more as in the single library problem. In the TT heuristic we also take back the same or more as in a single library problem. At each library we carry out threshold take-backs similar to the example in Section 3.1. The thresholds are calculated by solving the MDP for the $n = 1$ case for each library.

The number of threshold take-backs is then as follows. Let $x_i^*(y_i)$ denote the single library threshold level for library i with y_i rented items. Starting with inventory x_i' and rented items y_i', the threshold take-back z_i'' for library i is

$$z_i'' = (x_i' - x_i^*(y_i'))^+.$$

This is the starting value for z_i''. We continue with this initial take-back decision in the reactive take-back phase.

5.3. Phase 2: reactive take-backs

In Section 3.2 we see that reactive take-backs are typically carried out when there are backorders, unless libraries with a backorder have a significant number of rented items. Therefore we first determine for each current backorder whether or not it should be met by a reactive take-back. The observations in Section 3.3 indicate that libraries with low stock-out probabilities are typically the best candidates from which to take back stock. We will therefore ship the items from the library with the longest expected time until stock-out.

In order to determine whether a reactive take-back is required, for each current backorder in the system we compare the expected backorder costs of no take-back with the transportation costs of a take-back. Suppose that library i has a backorder. As a measure of the expected number of periods until the first item returns, define

$$f_i(y_i) = \begin{cases} \dfrac{1}{py_i} & \text{if } y_i > 0 \\ \infty & \text{if } y_i = 0 \end{cases},$$

which is the reciprocal of the average number of returns per period with y_i rented items. If the choice is not to carry out a reactive take-back, a cost b would be incurred for an expected duration of $f_i(y_i)$ periods. Otherwise, a cost $2c + h_0 - h$ would be incurred once, which includes the holding cost difference from storing an item at the depot for one period.

Let $I^- = \{i : x_i' < 0\}$ be the set of libraries with backorders. Define by z_i^r the number of backorders at library $i \in I^-$ requiring reactive take-backs. We calculate this as

$$z_i^r = \underset{k \in \{1,\dots,(x_i')^-\}}{\arg\max} \{k : 2c + h_0 - h < bf_i(y_i + k - 1)\}, \tag{9}$$

or $z_i^r = 0$ if the condition is never satisfied. Here, z_i^r gives the maximum number of take-backs for which the direct cost of shipping are lower than the expected backorder costs. For the shipment of the k-th item, we assume that the previous $k - 1$ items have already been used to fulfill backorders. This accounts for the fact that shipped items can solve additional backorders in the future after completing a rental period. The total number of backorders requiring reactive take-backs is then

$$z_0^r = \sum_{i \in I^-} z_i^r.$$

Now we carry out take-backs from the libraries to reach a stock-level z_0^r at the depot. As stated, we take-back from libraries with the longest expected stock-out times. Let $I^+ = \{i : x_i' - z_i'' > 0\}$ be the set of libraries with stock on-hand. The candidate library j is then

$$j = \underset{i \in I^+}{\arg\max} \{E[T_i(x_i' - z_i'' - 1, y_i)]\}, \tag{10}$$

where $x_i' - z_i'' - 1$ is the new stock level after the projected additional take-back.

Now iteratively increase the z_j'' of each subsequent candidate library j by 1 until the depot's stock reaches z_0^r or stock of all libraries is depleted:

$$x_0' + \sum_{i=1}^{n} z_i'' \geq z_0^r \text{ or } \sum_{i=1}^{n} (x_i' - z_i'')^+ = 0.$$

Note that the condition $x_0' + \sum_{i=1}^{n} z_i'' \geq z_0^r$ may already be met due to the take-backs in Phase 1.

5.4. Phase 3: preventive take-backs

In the example of Section 3.2, a preventive take-back typically occurs when the depot has a significant probability of failing to meet demand during the next period, provided the take-back does not increase the stock-out probability of the candidate library too much. We apply that same idea in the TT heuristic. We balance the regret of not taking back an item when it is required elsewhere in the system, against the regret of taking back an item when it is demanded at the candidate library itself in the next period.

As initial step in the preventive take-back phase, we will correct the starting stock levels for reactive take-backs. The z_0^r items at the depot from the reactive take-back phase have already been reserved for dealing with existing backorders, so this stock cannot be used for preventing future backorders. Therefore, during the preventive take-back phase we consider the inventory levels

$$\hat{x}_i = (x_i' - z_i'')^+ \text{ and } \hat{x}_0 = (x_0' + \sum_{i=1}^{n} z_i'' - z_0^r)^+,$$

i.e., the current on-hand inventory at the libraries and the current stock at the depot not reserved for reactive take-backs.

