Prioritizing Replenishments of the Forward Reserve Area

Harwin de Vries*,

Ruth Carrasco-Gallego †,

Taoying Farenhorst-Yuan, Rommert Dekker ‡

May 9, 2012

Report Econometric Institute EI2012-06

Abstract

Having sufficient inventories in the piece-picking area of a warehouse is an essential condition for warehouse operations. This requires a timely replenishment of the products from a reserve area in case they could run out of stock. In this paper we develop analytical models to arrive at priority rules for these replenishments in case replenishments and order picking are done simultaneously because of time pressure. This problem was observed in a warehouse of a large cosmetics firm. The priority rules are compared by means of simulation and regression. Finally we present the results of applying one of these rules in practice.

*Econometric Institute, Erasmus University Rotterdam, P.O. Box 1738, NL-3000 DR, Rotterdam, The Netherlands. E-mail: harwindv@hotmail.com.

†Escuela Técnica Superior de Ingenieros Industriales, Universidad Politécnica de Madrid, C\José Gutierrez Abascal 2, 28006 Madrid Spain. E-mail: ruth.carrasco@upm.es.

‡Econometric Institute, Erasmus University Rotterdam, P.O. Box 1738, NL-3000 DR, Rotterdam, The Netherlands. E-mail:{farenhorst,rdekker}@ese.eur.nl.

1 Introduction

Warehouses are one of the most important elements in physical distribution and in the last decades, much research has been done about them, see Gu et al. (2007) and de Koster et al. (2007) for reviews. Among all warehouse activities, order picking is getting most attention. It is estimated that more than fifty percent of the operational costs are related to the order picking process (Coyle et al. (1996)), which makes optimization in this field very beneficial. These optimizations are performed by the development of (combinations of) order batching, order picking routing and rack assignment strategies.

The increasing popularity of e-commerce has made order picking even more important, as warehouses are faced with quite late order cut-off times and the time for picking is short. One of the strategies employed to fasten order picking is to operate a so-called forward-reserve storage area from which the most demanded products can be picked quickly, see Gu et al. (2007) and de Koster et al. (2007). In this area one does piece picking instead of bulk picking. In order to let many products benefit from such an area, the stocks in this area are limited and they are replenished from a bulk reserve area from the back of the warehouse. Often, these replenishments are planned following a so-called (s, S)-policy. In this policy the stock of a product in the forward area is replenished up to the level S as soon as it drops below s items (s and S are specified for each product, based on long term demand information and rack space).

However, since time pressure forces many warehouses to simultaneously perform order picking and replenishment operations, it is not infrequent to find that an order picker has to pick a product before the replenishment crew had the time to replenish it and thus faces a stockout (which we will call a 0-pick). We observed this problem in a warehouse belonging to a renowned luxury cosmetics firm.

The problem mentioned above can be addressed if each product's short-term demand is known. This information makes it possible to identify products which may cause a 0-pick if not replenished soon enough. In the warehouse studied in this paper, this demand information is available because the warehouse uses a wave-picking strategy, in which 'waves' of orders are released in succession. Namely, at the beginning of a wave, every order line that will be picked in that wave is known. Predicting the exact moment at which the products will run out of stock is however difficult, as there can be quite some variation in the order picking times and the exact order pick moment depends on the routes taken.

Although there is some literature about forward-reserve areas, it mainly concerns its size and the quantity of replenishments, but not about prioritizing the replenishments. There do exists resemblances with inventory-routing problems and our results could be used in that area as well. In this paper, three policies are proposed that prioritize replenishment orders based on the short-term demand information mentioned above. The goal of these policies is to minimize the number of 0-picks. We first prove optimality results of some policies and next we compare them using simulation. Finally, the real-life results of implementing one of these three replenishment policies in a warehouse are presented.

This paper is organized as follows. Section 2 contains a detailed description of the problem. In section 3 a short overview of the literature dealing with this problem is given. Section 4 describes the problem in a more stylized way, namely by means of assumptions. Based on these assumptions, three replenishment policies are proposed in section 5. In section 6, these policies are compared by means of simulation. Section 7 contains the results of implementing one of the three stock replenishment policies. Finally, section 8 states a few general conclusions that can be drawn from this paper.

2 Problem Description

As mentioned in the introduction, the analyzed warehouse uses a forward-reserve storage strategy. The *forward area*, also called the piece picking area, is restricted to a separate area of the warehouse, while bulk or mass storage takes place in the so-called *reserve area*. The biggest advantage of this storage strategy is that it decreases the order pick times. The distances the order pickers have to cover are relatively small, because the order picking process only takes place in a limited part of the warehouse. Another advantage is that products can be easily stocked in different quantities (e.g. per box or per pallet). The biggest disadvantage is that the stock in the piece picking area has to be replenished from the reserve area.

The picking policy applied in this warehouse is wave-picking. Each picking wave corresponds to a predefined shipping schedule, meaning that all orders which have to be shipped together are released at the same moment in a certain order. There is no information about the time at which an individual order line will be handled, except for the fact that it takes place somewhere between the beginning and the end of the wave.

A replenishment order corresponds to *one* product that is replenished. In this paper we associate a priority to every replenishment order and execute the one(s) with the highest priority first. Thanks to online control of the (piece picking) inventory, the inventory level in this warehouse is updated continuously. This makes it possible to launch automatic replenishment orders from the *Warehouse Management System* (WMS) when the stock level of a given product declines to a predetermined limit level. In this case, 30% of the total inventory capacity of the rack is taken as the limit. This replenishment policy is often called the (s, S)-policy. Here s denotes the reorder point (in the case taken as 30% of the rack capacity), and S the so-called order-up-to level (the rack capacity). The product-specific values of s and S are periodically reviewed by the warehouse management by solving the so-called *forward-reserve problem* (see for example Hackman et al. (1990) and Frazelle et al. (1994)). As soon as a replenishment order is placed, it is placed at the bottom of the list of replenishment orders, meaning that a FIFO (first-in-first-out) strategy is used.

