Using repair priorities to reduce stock investment in spare part networks

A. Sleptchenko a,b,*, M.C. van der Heijden b, A. van Harten b

a EURANDOM and Department of Technology and Management, Eindhoven University of Technology, P.O. Box 513, 5600 MB Eindhoven, The Netherlands
b Faculty of Business, Public Administration and Technology, Department of Operational Methods for Production and Logistics, University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands

Received 26 November 2003
Available online 12 March 2004

Abstract

In this paper, we examine the impact of repair priorities in spare part networks. Several heuristics for assigning priorities to items as well as optimising stock levels are developed, extending the well-known VARI-METRIC method. We model repair shops by multi-class, multi-server priority queues. A proper priority setting may lead to a significant reduction in the inventory investment required to attain a target system availability (usually 10–20%). The saving opportunities are particularly high if the utilisation of the repair shops is high and if the item types sharing the same repair shop have clearly different characteristics (price, repair time). For example, we find an investment reduction of 73% for a system with single server repair shops with an utilisation of 0.90 that handle five different item types.

© 2004 Elsevier B.V. All rights reserved.

Keywords: Inventory; Queuing; Spare parts

1. Introduction

For advanced technical systems, such as engines, medical equipment, and airplanes, a sufficiently high system availability is enhanced by a distribution network for repairable spare parts. In this network, a failed item can follow two different routes through the system: either it enters the local repair facility, or it is forwarded to the next echelon upstream to be repaired there (e.g. the downstream echelon consists of frigates, the upstream echelon consists of a depot in the harbour, see Fig. 1). Usually items are sent to the higher echelon if local repair is technically impossible, i.e. if the local repair shop does not have appropriate equipment or skills. It is usually (and here) assumed that the decision whether to repair locally or not is based on such technical considerations only, and not by e.g. current repair shop workload. This is modelled by a fixed probability that the item can be repaired locally, independent of the system state. We will use the same assumption.
It is also assumed that items have multi-indenture structure, which means that every item (assembly) may consist of other items (subassemblies), see Fig. 1. If an assembly fails, we assume that either the failure is caused by the failure of exactly one of its subassemblies (so the replacement of this subassembly is sufficient to repair the assembly), or there is no specific subassembly causing the assembly failure (hence the assembly as a whole has to be repaired). In the case of a subassembly failure the assembly is disassembled and the failed subassembly is sent further for subassembly repair.

Next to each repair facility there is a stock with new items and when a failed item arrives at the repair shop an order for a new item is issued. If a new item is available on stock, it is dispatched to replace the failed item. At the same time, the failed item either enters the local the repair shop or dispatched to external repair/stock facility (subassembly repair of higher echelon repair) and after repair it is added to stock (Fig. 2).

A well-known problem is to choose optimal spare part stock levels in the network, such that target system availability is attained at minimal inventory investment. Inventory levels have to be selected for each spare part, being an item in a multi-indenture product structure, and for each location in a multi-echelon distribution network structure.

The class of VARI-METRIC models (cf. Slay, 1984; Sherbrooke, 1992) has been developed to tackle this problem. Their method aims to determine initial stock levels, assuming that all failed items are either repaired or replaced by new items if repair is impossible, so that the original amount of items remains circulating through the network. A deficit of the original VARI-METRIC method is the assumption that the repair shop capacities are infinite. As has been shown by Sleptchenko et al. (2002a) this may lead to significant errors in estimating system availability and determining spare part stock levels if the repair shop utilisation is high. Therefore, they model repair shops as multi-class, multi-server queuing systems to resolve this issue. Thus, they develop a variant of VARI-METRIC with finite repair capacities. In such models, we can introduce other degrees of freedom for efficiency improvement, such as capacities of the repair facilities and repair priorities. Sleptchenko et al. (2003) use a finite repair capacity model to make a trade-off between...
capacities of repair facilities and inventory stock levels. In their model all repair jobs have the same priority. Here we discuss opportunities for efficiency improvement by setting appropriate priorities to different repair jobs under finite capacity. We explain this below.