Now we calculate the one-period regret and savings. Two events may occur. The first event is that an item is taken back from library j in the current period and it is demanded at that same library in the next period. Then the regret is $2c - h + h_0$, which consists of the transportation costs minus the decrease in holding cost for storing the item at the depot for one period. The second possible event is that no item is taken back from library j, but it is required by another library in the next period. Then the savings are $b + h - h_0$. This occurs whenever the item is not demanded at library j, while more than $\hat{x}_0$ shipments are required in total. The probability of the former event is

$$p_j = P\Big(D_j - R_j > \hat{x}_j - 1\Big),$$

where $\hat{x}_j - 1$ is the stock level after the take-back. The probability of the latter event is

$$P\Big(D_j - R_j \leq \hat{x}_j - 1\Big) P\Big(\sum_{i \neq j} (D_i - R_i - \hat{x}_i)^+ > \hat{x}_0\Big) = (1 - p_j)s_j$$

where $s_j = P\Big(\sum_{i \neq j} (D_i - R_i - \hat{x}_i)^+ > \hat{x}_0\Big)$.

The probability s_j may be difficult to compute because it is a convolution of the demand and return distributions of $n - 1$ libraries. We therefore approximate s_j for large n by assuming that library i requests shipments from the depot according to a Poisson distribution with demand rate

$$\tilde{\lambda}_i = -\log\Big(P\big(D_i - R_i \leq \hat{x}_i\big)\Big).$$

With this demand rate, the approximate probability of zero shipment requests from library i equals the exact probability, namely $P\big(D_i - R_i \leq \hat{x}_i\big)$. The probabilities for overflow, i.e., $D_i > R_i$, are approximate. In case $\hat{x}_i = 0, y_i = 0$, it is easily established that $\tilde{\lambda}_i$ equals the exact demand rate for shipments λ_i. The aggregate demand $\tilde{D}_0$ for shipments is Poisson$\big(\sum_{i \neq j} \tilde{\lambda}_i\big)$ distributed. The approximate value for s_j is therefore given by

$$\tilde{s}_j = P(\tilde{D}_0 > \hat{x}_0).$$

Table 3
Look-up table for the names of the heuristics.

Shipment heuristics	
SO	Sort by lowest rented items, highest backorders, and highest demand rate
SF	Ship to the first library with backorders
Take-back heuristics	
TT	Three phase heuristic
TTa	Three phase heuristic with an approximation for s_j
TB	Take-back according to a base-stock policy at the depot
TA	Take back all stock from the libraries

Table 4
Experimental design for each n.

#	h_0	b	c	B	λ_1	λ_2	λ_3	λ_4
1	0.5	10	1	1	0.05	0.05	0.05	0.05
2	0.5	10	4	2	0.15	0.15	0.15	0.15
3	0.5	10	7	3	0.25	0.25	0.25	0.25
4	0.5	15	1	1	0.15	0.15	0.25	0.25
5	0.5	15	4	2	0.25	0.25	0.05	0.05
6	0.5	15	7	3	0.05	0.05	0.15	0.15
7	0.5	20	1	2	0.05	0.25	0.15	0.25
8	0.5	20	4	3	0.15	0.05	0.25	0.05
9	0.5	20	7	1	0.25	0.15	0.05	0.15
10	0.8	10	1	3	0.25	0.15	0.15	0.05
11	0.8	10	4	1	0.05	0.25	0.25	0.15
12	0.8	10	7	2	0.15	0.05	0.05	0.25
13	0.8	15	1	2	0.25	0.05	0.25	0.15
14	0.8	15	4	3	0.05	0.15	0.05	0.25
15	0.8	15	7	1	0.15	0.25	0.15	0.05
16	0.8	20	1	3	0.15	0.25	0.05	0.15
17	0.8	20	4	1	0.25	0.05	0.15	0.25
18	0.8	20	7	2	0.05	0.15	0.25	0.05

We test the effectiveness of this approximation in the results section.