Replenishment workload is divided depending on the destination point. The piece picking area is split up in three U-shaped zones, and each of these zones is replenished by one forklift driver. Each driver receives the replenishment orders by means of a print-out. This list of replenishment orders is edited as soon as the WMS detects a need for replenishment. These replenishment orders are executed in small batches to reduce travelling time. Because of time pressure, the picking and replenishment operations take place throughout the working day in a continuous and simultaneous way.

However, although the replenishment orders launched by the WMS were diligently executed by the forklift drivers, it was not infrequent to find that the more frequently demanded products in a given picking wave were out of stock before the replenishment crew has had the time to replenish them. There are two main causes of this problem. First, the stock levels in the forward area are relatively small due to the restricted amount of space that is assigned to this area. The second cause can be found in the relatively large variation in a product's demand, which is mainly because the orders coming from the same class of retailers are often prepared together. Hence one of the assumptions underlying the calculation of the s and S parameters, being a stationary demand process, is violated, yet an alternative calculation is very difficult.

As scientific literature could not provide a replenishment policy with *simultaneous* picking and replenishment operations aiming to avoid 0-picks in the forward area, a new replenishment policy had to be designed.

3 Literature Review

There is hardly any literature available about the problem described in section 2. Some more details on the problem, but no analysis of the priority rules, can be found in Carrasco-Gallego & Ponce-Cueto (2009). Particularly the combination of wave-picking and replenishing during the order-picking process makes our problem very specific. We only found two papers that can be applied (partly) to our situation.

Gagliardi et al. (2008) also consider a warehouse with a forward-reserve storage strategy and a pick-to-belt operation. Four heuristic replenishment policies are proposed in that paper. Two of them are based on long-term demand, while the other two take information about short-term demand into account by checking incoming orders. He considers only the next product to be replenished, while we consider wave-picking and set priorities for all products to be replenished by several people.

Another paper that has to be mentioned is the one of Van den Berg et al. (1998), which is also about replenishment policies. This paper assumes replenishments during idle periods. Therefore, the priorities of the replenishment orders is irrelevant, opposed to our situation. The goal of their replenishment policy is to determine which (unit-load) replenishments minimize the expected amount of labor (picking-time and replenishment-time) during the picking period. So, the emphasis lays on the costs of picking and replenishment. In this paper, we concentrate on the costs of 0-picks only.

The papers described above focus on the replenishment process. Alternatively, the problem of 0-picks could be tackled by optimizing the inventory level of the products in the piece picking area. The problem of determining which products should be stored in the forward area and in what quantities is well-known as the forward-reserve problem. Hackman et al. (1990) were the first who presented a heuristic for this problem, aiming to minimize both the replenishment costs and the order-pick costs. Later, this research was extended by Frazelle et al. (1994) by also determining the optimal size of the forward area. Again, the emphasis lays on the costs of picking and replenishment.

Although the problem originates from replenishment in warehouses, one may also encounter it in other situations where stock replenishments have to be prioritized, e.g. in case retailers need to be replenished under a vendor-management inventory scheme and the transport capacity is limited. In the so-called inventory routing problems (see Moin & Salhi (2007), retailer's replenishments are combined with determining vehicle routings. Here one plans the vehicles for multiple days, while taking information on the inventory into account. The inventory evolution is predicted with deterministic or stochastic models, the latter being much more complex than the first one. One predicts per day whether there will be a stockout and use that information to prioritize the replenishment for certain days. Next one does a routing where one looks at routing efficiency versus extra inventories caused by replenishments done too early. An example using simulation optimization is from Vonolfen et al. (2011). In our case we have a continuous execution rather than a day planning and we take more demand information into account. Next, we do not model the replenishment vehicle routing, as that would make the model very complex. Finally, our approach is analytical with simulation used for performance evaluation.

4 Model

In this section we present a model for the problem of minimizing 0-picks described in section 2 by means of assumptions about the warehouse and its operations:

- A1: The warehouse uses a forward-reserve storage strategy.
- A2: Real-time information about the piece picking inventory is known.
- A3: The warehouse uses a wave-picking strategy.

- A4: Replenishment orders are executed during the order picking process.
- A5: The priority of a replenishment order is set at the beginning of the wave, and not changed afterwards.

The first four assumptions form necessary conditions for this paper to be directly applicable. It is important to notice that A2 provides information about the inventory level in the piece picking area and that A3 provides the short-term demand information. Together these assumptions make it possible to identify those products whose stock levels are insufficient to fulfill the needs in the wave. We will call these products *emergency products* from now on.

The fifth assumption requires some extra explanation. Replenishment orders and their priorities can be determined either continuously, or periodically. In the first case, these are updated by continuously reviewing information about demand and inventory. In the latter case, replenishment orders and their priorities are set once in a certain period (for example: once in a wave), and not changed afterwards. Although the first approach may be better in terms of 0-picks, it has some major disadvantages. First of all it is very hard to implement. Second, this approach may cause many changes of replenishment orders during a wave, which may cause a lot of stress among the replenishment crew. That is why assumption A5 is made.

5 Replenishment Policies

Now we have framed the problem of minimizing 0-picks by means of assumptions, three different stock replenishment policies will be proposed, aiming to minimize the number of 0-picks in a given wave. Before the replenishment policies are explained, some notation is given that is required for understanding those policies.

5.1 Notations

- T: the duration of the wave.
- H: the set of the emergency products; and its elements $h \in \{1, 2, \ldots, |H|\}$.
- N_h: the number of orders containing product h in the wave.