One of the key parameters determining stock levels of repairable items in spare part networks is the repair shop throughput time. It is clear that less inventory is needed to obtain a certain system availability if the throughput times can be reduced. If a repair shop handles a variety of item types and its capacity is limited, we can reduce the throughput times of expensive items by giving the corresponding repair jobs high priority. As a consequence, other item types having low priority will have to face longer throughput times. This means that priority setting leads to reduced stock levels for high priority items and increased stock levels for low priority items. This provides us an opportunity for inventory investment reduction while maintaining the same system availability: if we choose expensive slow movers as high priority items and cheap fast movers as low priority items, the reduction of inventory investment in high priority items will probably be more than the increase of investment in low priority items. Equivalently, we can increase the system availability with the same budget by introducing adequate priority rules.

In this paper, we examine the impact of using repair priorities in spare part distribution networks. We focus on static priorities, i.e. an a priori assignment of item types to priority classes for each repair shop, which is state-independent. We need such priorities, because we have to choose stock levels a priori as well. In fact, we focus on logistic decisions at a tactical level. At the operational level, one may consider using dynamic priorities additionally. That is, we may sequence repair jobs based on the specific system state at a specific point in time. For example, we may consider giving high priority to an item for which no stock is available, even if it has been classified as low priority originally. Such an operational finetuning of priorities can further improve the system performance, but it is out of the scope of our research. We will address the following research questions in the remainder of this paper:

(1) To what extent can we reduce inventory investment using a priori priority setting of items?
(2) Which rule should we use to classify items as high priority or low priority?
(3) Which impact does priority setting have on the stock levels of individual items?

To answer these questions, we need (1) an algorithm for assigning priorities to items sharing the same repair capacity and (2) an algorithm for spare part inventory optimisation based on these priorities. We will discuss both sequential optimisation and integral priority assignment/stock optimisation. In our approach, we model the repair shops as multi-class, multi-server queues with preemptive priorities. Our model contains two priority groups (high and low), each consisting of one or more subclasses. Hence priority assignment boils down to assigning either high or low priority to each item. To analyse these queuing models, we use the method as developed by Sleptchenko et al. (2002b). They solve the steady state equations, assuming that the probabilities of states with a high number of items in the system can be neglected. The performance measures required for the extended VARI-METRIC model can be obtained from these state probabilities. Because we have to solve high dimensional matrix equations, we have to restrict the number of item types (classes) and the number of servers per repair shop. Note that our priority assignment heuristics do not lead to optimal solutions. Even the case without priorities for capacitated repair shops is already that complex that optimal stock levels cannot be guaranteed except for some special cases.

Before we discuss our algorithms, we first position our research within the related spare part management literature (Section 2). Next, we explore the potential of repair priorities using a variant of VARI-METRIC for given priorities in Section 3. In Section 4, we discuss an algorithm for integral priority assignment and stock optimisation and a sequential algorithm. We compare the various heuristics in a numerical experiment in
Section 5. As performance criteria, we use run time and solution quality, that is, the inventory investment needed to achieve a target availability. Finally, we give our conclusions in Section 6.

2. Literature

It has been noted in several publications that efficiency gain is possible using repair priorities, see e.g. Pyke (1990) and Verrijdt et al. (1998). Still, repair priority setting in spare part distribution networks has received only little attention in the literature. A possible cause is the lack of appropriate queuing models to analyse such repair shops until recently. The analysis of spare part networks with finite capacity repair shops using simple first come first serve (FCFS) rule is even not straightforward, as is shown in Diaz and Fu (1997), Avsar and Zijm (2000) and Sleptchenko et al. (2002a).