Candidates are selected, as in the reactive take-back phase, according to Eq. (10). A take-back from candidate library j is carried out whenever its expected savings exceed its regret, that is,

$$(2c - h + h_0)p_j \leq (b + h - h_0)(1 - p_j)s_j. \tag{11}$$

If (11) holds, then the z''_j of the candidate is increased by one. With the updated stock levels and a new candidate, we check the criterion until it is not satisfied or until $\sum_{i=1}^{n} \hat{x}_i = 0$. This gives the final value of the take-back decision z''.

5.5. Alternative take-back heuristics

The presented take-back heuristic has several complex steps, including the calculation of a single library policy using an MDP. In order to assess the added value of these complexities, we introduce two simple alternative heuristics for comparative experiments.

The Take-back Base-stock heuristic (TB) has base-stock level Q. Every review period it is ensured that the depot has Q units, if the total on-hand stock in the system so permits. Hence, we set

$$Z = \min\{Q - x_0, \sum_{i=1}^{n}(x_i)^+\}$$

as the total number of take-backs and iteratively ship these from the candidate library according to (10). This heuristic provides a practical and intuitive way to resupply the depot.

The Take-back All (TA) heuristic takes back all on-hand stock at the libraries to the depot. This is achieved by setting $z''_i = (x_i)^+$. The TA heuristic is a special case of the TB heuristic with $Q = K$.

6. Results

In this section we evaluate the performance of the shipment and take-back heuristics. We test the heuristics in instances with small n by comparing them with the optimal solution of the MDP. For large n we run the heuristics and compare them with each other. For reference, Table 3 summarizes the names and main concepts of the heuristics.

The experiments will be based on a Taguchi design (Taguchi, 1986), which, due to their orthogonality, are deemed suitable to test a wide range of parameter values with a relatively small number of scenarios. We use a Taguchi design with 18 scenarios and 3 levels per factor. The values for the parameters h_0, b, c, B, and $\lambda_i, i = 1, \ldots, 4$ in the scenarios are specified in Table 4. For instances with $n > 4$, we explain later how we specify the demand rates. While it seems obvious to vary the cost and demand parameters, we also vary B to observe the effect of the partial backordering on the effectiveness of the heuristics. In each scenario, we set $h = 1,\ \ell = 2b$ and $p = 0.3$. This value for p is the weekly return rate as estimated from a data set with 4 million library loans in the Netherlands in the year 2013. We later specify the choice of K. The value iteration algorithm from Section 2.5 is run with $\epsilon = 10^{-6}$.

6.1. Solution times of the MDP

We first discuss solution times of the MDP for instances of various size, shown in Table 5. We vary n and K while keeping the other variables at $h_0 = 0.5,\ c = 4,\ b = 10,\ \ell = 20,\ B = 2,$ and $\lambda_i = 0.25$ for $i = 1, \ldots, n$. The solution time includes all steps required to solve the MDP, such as generating the state space and the transition matrix. Table 6 shows the number of states in the same instances. The program has been implemented in Python and the instances were run on a Core i7-4770 CPU (3.3 GHz) with 16GB memory.

Instances with $n \leq 4$ are solved within several minutes. For $n = 5$ and $K > 7$, where instances have more than 1 million states, the solution times become several hours. For the instance with $n = 5$ and $K = 9$ the transition matrix exceeded memory limits. An important practical problem when solving instances is that when n increases by 1, K needs to increase as well to cover demand from the extra library. This reinforces the need for heuristics when n increases.

6.2. Performance of heuristics in small instances

We now compare shipment and take-back heuristics to the optimal policy in an experiment with small instances. The number of libraries in the scenarios varies between $n = 2, 3, 4$. For a specific n, only demand parameters λ_i, $i \leq n$ will be included. The number of items K is specified according to the rule

$$K = \left\lceil 1.8 \frac{\sum_{i=1}^{n} \lambda_i}{p} \right\rceil, \tag{12}$$

which can roughly be interpreted as having 1.8 times the stock required to meet the total system demand, scaled by the average return time $\frac{1}{p}$ of a rented item. The base-stock level Q of the TB policy is determined by calculating the average cost per time unit for $Q = 1, \ldots, K$ and taking the cost-minimizing value.

Table 7 shows for each value of n the average percentage difference of the heuristics from the optimal policy (obtained with the MDP from 2.5) in the 18 scenarios, as well as the maximum and the standard deviation. The most important column is SO/TTa, showing the performance of our combined shipment and take-back heuristic. In order to study performance of each shipment and take-back heuristic in isolation, we also measure the best-case performance of each individual heuristic in the columns SO,SF,TT,TTa,TB, and TA. The best-case performance is obtained by forcing the MDP to take the heuristic shipment or take-back decisions, while letting the other decisions be optimized freely. The

Table 5
Solution time in seconds of the MDP for different values of n and K.