- Π_h: the set of all possible orderings in which all N_h order lines containing product h in the wave can be picked. This means $N_h!$ permutations; and its elements π_h.
- $Q_h(k)$: the quantity required of product h, in the kth time product h is picked in the wave $(1 \leq k \leq N_h)$. Obviously $Q_h(k)$ depends on π_h.
- $S_h(k)$: the inventory level of product h before the kth time product h is picked in the wave $(1 \leq k \leq N_h)$. Obviously $S_h(k)$ depends on π_h.
- I: the set of the replenishment orders that will be executed in the wave; and its elements $i \in \{1, 2, \ldots, |I|\}$.
- t_i: time at which replenishment order i is executed ($t_i \leq t_j$ for all $i < j$).
- $X_h(t_i)$: the number of 0-picks occurring for product h in the wave, in case that product h is replenished at time t_i.
- $f_{hi} = \mathbb{E}[X_h(t_i)]$: the expected total number of 0-picks occurring for product h in the wave, in case that this product is replenished at time t_i.
- $Y_h(t_i)$: the number of times product h is picked in the wave (0-picks included) before time t_i.
- The decision variable:

$$z_{hi} = \begin{cases} 1 & \text{if emergency product } h \text{ is assigned to replenishment } i, \\ 0 & \text{otherwise .} \end{cases}$$

5.2 Stock Replenishment Policies

In this section, three replenishment policies will be explained. The idea of these replenishment policies is to assign priorities to replenishment orders based on the expected number of 0-picks of a product. Each of the policies uses different amount of information from the WMS and different assumptions, so that these policies differ in complexity.

5.2.1 Stock-Needs Rule (SNR)

The first replenishment policy is the least complex one. The idea of this replenishment policy is to start identifying the emergency products. This can be done because of assumptions A2 and A3. After that, different levels of priority are set among these products. These priorities are set based on the following reasoning: the lower the *stock* level of a given product in the piece picking area (i.e., $S_k(1)$) compared to the registered *needs* for that product in a certain wave $\left(\text{i.e., } \sum_{k=1}^{N_h} Q_h(k)\right)$, the higher the chances of having a 0-pick for that product in that wave. Therefore, the ratio *stock / needs* $\left(\text{i.e., } S_k(1) \Big/ \sum_{k=1}^{N_h} Q_h(k)\right)$ is used to assign different replenishment priorities among the emergency products. The emergency product which has the lowest value for this ratio gets the highest priority, the one with the second-lowest value gets the second-highest priority and so on. These priorities of the replenishment orders are set at the beginning of the wave and executed in small batches afterward. After all emergency products are replenished, forklift drivers can keep on performing replenishment operations following the traditional replenishment policy (i.e. the (s, S)-policy). The replenishment policy described above will be called the *Stock-Needs Rule* (SNR) from now on.

5.2.2 Order-Quantity-Based Rule (OQBR)

Whereas the SNR only looks at the stock-level compared to the aggregate demand in a wave, the replenishment policy presented next uses *all short-term demand information* that is available. Namely, it models the expected number of 0-picks occurring for a certain product as a function of the time it is replenished by taking all individual order lines into account. Given these functions of all emergency products, a Linear Assignment Problem (LAP) can be solved to minimize the total expected number of 0-picks. However, in order to do this, some extra assumptions are required:

- A6: The duration of a wave is known in advance
- A7: The time at which an order line will be handled (pick time), has a uniform distribution between the starting-time and the end-time of a wave.
- A8: The pick times of the order lines containing a certain product are independent of each other.

- A9: The way in which replenishments are batched is independent from labor costs
- A10: The time at which the ith replenishment order is executed, t_i, can be predicted at the beginning of the wave and is independent of the product that is replenished.
- A11: The inventory levels in the warehouse (forward + reserve) are sufficient to complete the customers' orders.
- A12: After a product is replenished, the inventory level is at least such that it fulfills the needs during the rest of the wave.
- A13: If the stock-level of a certain product is insufficient to fulfill the requirements of an order line, nothing will be picked.

Some of the assumptions stated above (e.g. A6, A8, or A10) may not be met in reality. For example, the duration of a wave and the replenishment times may have to be estimated. However, these assumptions have to be made in order to simplify the problem and to model it.

Generally, there is no information about the time at which an individual order line is handled. As in the case described in section 2, the only information one has is that it will be somewhere between the beginning and the end of the wave. This means that we have no reason for differentiating "pick-probability" between different time units, which explains assumption A7.

Assumption A9 is made in order to keep focusing on costs of 0-picks. If the problem would be extended to both minimizing costs of 0-picks and costs of replenishment, calculations in the replenishment policy presented next would become too complex. If a warehouse doesn't meet assumptions A11 till A13, some little adaptations in the replenishment policy presented below can be made so that these settings are taken into account.

Based on assumptions A1 till A13, we can start modeling the problem of minimizing the total expected number of 0-picks in a certain wave now. Due to assumption A10 we know that the set of replenishment orders, as well as the times at which they are executed, is known at the beginning of the wave. This means that the priority of a replenishment order is automatically determined by its replenishment time (the product which corresponds to the replenishment order that has the earliest replenishment time has the highest priority). Observing this, the problem of assigning replenishment priorities to emergency products in order to minimize the costs created by 0-picks in a given wave is the same as 'assigning' emergency products to pre-defined replenishment orders in such a way that the expected number of 0-picks is minimized. Furthermore, notice that the way in which these replenishments are batched is independent from costs (A9), and that t_i is independent of the product replenished (A10). Therefore, the problem of assigning optimal replenishment priorities to emergency products can be modeled as an assignment problem:

$$\min \sum_{h\in H}\sum_{i\in I} z_{hi} f_{hi} \tag{1}$$

$$s.t. \sum_{h\in H} z_{hi} \leq 1, \forall i \in I \tag{2}$$

$$\sum_{i\in I} z_{hi} = 1, \forall h \in H \tag{3}$$

$$z_{hi} \in \{0,1\}, \forall i \in I, \forall h \in H. \tag{4}$$

The objective function tells us that we want to minimize the expected costs caused by 0-picks of all emergency products. Restriction (2) makes sure that at most 1 emergency product is assigned to a replenishment order. Restriction (3) tells us that every emergency product has to be replenished.

In case that $|H| > |I|$, restriction (2) should be changed into

$$\sum_{h\in H} z_{hi} = 1, \forall i \in I,$$

and restriction (3) should be changed into

$$\sum_{i\in I} z_{hi} \leq 1, \forall h \in H,$$

so that exactly one emergency product is assigned to every replenishment order.