Some insight has been gained using two approaches. In the first approach, repair priorities are modelled as a dual mode repair shop providing normal and emergency service, see Verrijdt et al. (1998) and their references. In their model, a spare part might receive a fast emergency repair if net inventory of these parts is lower than or equal to some level, but at higher costs. The model is restricted to a single-indenture, single-echelon case with ample repair capacity. A related model is discussed in Perlman et al. (2001), who assume that items arriving at repair shop \( i \) are sent to emergency repair with probability \( p_i \) and to normal repair with probability \( 1 - p_i \). They consider a single-item, two-echelon model with finite capacity repair shops. They optimise the system with respect to \( p_i \). Because it is a single-item model, the impact of priority assignment to various items sharing the same repair capacity is not examined. Both Verrijdt et al. (1998) and Perlman et al. (2001) assume independent processing of normal and emergency items, which is not valid if both processes share the same repair capacity.

A second approach is to use simulation modelling. Hausman and Scudder (1982) analyse the impact of static and dynamic priority rules for infinite capacity repair shops using simulation. The case with finite capacity repair shops is described in Scudder and Hausman (1982) and extended to the multiple-failure case in Scudder (1984). Pyke (1990) uses simulation to examine the impact of finite capacities and dynamic repair priorities along with dispatch priorities. He studies a single repair facility in a three-echelon network with single-indenture items only. For repair priorities, he considers for each spare part the number of systems that are down because of the failure of that specific item. The item that is contributing to most systems is repaired first, so this is a dynamic (state-dependent) priority rule. Pyke (1990) shows that his priority rule leads to a drastic decrease in the number of non-available systems for given stock levels if repair shops are heavily loaded (utilisation = 0.96). The impact of the priority rule is only small if the utilisation is 0.6. Also, he finds that the average number of failed systems depends strongly on the repair shop utilisation under FCFS repair, whereas the impact of the utilisation is much smaller if proper priority rules are used. Note that Pyke (1990) does not consider the impact on initial inventory investment. Given the amount of decision variables and the use of simulation rather than an analytic model, this is not possible. Also, his model is restricted to single-indenture items and a single repair shop in a two-echelon model, whereas we address general multi-echelon, multi-indenture systems with an arbitrary number of repair shops.

Next to these approaches to gain insight in the impact of repair priorities, other related research deals with the choice of dynamic priority rules in repair shops that are characterised by uncertain processing times, see e.g. Guide et al. (2000). Their main question is which priority scheduling policies minimise the average repair shop throughput time if no spares are allowed. Based on a simulation model, they find that the shortest processing time (SPT) rule is usually dominant. However, they do not take into account differences in item costs. If item costs are very different, it is intuitively clear that this characteristic should be taken into account when setting priorities.

To our best knowledge, there is no analytical model for multi-echelon, multi-indenture inventory control that addresses both finite capacity and
repair priorities. Therefore, our contribution consists of

1. the extension of VARI-METRIC to finite capacity repair shops with fixed priority classes, i.e. to optimise stock levels analytically, given item priorities (Section 3);
2. heuristics for simultaneous priority assignment to items and stock optimisation, thereby providing the opportunity to reduce initial inventory investment in spare parts (Section 4);
3. insight in the impact of repair priorities on the stock investment and stock allocation, obtained from numerical experiments using our new algorithms (Section 5).

For background literature on VARI-METRIC and its extensions, we refer to the reviews by Guide and Srivastava (1997), Rustenburg et al. (2001) and Kennedy et al. (2002).

3. The impact of repair priorities

In this section, we will first discuss how VARI-METRIC can be extended to finite capacity repair shops and with job priorities. Using this model, we can then optimise spare part stock levels for a given assignment of items to (fixed) priority classes. Next, we will compare the inventory investment required for various priority classifications to the investment requirements if no priorities are used. In all cases, we consider finite capacity repair shops. We use the following assumptions, taken from VARI-METRIC (Sherbrooke, 1992):

1. System failures occur according to a stationary Poisson process.
2. Each assembly failure is caused by at most one subassembly failure; we represent this failure cause by a discrete probability distribution.
3. Requests for items are handled first come first serve (FCFS) at each stockpoint.
4. Each stockpoint has exactly one supplier.
5. Each stockpoint uses an one-for-one inventory replenishment rule for each item.
6. The assignment of jobs to repair shops is only based on technical issues such as the failure cause, the skills of the repair men and the availability of tools; this is represented by a discrete probability distribution.
7. The item repair times are independent, identically distributed random variables.
8. All item failures cause system failures, so all items considered are critical.