	K							
n	2	3	4	5	6	7	8	9
2	0.01	0.01	0.02	0.04	0.06	0.08	0.11	0.16
3	0.04	0.09	0.20	0.37	0.72	1.57	3.22	6.16
4	0.46	1.40	4.04	9.72	21.86	47.07	103.58	221.69
5	9.54	36.76	112.93	331.93	921.63	2204.05	6159.06	–

Table 6
Number of states in the state space for different values n and K.

	K							
n	2	3	4	5	6	7	8	9
2	79	155	270	434	658	954	1335	1815
3	414	1000	2086	3934	6888	11,388	17,985	27,357
4	1917	5493	13,327	28,791	57,051	105,699	185,526	311,454
5	8208	27,198	75,183	182,887	404,162	828,432	1,597,882	2,930,642

Table 7
Statistics for the percentage difference from the optimal solution for various heuristics.

	n	SO/TTa	SO	SF	TT	TTa	TB	TA
Average	2	0.67%	0.00%	0.95%	0.63%	0.67%	3.58%	14.40%
	3	0.81%	0.11%	1.95%	0.64%	0.70%	2.52%	14.33%
	4	0.81%	0.26%	3.20%	0.54%	0.54%	1.87%	10.30%
Maximum	2	2.48%	0.02%	3.07%	2.48%	2.48%	11.79%	53.46%
	3	2.86%	0.35%	3.93%	2.85%	2.85%	7.41%	47.17%
	4	1.81%	0.66%	5.70%	1.59%	1.59%	4.97%	34.82%
St. Dev.	2	0.70%	0.01%	0.92%	0.70%	0.70%	3.91%	16.25%
	3	0.71%	0.10%	1.18%	0.72%	0.71%	2.19%	15.65%
	4	0.45%	0.21%	1.48%	0.47%	0.46%	1.48%	10.47%

percentage differences of each individual scenario are shown in Fig. 6. Since the differences for the TTa and TT heuristic are near-identical, Fig. 6 does not show the TT heuristic.

From Fig. 6 we can immediately see that the SO heuristic outperforms the SF heuristic in each scenario and for each value of n. Its average percentage difference over all scenarios is 0.12% and its maximum difference is below 1%. The SO heuristic seems to perform slightly worse as n grows, however shipping with priority to libraries with the least rented items seems appropriate in most instances. In any case, it outperforms shipping with priority to the first library with a backorder, as in the SF heuristic. The SF heuristic has an average percentage difference of 2.04% over all scenarios and its performance is highly volatile.

The take-back heuristics (TT, TTa, TB, and TA) typically have larger differences from optimality than the shipment heuristics because take-back decisions are more complex. Not only do we have to specify which candidates to take-back from, but also the amounts. From Fig. 6 it seems that the take-back heuristics have the following order in terms of performance: TTa outperforms TB, which in turn outperforms TA. The TA heuristic is on average 13.01% from the optimal solution, implying that there are high costs associated with overestimating the number of required take-backs. Taking back all on-hand stock can be optimal for scenarios with low demand rates at all libraries and low shipment costs, but leads to high costs in most other scenarios. The Take-back Base-stock heuristic (TB) has a relatively large average difference of 2.66% over all scenarios. Since the TB heuristic has a static rule independent of the state, it is incapable of adapting to the situation when required, leading to large differences in some scenarios. However, its percentage differences seem to decrease as n increases.

The three phase take-back heuristics, TT and TTa, are within 1% from optimality on average and all differences are below 3%. In most scenarios TTa and TT have the same percentage differences, although in some scenarios there are minor differences. Overall the choice between using the exact or approximate stock-out probability seems to have negligible effect on costs. When combining the SO and TTa heuristic, we see in the column SO/TTa that the gap with the optimal solution remains below 1% on average. In most scenarios the total percentage difference from this combined policy is close to the sum of the percentage differences of the individual policies. The interaction between the shipment and take-back heuristic thus appears limited.