The binary programming model described in (1)-(4) is an NP-hard problem (Martello & Toth (1987)). Moreover, this kind of problems is hard to implement in a WMS, because complicated solution tools are required for solving this problem. However, observe the following: given that $|H| \leq |I|$, the emergency products will be assigned to the first $|H|$ replenishment orders in order to minimize the total expected number of 0-picks. Therefore we can leave the other replenishment orders out of consideration. Let us call the new set of replenishment orders I^*. As $|H| = |I^*|$, the inequality of restriction (2) can be replaced by an equality. By doing this, we translate the problem described above into a classical *Linear Sum Assignment Problem* (LSAP) (Burkard & Çela (1999)). Similarly, in case that $|H| > |I|$, the problem can still be modeled as an LSAP, by adding $(|H| - |I|)$ extra imaginary replenishments at $t_i = T$. Let us call the new set of replenishment orders I^{**}. By this trick, it holds that $|H| = |I^{**}|$, which makes (1)-(4) an LSAP again. After the assignment, the products that are assigned to the imaginary replenishments must be erased from the replenishment list.

There are several algorithms that can solve an LSAP very fast, and that are relatively easy to implement. For example, the Hungarian algorithm (Kuhn (1955)) or the Munkres Algorithm (Munkres (1957)). Alternatively, the problem could be solved by means of heuristics. For example, an ordered list heuristic in which, starting with the first replenishment order, sequentially the unassigned emergency product with the highest value for f_{hi} is assigned , see De Vries (2010) for more details. In order to improve the heuristic solution, a local search can be applied afterward.

In order to solve the LSAP, the parameters f_{hi} have to be known. These parameters can be calculated based on assumptions A1-A13. Another way of calculating this parameter will be presented in section 5.2.3. In order to distinguish them, we give this one a superscript *OQBR* (Order-Quantity-Based Rule: this policy takes the quantity required in each order line into account), which is the name of the replenishment policy that uses this version of f_{hi}.

$$
\begin{aligned}
f_{hi}^{OQBR} &= \mathbb{E}^{OQBR}[X_h(t_i)] = \sum_{j=1}^{N_h} \mathbb{E}\left[X_h(t_i) | Y_h(t_i) = j\right] \cdot \mathbb{P}(Y_h(t_i) = j) \\
&\overset{A6-A8}{=} \sum_{j=1}^{N_h} \mathbb{E}\left[X_h(t_i) | Y_h(t_i) = j\right] \cdot \binom{N_h}{j} \cdot \left(\frac{t_i}{T}\right)^j \cdot \left(\frac{T - t_i}{T}\right)^{N_h - j} \qquad (5) \\
&= \sum_{j=1}^{N_h} \sum_{\pi_h \in \Pi_h} \mathbb{E}\left[X_h(t_i) | Y_h(t_i) = j, \pi_h\right] \cdot \mathbb{P}(\pi_h) \cdot \binom{N_h}{j} \cdot \left(\frac{t_i}{T}\right)^j \cdot \left(\frac{T - t_i}{T}\right)^{N_h - j} \\
&\overset{A6-A8}{=} \sum_{j=1}^{N_h} \sum_{\pi_h \in \Pi_h} \mathbb{E}\left[X_h(t_i) | Y_h(t_i) = j, \pi_h\right] \cdot \frac{1}{N_h!} \cdot \binom{N_h}{j} \cdot \left(\frac{t_i}{T}\right)^j \cdot \left(\frac{T - t_i}{T}\right)^{N_h - j} \\
&\overset{A11-A13}{=} \sum_{j=1}^{N_h} \sum_{\pi_h \in \Pi_h} \sum_{k=1}^{j} \mathbb{1}_{S_h(k) < Q_h(k)} \cdot \frac{1}{N_h!} \cdot \binom{N_h}{j} \cdot \left(\frac{t_i}{T}\right)^j \cdot \left(\frac{T - t_i}{T}\right)^{N_h - j} \qquad (6)
\end{aligned}
$$

where

$$S_h(k) = \begin{cases} S_h(1) - \sum_{n=1}^{k-1} Q_h(n) \cdot \mathbb{1}_{S_h(n) > Q_h(n)} & \text{if } k > 1, \\ S_h(1) & \text{otherwise }. \end{cases}$$

In order to calculate the expected number of *0-picks before replenishment* ($\mathbb{E}^{OQBR}[X_h(t_i)]$), this number is conditioned on *the number of picks before replenishment* ($Y_h(t_i)$), and multiplied by the probability that this situation occurs ($\mathbb{P}(Y_h(t_i) = j)$). This probability is easy to calculate if you observe that under assumptions A6-A8, this situation is a binomial probability experiment of N_h trials (with $p(\text{success}) = t_i/T$, and $p(\text{failure}) = (T - t_i)/T$,). Next, the expected number of 0-picks given the number of picks before replenishment ($\mathbb{E}\left[X_h(t_i)|Y_h(t_i) = j\right]$) is conditioned on the ordering in which the N_h order lines containing product h are picked (π_h), and multiplied by the probability that this ordering occurs. Because of assumptions A6-A8, all permutations have the same probability. After that, given the number of picks before replenishment ($Y_h(t_i)$), given the ordering of the picks (π_h), the number of 0-picks can be counted, which is done by the indicator function ($\mathbb{1}_{S_h(k) < Q_h(k)}$). This indicator function will be equal to one if an order picker meets insufficient stock to fulfill the needs of its order (A13) and if $k \leq j$. This way we deal with the assumption that a 0-pick can only occur before replenishment (A11, A12). Because of assumptions A2 and A3, all parameters $S_h(k), Q_h(k)$, and N_h are known. Furthermore, A6 provides T and A10 provides t_i for every i.

As we already showed, the problem of minimizing the number of 0-picks can be modeled by (1)-(4) under A1-A13. If parameters f_{hi} in (1) are unbiased, solving (1)-(4) will yield the optimal solution under these assumptions. Since formula (6) provides an unbiased estimate of the parameters f_{hi} under these assumptions, we can conclude the following proposition:

P1 : the Order-Quantity Based Rule provides the optimal replenishment orders under assumptions A1-A13

After all parameters f_{hi}^{OQBR} are obtained by (6), and problem (1)-(4) is solved, the replenishment crew can be provided with the replenishment orders and their priorities (provided by the optimal values of the variables z_{hi}). As in the SNR, as soon as all emergency products are replenished, forklift drivers can keep on performing replenishment operations following the traditional replenishment policy (i.e. the (s, S)-policy).

5.2.3 Order-Based Rule (OBR)

Though the OQBR will find the optimal assignment of priorities to replenishment orders in case that assumptions A1 till A13 hold, it has one major disadvantage. Namely, all possible orderings in which the N_h orders containing product h are picked have to be distinguished in (6). The number of orderings increases exponentially as N_h increases, and the computation time also does so. This implicates that the OQBR will face the problem of a huge computation time as soon as there are some products that are ordered a lot. That is why we also present a variant of the replenishment policy presented in section 5.2.2, which uses a slightly different formula for calculating f_{hi}. This version does not have the problem of increasing computation times, but simplifies the problem by assuming the following:

- A14: All order lines containing product h, require the same number of products:
$$\bar{Q}_h = \frac{\sum_{k=1}^{N_h} Q_h(k)}{N_h}.$$

Assuming A1-A14, the expression for f_{hi} can be calculated with the formula derived below. This time the superscript *OBR* (Order-Based Rule: this policy particularly looks at the number of orders for each emergency product) is used in order to distinguish the replenishment policy that uses this formula from the one that is based on assumptions A1-A13.

$$\begin{aligned}
f_{hi}^{OBR} &= \mathbb{E}^{OBR}[X_h(t_i)] = \sum_{j=1}^{N_h} \mathbb{E}\left[X_h(t_i)|Y_h(t_i)=j\right] \cdot \mathbb{P}(Y_h(t_i)=j) \\
&\overset{A6-A8}{=} \sum_{j=1}^{N_h} \mathbb{E}\left[X_h(t_i)|Y_h(t_i)=j\right] \cdot \binom{N_h}{j} \cdot \left(\frac{t_i}{T}\right)^j \cdot \left(\frac{T-t_i}{T}\right)^{N_h-j} \\
&\overset{A11-A14}{=} \sum_{j=1}^{N_h} \mathbb{1}_{S_h(1)-j\,\bar{Q}_h<0} \cdot \left\lceil \frac{j\,\bar{Q}_h - S_h(1)}{\bar{Q}_h} \right\rceil \cdot \frac{1}{N_h!} \cdot \binom{N_h}{j} \cdot \left(\frac{t_i}{T}\right)^j \cdot \left(\frac{T-t_i}{T}\right)^{N_h-j}.
\end{aligned} \tag{7}$$

Like we did in the derivation of f_{hi}^{OQBR}, the expected number of 0-picks before replenishment is conditioned on the number of picks before replenishment and multiplied by the probability that this situation occurs. This probability is calculated the same way as in (5). Because of assumption A14, all orders containing product h can be seen as equal. This means that the number of 0-picks before replenishment ($\mathbb{E}\left[X_h(t_i)\right]$), *given* an number of picks before replenishment ($Y_h(t_i)$), can be calculated by counting the number of times that there is no $\bar{Q}_h$

items on stock $\left(\mathbb{1}_{S_h(1)-j\,\bar{Q}_h<0} \cdot \left\lceil \frac{j\,\bar{Q}_h - S_h(1)}{\bar{Q}_h} \right\rceil\right)$. This is the main difference with f_{hi}^{OQBR}, which had to define all possible orderings in which the orders containing product h could be picked in order to 'count' the number of 0-picks. Again, A2, A3, A6 and A10 provide all parameters required, and assumptions A11-A13 are taken into account by only counting a 0-pick if it occurs before replenishment and if the stock-level is insufficient to fulfill the needs of an order line.

Under A1-A14 the problem of minimizing the number of 0-picks can still be modeled by (1)-(4), which will yield the optimal solution if parameters f_{hi} in (1) are unbiased. Since formula (7) provides an unbiased estimate of the parameters f_{hi} under these assumptions, we can conclude the following proposition:

- **P2**: the Order Based Rule provides the optimal replenishment orders under assumptions A1-A14

Like in the OQBR, the replenishment crew can be provided with the replenishment orders and their priorities (provided by the variables z_{hi}) after solving (1)-(4), with parameters f_{hi} calculated based on (7). The forklift drivers can keep on performing replenishment operations following the traditional (s, S)-policy as soon as all emergency products are replenished.

6 Comparison of Stock Replenishment Policies

In the previous sections we presented three stock replenishment policies, which differ in complexity. Without a doubt, the OBR is more complex than the SNR, and the OQBR is more complex than the OBR. As complexity brings about costs in the form of implementation, maintenance, and updates of the software, it is relevant to know whether those costs are worth to be made. In this section we try to give an answer to this question by comparing the replenishment policies by simulating replenishments during a wave.

6.1 Methodology and design of experiments

The idea of the simulation experiment presented next is to apply the replenishment policies presented in this paper to imaginary waves. However, since the difference between two replenishment policies in terms of 0-picks differs per wave, and since the characteristics of (the orders generated in) a wave differ a lot among warehouses, it makes no sense to draw quantitative conclusions (e.g., policy A always performs x percent better than policy B) from the results of these simulation experiments. Instead, the goal of the simulation experiment presented next is to uncover qualitative differences between the policies. The choices for the characteristics of the simulated waves presented next are therefore more or less random, and only meant to reveal these differences between the replenishment policies. Next to uncovering qualitative differences, the results of the simulation experiments may give an impression of what the quantitative differences could be under certain circumstances.

The simulation model uses the open source software that solves the LSAP (Cao (2008)). This software contains the so-called Munkres algorithm (Munkres (1957)). In order to check the validity of our program, we compared simulation results with the theoretical performance of the replenishment policies.