6.3. Take-back heuristics in larger instances

In order to gain insight in performance of the take-back heuristics in instances with a higher number of libraries, we carry out a simulation experiment. The costs of the TTa, TA, and TB heuristic are compared with each other in instances with $n = 5, 10, 20, 50,$ and 100 libraries. For all these take-back heuristics, we take the shipment decisions according to the SO heuristic. Common random numbers are used, to the extent possible, to reduce the variability of the results. All heuristics face common Poisson demand D_t in period t of a given simulation run. Since the state variable y_t varies between heuristics, we chose not to take common random returns R_t. Instead, each R_t is drawn independently from a Binomial distribution.

As before, the experiment follows the design from Table 4. Since there are only four values for demand rates in this table, we need a new way to specify the demand rates. The following method is applied. The base demand rate for each library is λ_1 from Table 4, now denoted $\bar{\lambda}$. For each library, we set

$$\lambda_i = \bar{\lambda} + u_i,$$

with $u_i \sim N(0, 0.2\bar{\lambda})$ distributed. Adding this normal random noise leads to libraries with varied demand rates in the experiment. The number of items K is again set according to (12).

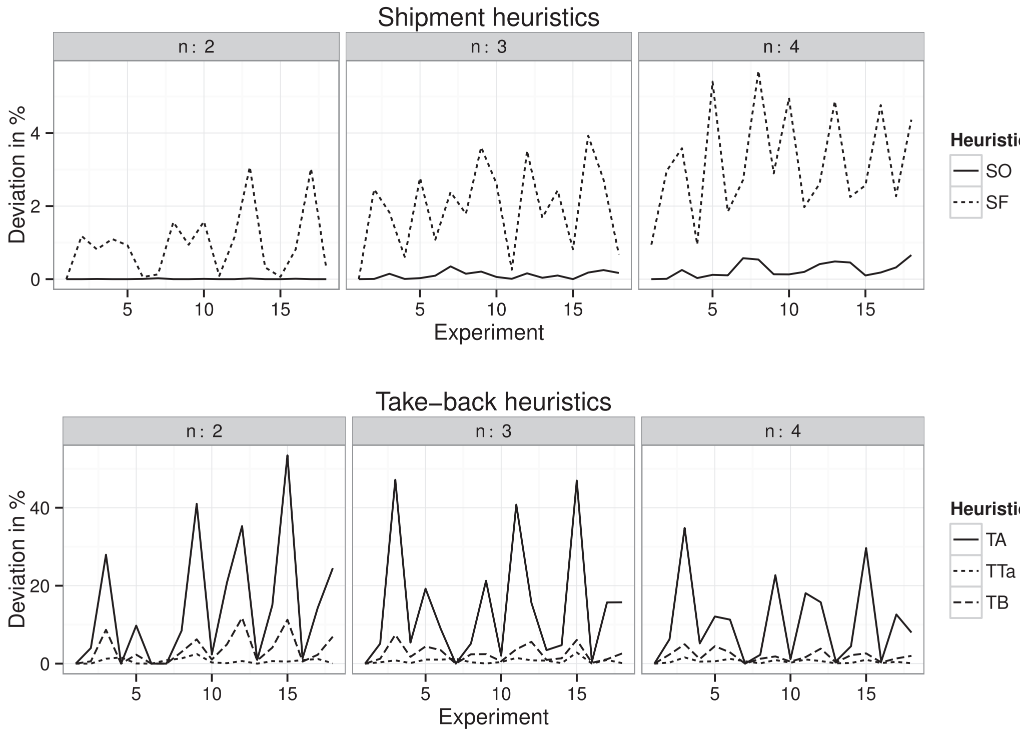

Fig. 6. Comparison of the percentage differences for the shipment and take-back heuristics.

Table 8
The percentage increase in costs for the TA and TB heuristic compared to the TTa heuristic for a varying number of libraries.