The imaginary warehouse meets all features described in the assumptions A1-A13. In order to test the quality of the policies, imaginary waves are generated based on the following characteristics:

- $T = 3, |H| = 20, |I| = 20$;
- $N_h =$ uniformly distributed on integers $\{1, 2, , 10\},\ \forall h \in H$;
- $Q_h(k) =$ uniformly distributed on integers $\{1, 2, , 10\},\ \forall h \in H, \forall k = 1, \ldots, N_h$;
- $S_h(1) =$ uniformly distributed on integers $\{0, 1, , D_h - 1\},\ \forall h \in H$, where $D_h = \sum_{k=1}^{N_h} Q_h(k)$;
- $t_i = i \cdot T_R$, where $T_R = T/|I|,\ \forall i \in I$;
- s_h : reorder point product h in (s, S) policy $= \mathbb{E}[D_h] + 1.5\sqrt{\mathbb{V}(D_h)},\ \forall h \in H$.

The reason why the uniform distribution is chosen in order to generate N_h, is to reveal the qualitative difference between the OQBR (/OBR) and the SNR. The sub-optimality of the SNR is mainly caused by the fact that this policy rejects part of the short-term demand information. As it only looks at the aggregate demand of a product, the SNR tends to assign lower than optimal priority to products for which N_h is great (high number of 0-picks expected) and vice versa for products for which N_h is small. In order to reveal this shortcoming, it is required that N_h shows some variation. This is done by generating N_h out of a uniform distribution.

In order to show the difference between the OQBR and the OBR, some variation in $Q_h(k)$ is required. Namely, by making assumption A14 the OBR implicitly assumes zero variation in $Q_h(k)$. In order show the effect of this shortcoming, $Q_h(k)$ is generated out of a uniform distribution.

In the simulation experiment, also the performance of the Random + (s, S)-policy is tested. In reality this policy causes replenishment of products that do not need to be replenished on the short term and vice versa. In order to simulate these cases we base the product specific s_h on the long-term demand. Namely by taking s_h as the expected demand per wave plus 1.5 times the standard deviation of the demand per wave. This can be calculated by means of the law of total variation (and expectation), using the distributions of N_h and $Q_h(k)$ as given above. This gives a value of $30.25 + 1.5 \times 17.17 \approx 56$ for every s_h.

Last, also the initial inventory level ($S_h(1)$) of product h is randomly generated. Since D_h of emergency products exceeds $S_h(1)$, the inventory level has to be initiated somewhere between zero and $D_h - 1$, which is done by a uniform distribution.

Based on the characteristics listed above, the orders to be picked in a wave, the inventory levels and s_h are randomly generated. Given these orders, the three replenishment policies presented in this paper prioritize the replenishment orders. Next, for every replenishment policy, simulation is used to obtain the distribution of the number of 0-picks resulting from applying the replenishment orders generated by this policy to this wave. Every time the wave is simulated (5.000 times), new order pick times are randomly generated based on A7. Common random numbers are used for every replenishment policy. In order to add more generality, this simulation is done for 1.000 waves, generating new orders and inventory levels for every wave.

6.2 Results

The results of applying the replenishment policies 5.000 times to 1.000 randomly generated waves, creating new random pick times every simulated wave, are stated below in Table 1.

The OQBR is modeled based on assumptions A1-A13, which makes it quite logical that this stock replenishment policy performs best in a simulation model that acts based on these

Table 1: Results of applying the replenishment policies to 1.000 randomly generated waves

Replenishment Policy	Mean # 0 picks per wave	Mean % dif. with the OQBR	Std. of % dif. with the OQBR	95% Conf. Interval of mean % dif. with the OQBR
Random + (s, S)	53.59	491.5	86.73	[486.2, 496.9]
SNR	12.56	35.5	14.24	[34.6, 36.4]
OBR	9.61	3.7	2.83	[3.5, 3.9]
OQBR	9.29	-	-	-

assumptions. The OBR performs only a little worse, which must be due to assumption A14. Among the three replenishment policies we developed, the SNR performs worst, which is not strange because of the amount of short-term demand information it rejects. Still, differences among the three are relatively small, particularly when these differences are compared to the difference with the Random + (s, S) policy.

As we said, the results depend on the settings of the imaginary warehouse. Yet, there are some factors that may indicate to what extent the number of 0-picks will change when one replenishment policy is chosen instead of another.

As we already suggested in section 6.1, variance in N_h causes the difference in performance between the OQBR and the OBR, whereas variance in $Q_h(k)$ causes the difference in performance between the OQBR (/OBR) and the SNR. This induces the question whether there is a correlation between the difference in performance between two policies (0-picks) and these variances. Namely, if these relations exist, decision makers could measure the value of these variances in their warehouse. This way, they can get an idea of the relative benefits of implementing a certain replenishment policy instead of another.

In order to test this, 1.000 waves are simulated (each one 5.000 times) again based on the characteristics listed in section 6.1. However, in order to get a better insight in the relation between the variance in $Q_h(k)$ and the difference in performance between the OQBR and the OBR, we generated $Q_h(k)$ out of a uniform distribution on the integers $\{1, \ldots, 10\}$ for 200 of the 1000 waves, on the integers $\{2, \ldots, 9\}$ for 200 waves, on the integers $\{3, \ldots, 8\}$ for 200 waves, and so on. Similarly, in order to get a better insight in the relation between the variance in N_h and the difference between the OQBR and the SNR, we generated N_h out of a uniform distribution on the integers $\{1, \ldots, 10\}$ for 200 of the 1000 waves, on the integers $\{2, \ldots, 9\}$ for 200 waves, on the integers $\{3, \ldots, 8\}$ for 200 waves, and so on.

After simulation, for all of these 1.000 waves the difference in performance between the OQBR and the OBR is calculated. Let n_{OBR} and n_{OQBR} be the number of 0-picks resulting from applying the OBR and the OQBR to a certain wave respectively. Then, for a given wave the difference in performance between the OBR and the OQBR is calculated by taking the average value of $100 \times ((n_{OBR} - n_{OQBR})/n_{OQBR} - 1)$ over the 5.000 simulations. For all 1000 waves this mean difference in performance is compared with the variance in $Q_h(k)$. This way the correlation between these statistics can be calculated. The same is done for the difference between the OQBR and the SNR compared to the variance in N_h.

As shown in Figure 1, the results indeed give a positive correlation of 0.36 between the percentage difference between the number of 0-picks of OQBR and OBR and the variance in $Q_h(k)$. The correlation between the percentage difference between the OQBR and the SNR and the variance in N_h is even equal to 0.85, see Figure 2.

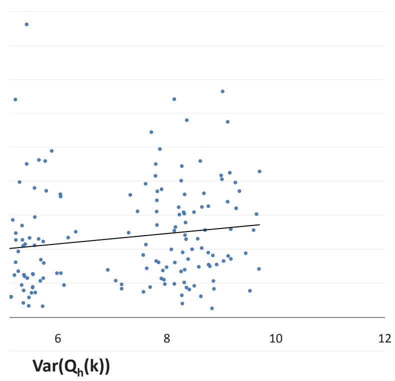

Figure 1: $Var(Q_h(k))$ vs. % difference between OQBR and OBR of 1000 waves

A good explanation of the first correlation can be found in the fact that the OBR assumes A14. If this assumption is met, there won't be any difference between the results of the OBR and the OQBR. However, when the distance between the reality and this assumption is growing bigger, which is the case when $Q_h(k)$ shows more variation, the bias in the calculations made by the OBR will grow bigger, causing that this difference increases.

The difference between the OQBR(/OBR) and the SNR can be explained by the fact that the

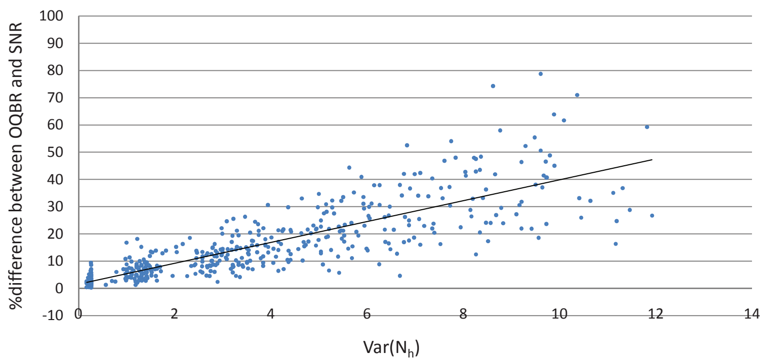

Figure 2: $Var(N_h)$ vs. % difference between OQBR and SNR of 1000 waves

first takes N_h into account, while the SNR doesn't. This will cause sub-optimality in the SNR, because N_h is a good indicator of the number of 0-picks that may occur for a certain product. As stated in section 6.1, this causes that the SNR tends to assign lower than optimal priority to products for which N_h is great and vice versa for products for which N_h is small. If N_h of all emergency products is almost the same, this sub-optimality will not be very big. This will often be the case when a wave lasts very short. In contrast, when there is much variation in this statistic, this will cause that the difference between these two replenishment policies increases.

In short, there are good reasons to assume that the difference between the OQBR and the OBR increases when the variance in $Q_h(k)$ increases, and that the difference between the SNR and the OQBR(/OBR) increases when the variance in N_h increases (so also often when T increases). This is also confirmed in experiments performed in De Vries (2010).

Last, also the mean computation times of the process of transforming demand and inventory information into replenishment orders are calculated for each replenishment policy, see Table 2. The computation-times and their variances are relatively small. The replenishment orders are generated only once per wave, so that the computation times stated above won't cause a waste of labor due to waiting for replenishment orders. Because the (s, S) policy is a continuous review policy in which replenishment orders are updated continuously, there is no (measurable) computation time for generating the replenishment orders for this policy.

Table 2: Mean and variance of computation time of generating the replenishment orders per replenishment policy

Replenishment Policy	Mean (seconds)	Variance
(s, S)	-	-
SNR	0.003	0.000
OBR	0.030	0.001
OQBR	1.707	5.458

7 Results of implementation in warehouse

At the time of the case study (see also Carrasco-Gallego & Ponce-Cueto (2009)), the SNR was the only available replenishment policy that takes short-term demand information into account. Therefore this policy was chosen to be implemented in the warehouse. After implementation in the warehouse's WMS, its effectiveness was monitored by measuring the relative number of order lines that caused a 0-pick. Figure 3 shows how the effect of implementing the SNR is. The SNR is progressively implemented in each of the three piece picking areas during the following months: April, May, June. Before policy implementation in March, there were almost 14 order lines for every 1000 which caused a 0-pick. When implementation was considered complete, 0-picks had been reduced to less than 3 for every 1000 lines. This means that using the SNR instead of the (s, S) policy decreased the number of 0-picks with about 80%. Those results obtained after the implementation of the SNR policy in the real life warehouse are coherent with the results obtained in the simulation study. In the latter, the switch from the Random+(s, S) policy to the SNR reduced the mean 0-picks per wave from 53.59 to 12.56, that is a 76.5% decrease.

8 Conclusions

In this paper new internal stock replenishment policies for warehouses using a forward-reserve storage strategy have been presented, in order to minimize the problem of 0-picks. The unique feature of these policies is that they assign priorities to replenishment orders based on short-term demand information that is available because of the wave-picking strategy used in the

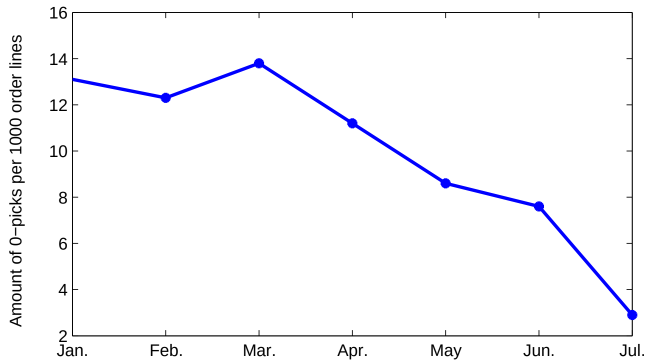

Figure 3: SNR implementation results in a warehouse

warehouse.