	Cost increase of TA in %					Cost increase of TB in %				
exp\\n	5	10	20	50	100	5	10	20	50	100
1	−0.11	0.13	0.10	0.01	0.02	−0.01	0.01	0.59	0.51	0.39
2	10.13	9.58	12.36	16.59	19.03	2.30	1.52	0.36	0.22	0.10
3	48.38	50.96	59.30	66.53	69.84	1.83	2.01	0.47	0.28	0.34
4	1.54	2.52	3.99	7.00	8.31	1.07	0.80	0.41	0.17	0.14
5	20.63	27.04	33.83	41.75	45.90	2.49	1.48	0.99	0.39	0.18
6	−0.57	0.17	0.12	1.30	2.03	−0.55	0.19	−0.08	0.05	−0.01
7	0.18	0.13	0.12	0.19	0.05	0.07	1.19	0.83	2.14	1.62
8	3.44	5.90	9.18	14.71	17.63	0.31	0.72	0.41	0.13	0.07
9	51.13	62.39	71.04	80.99	84.38	1.05	−0.11	−0.29	0.09	0.14
10	4.03	5.97	8.63	11.73	12.93	0.93	0.54	0.33	0.12	0.01
11	0.41	0.91	1.14	2.34	3.42	0.40	0.12	0.28	0.14	0.04
12	22.52	25.37	27.58	32.35	35.67	0.92	1.69	0.29	−0.02	−0.11
13	3.90	7.88	10.63	14.71	16.10	1.21	0.78	0.39	0.16	0.04
14	−1.20	0.17	0.53	1.28	2.24	−1.02	0.03	0.11	0.03	0.03
15	25.64	26.71	31.82	36.79	40.63	0.84	0.38	−0.16	−0.31	−0.26
16	−0.42	0.34	1.22	3.18	4.24	0.14	−0.10	0.04	−0.17	−0.16
17	41.42	52.10	61.05	74.76	78.78	0.55	−1.21	−0.23	−0.25	−0.06
18	0.23	1.44	1.55	2.36	3.95	0.72	0.51	−0.01	0.21	0.12

For each scenario in the L18 array, we take 10 different random draws of the demand rates. For each random draw, 1000 simulation runs are carried out, giving 10,000 runs in total for each scenario. In every run the system is simulated for 1000 weeks, excluding a warmup period of 100 weeks. The starting state for each policy is $x_0 = K$, $x_i = 0$ for $i = 1, \dots, n$. For each random draw, the base-stock level Q of the TB policy is determined by simulation. For $Q = 1, \dots, K$, we calculate the average costs of 100 simulation runs and stop as soon as the average costs increase.

Table 8 shows the outcome of the experiment. The columns for the TA and TB heuristic give the percentage cost increase in each scenario relative to the TTa heuristic (averaged over the 10 random draws). For each n, Table 9 summarizes the average, maximum, minimum and standard deviation over all 180 combinations of scenarios and draws. On average we see that the TTa heuristic is better than the TA and TB heuristic. The negative minima indicate that the TTa heuristic can be worse than TA and TB in some combinations of scenarios and draws, however, this percentage is small and decreases quickly in n.

The TA heuristic leads, as before, to relatively high costs. The TA heuristic has the lowest backorder and holding costs of all heuristics, but this is at the expense of extreme transportation costs. Because libraries hold no stock, there may also be costs for lost demands. Taking back all stock seems appropriate in scenarios with low demand. In these scenarios the single library thresholds are zero, hence all heuristics lead to the same take-back decisions. However, the cost differences in scenarios with higher demand are significant. The performance of the TA policy becomes relatively worse compared to the TTa heuristic as the number of libraries increases, because for high n the total system demand has relatively low variance, requiring a lower amount of stock at the depot.

Table 9
Statistics of the TA and TB heuristic over all scenarios and random draws for a given n.

	Average		Minimum		Maximum		St. Dev.	
n	TA	TB	TA	TB	TA	TB	TA	TB
5	12.85%	0.74%	−2.41%	−2.14%	64.36%	3.74%	17.90%	1.09%
10	15.54%	0.59%	−1.37%	−1.42%	71.82%	6.68%	20.27%	0.96%
20	18.57%	0.26%	−1.43%	−0.63%	79.56%	3.33%	23.10%	0.48%
50	22.70%	0.22%	−0.16%	−0.42%	87.89%	5.89%	26.33%	0.68%
100	24.73%	0.14%	−0.06%	−0.36%	86.58%	4.76%	27.43%	0.51%

It should be clear that the TB policy performs reasonably well in this simulation. While the TTa policy is better than the TB policy in many of the scenarios, the differences becomes smaller as n increases. For small n demand is volatile, requiring a policy which is capable of adapting to the situation. For large n an invariant base-stock level will usually suffice, because demand for shipments from the depot becomes quite constant. This can be regarded as a certain kind of pooling of demand. The TB policy could therefore be a suitable alternative in situations with a large number of libraries.

7. Conclusion

This paper considers a library system with a low-cost depot from which items are shipped in case of stock-outs at the libraries. In contrast to regular inventory systems, 100% of the rented items in a library system return. Therefore, restocking of the depot must be achieved by taking back items from the libraries. We formulate and solve an MDP for several scenarios and use the obtained insights to create heuristics for large size problems. With this work we target two specific issues that are of interest to library organizations with such a system: storage of low-demand items and resupplying the depot for future shipment requests.