The first stock replenishment policy proposed in this paper, called the Stock-Needs Rule, is based on the idea of assigning priorities to replenishment orders based on a ratio dividing available inventory by wave-demand of the corresponding product.

The second replenishment policy makes use of *all* short-term demand information that is available in order to assign priorities to replenishment orders. This policy is called the Order-Quantity Based Rule. The problem of minimizing the total expected number of 0-picks is modeled as a Linear Sum Assignment Problem. In this problem products are assigned to replenishment orders based on the expected number of 0-picks that occurs for a product if it is assigned to a certain replenishment order (i.e., replenished at a pre-known time). This value can be calculated exactly because of some assumptions that are made (i.e., A6-A8). Because the computation time of calculating this value explodes when some products are ordered often, a simplified (biased) version of the formula that calculates this value is made. The replenishment policy that uses this formula instead of the original one is called the Order Based Rule.

Though the OQBR outperformed the other replenishment policies in a simulation experiment, it is not true that this one should always be favored at the expense of the other policies. First, the replenishment policies should also be compared in terms of costs of implementation, maintenance, and updates of replenishment policy in the warehousing software. Namely, these costs will be relatively high for the OQBR and the OBR.

Second, if the assumptions the OQBR and the OBR are based on are not met, the difference in performance between replenishment policies can decrease. For example, simulation experiments in De Vries (2010) have shown that the difference between the OQBR and the other policies gets a bit smaller when the replenishment times are stochastic instead of deterministic. However even if there is a substantial bias in their calculations, the OQBR and the OBR have the great advantage of taking all short-term demand information into account.

Last, the relative performance of a replenishment policy depends very much on the settings of the warehouse. Two variables are explained that are correlated with the difference between the replenishment policies in terms of 0-picks: the variance in N_h and the variance in $Q_h(k)$. Decision makers should measure the value of these variables in their warehouse, so that they can get an idea of the relative benefits of implementing a certain replenishment policy instead of another. This way they can make a good trade-off between benefits in terms of 0-picks and costs in terms of complexity.

Because the SNR was the only replenishment policy that was available at the time of the case study, this policy replaced the (s,S) policy in the warehouse. Implementation reduced the average number of 0-picks from 13.8 to 2.9 per 1000 order lines, which reveals the effectiveness of this stock replenishment policy.

References

Van den Berg, J. P., Sharp, G. P., Gademann, A. J., & Pochet, Y. (1998). Forward-reserve allocation in a warehouse with unit-load replenishments . *European Journal of Operational Research*, *111*, 98–113.

Burkard, R. E., & Çela, E. (1999). Linear assignment problems and extensions. In D.-Z. Du, & P. Pardalos (Eds.), *Handbook of Combinatorial Optimization* (pp. 75–149). Dordrecht: Kluwer Academic Publishers volume A.

Cao, Y. (2008). *Munkres Assignment Algorithm.* Matlab Center: http://www.mathworks.cn/matlabcentral/fileexchange/20328-munkres-assignment-algorithm.

Carrasco-Gallego, R., & Ponce-Cueto, E. (2009). Redesigning a Piece Picking Area Replenishment Process Supported by a WMS. In *International Conference on Computers & Industrial Engineering 2009. CIE 2009* (pp. 748 – 753).

Coyle, J., Bardi, E., & Langley, J. (1996). *The Management of Business Logistics*. (6th ed.). Minneapolis: West Publishing Company.

De Vries, H. (2010). *Optimization of the Piece Picking Area Replenishment Process*. Bachelor Thesis. Econometric Institute, Erasmus University Rotterdam.

Frazelle, E. H., Hackman, S. T., Passy, U., & Platzman, L. K. (1994). The forward-reserve problem. In T. A. Ciriani, & R. C. Leachman (Eds.), *Optimization in industry 2: Mathematical Programming and Modeling Techniques in Practice* (pp. 43–61). New York: John Wiley & Sons, Inc.

Gagliardi, J., Ruiz, A., & Renaud, J. (2008). Space allocation and stock replenishment synchronization in a distribution center. *International Journal of Production Economics*, *115*, 19–27.

Gu, J., Goetschalckx, M., & McGinnis, L. (2007). Research on warehouse operation: A comprehensive review. *European Journal of Operational Research*, *177*, 1–21.

Hackman, S., Rosenblatt, M., & Olin, J. M. (1990). Allocating items to an automated storage and retrieval system. *IIE Transactions*, *22*, 7–14.

de Koster, R., Le-Duc, T., & Roodbergen, K. (2007). Design and control of warehouse order picking: A literature review. *European Journal of Operational Research*, *182*, 481–501.

Kuhn, H. W. (1955). The Hungarian Method for the Assignment Problem. *Naval Research Logistics Quarterly*, *2*, 83–97.

Martello, S., & Toth, P. (1987). Linear assignment problems. In S. Martello, G. Laporte, M. Minoux, & C. Ribeiro (Eds.), *Surveys in Combinatorial Optimization, Annals of Discrete Mathematics* (pp. 259–282). Amsterdam, North-Holland: Elsevier Science Publisheers B.V. volume 31.

Moin, N., & Salhi, S. (2007). Inventory routing problems: a logistical overview. *Journal of the Operational Research Society*, *58*, 1185–1194.

Munkres, J. (1957). Algorithms for the Assignment and Transportation Problems. *Journal of the Society for Industrial and Applied Mathematics*, *5*, 32–38.

Vonolfen, S., Affenzeller, M., Beham, A., Lengauer, E., & Wagner, S. (2011). Simulation-based evolution of resupply and routing policies in rich vendor-managed inventory scenarios. *Central European Journal of Operations Research*, *DOI: 10.1007/s10100-011-0232-5*.