The single library problem provides some insights into storage of low-demand items. We observe that it is optimal to take back all on-hand items above a certain threshold which decreases in the number of rented items. Knowledge of this take-back policy can assist in practice in making decisions for removing rental items from libraries to create room for new and popular rental items.

The optimal resupply of the depot follows from studying the MDP. We find that the concepts of reactive and preventive pooling from inventory theory are of importance in library systems: we observed combinations of reactive and preventive take-backs. These take-backs are largely explained by economic trade-offs between backorder costs and shipment costs. For library systems with many libraries, we have shown that a base-stock policy at the depot may yield reasonable results. The above results have been communicated with various public library organizations in the Netherlands.

The results seem also to apply to more general rental systems with a depot. In case there is full backordering in those systems, we would need to increase the partial backorder level to a point where sales are not lost. Since the results and heuristics are mostly focused on situations with low demand rates, some adjustments may be required for higher demand rates.

As a future research direction one could consider substitution of products in case of stock-outs. The state space with multiple items is in general extremely large: if a single item state space has size $|\mathcal{S}|$, an m item state space has roughly $|\mathcal{S}|^m$ states. Therefore, it seems most practical to study substitution in a setting with small $|\mathcal{S}|$ and m, for example a setting with one rental location, one depot, and two item types. We expect that the fact that customers can substitute products reduces the number of required shipments and take-backs. An alternative direction is to include advance demand information in the model. In some rental systems there are two types of demand: regular demand and advance demand to be picked up by a customer in a specified later period. It is of interest to optimize decisions when there are two such demand streams which have possibly different backordering costs. This could lead to stock rationing policies at the depot for customers with high backorder costs. Finally, one could investigate optimal decisions when customers return items to a different location than the original rental location, which occurs frequently in car rental systems and occasionally in public library systems.

Acknowledgments

This research has been funded by Dinalog, the Dutch Institute for Advanced Logistics. We would like to thank the two anonymous referees for their helpful comments.

References

Axsäter, S., Howard, C., Marklund, J., 2013. A distribution inventory model with transshipments from a support warehouse. IIE Trans. 45 (3), 309–322.
Dell'Amico, M., Iori, M., Novellani, S., Stützle, T., 2016. A destroy and repair algorithm for the bike sharing rebalancing problem. Comput. Oper. Res. 71 (1), 149–162.
Ernst, A., Gavriliouk, E., Marquez, L., 2011. An efficient lagrangean heuristic for rental vehicle scheduling. Comput. Oper. Res. 38 (1), 216–226.
George, D., Xia, C., 2011. Fleet-sizing and service availability for a vehicle rental system via closed queueing networks. Eur. J. Oper. Res. 211 (1), 198–207.
Kemeny, J., Snell, J., 1976. Finite Markov Chains. Springer, Berlin.
Köchel, P., 2007. Order optimisation in multi-location models with hub-and-spoke structure. Int. J. Prod. Econ. 108 (1), 368–387.
Li, Z., Tao, F., 2010. On determining optimal fleet size and vehicle transfer policy for a car rental company. Comput. Oper. Res. 37 (2), 341–350.
Muckstadt, J., Thomas, L., 1980. Are multi-echelon inventory methods worth implementing in systems with low-demand-rate items? Manage. Sci. 26 (5), 483–494.
Song, D., Carter, J., 2008. Optimal empty vehicle redistribution for hub-and-spoke transportation systems. Nav. Res. Logist. 55 (2), 156–171.
Taguchi, G., 1986. Introduction to Quality Engineering: Designing Quality into Products and Processes. Asian Productivity Organization, Tokyo.
Tijms, H., 2003. A First Course in Stochastic Models. John Wiley & Sons, Chichester.
Van der Heide, G., 2015. Inventory Control for Multi-location Rental Systems. Ph.D. Thesis. University of Groningen.
Van der Heide, G., Roodbergen, K., 2013. Transshipment and rebalancing policies for library books. Eur. J. Oper. Res. 228 (2), 447–456.
Van Wijk, A., Adan, I., Van Houtum, G., 2013. Optimal allocation policy for a multi--location inventory system with a quick response warehouse. Operat. Res. Lett. 41 (3), 305–310.