Additionally, we use the following assumption for the finite capacity repair shops:

9. The item repair times are exponentially distributed and can have different means for different item types.
10. In case of priorities, the service of low priority jobs may be interrupted if a high priority job arrives and all servers are busy (i.e., there is preemption).

3.1. VARI-METRIC with fixed priority classes

VARI-METRIC is a greedy heuristic to optimise stock levels in spare part networks. In each iteration, the inventory level of a certain item at a certain location is increased. The selection criterion for item and location is the maximum backorder reduction per invested dollar, because it can be shown that maximising the average availability of the installed base is approximately equivalent to minimising the sum of the expected backorders of all highest indenture items at all downstream stock locations. Calculation of this criterion function is relatively easy if all repair shops are assumed to have infinite capacity. The reason is that we can apply Palm’s theorem that states that the number of items in an $M/G/\infty$ queue is Poisson distributed.

If a repair shop capacity is finite, we can model the repair shops by $M/M/k$ queues or $M/G/c$ queues. However, these models are not sufficient if multiple items having different repair times share the same repair shop, as is common in practice. In that case, we have to distinguish various job classes and we end up with a multi-class, multi-server queuing model. Sleptchenko et al. (2002a) show that such a model can be included in the VARI-METRIC framework using the multi-class $M/M/k$ queue rather than the default $M/G/\infty$ to calculate the first two moments of the number of items per
item class at each repair shop queue. Instead of simply using Palm’s theorem for the $M/G/\infty$ queue, they use an algorithm to analyse the multi-class $M/M/k$ queue as developed by Van Harten and Sleptchenko (2003). This algorithm aims to find the system state probability distribution, so that all relevant performance measures (e.g. all moments of the number of items in the system) can be calculated.

When embedding this queuing model in the VARI-METRIC framework, Sleptchenko et al. (2002a) approximate the distribution of the number of items in the system by a simple probability distribution fitted by the first two moments, see Adan et al. (1996). The reason for this approximation is to facilitate the calculation of the sum of the mean number of backorders of all highest indenture items at all downstream stock locations. These mean backorders depend on the probability distributions of the number of all items in all repair shops, so that using the exact probability distributions would lead to excessive calculations. Although the VARI-METRIC approach can still be applied, the analysis is complicated by the fact that the backorders of various items are correlated if the repair shop capacities are finite. Ignoring these correlations leads to an error while estimating the average system availability (cf. Sleptchenko, 2002).

Similarly, other queuing models for the repair shops can be embedded in VARI-METRIC if the first two moment of the number of items in the queuing system can be evaluated. The latter condition is not satisfied for many queuing models that have been discussed in the literature. Often, queuing analysis is limited to a few core performance characteristics like the mean waiting time and the mean queue length. Even if the variance of the queue length can be evaluated, this is not sufficient for use in VARI-METRIC, because we need the variance of the total number of items in the system. Note that the number of items in queue and the number of items in repair are correlated, so that we cannot simply add up both variances.

A suitable priority queuing model for our purpose is the multi-class $M/M/k$ priority model that has been analysed by Sleptchenko et al. (2002b), being an extension of the non-priority model as has been analysed by Van Harten and Sleptchenko (2003). Basic characteristics are:

- repair jobs are classified in item classes $i = 1 \ldots I$;
- each item class is assigned either high or low priority; hence we have two priority groups each consisting of one or more item classes;
- repair job arrivals are Poisson distributed with an item class dependent arrival rate $\lambda_i$;
- repair times are exponentially distributed with an item class dependent service rate $\mu_i$;
- all jobs are handled by $k$ identical servers;
- a low priority job is interrupted if a high priority job arrives finding all servers busy; if multiple low priority are in service, we select the job to be postponed randomly with equal probabilities.

In a spare part network, we may have dedicated repair capacity for certain job types. In that case, we model each dedicated repair facility as a multi-class $M/M/k$ priority model.

3.2. Comparing priority and non-priority repair shops

In this section, we will use two variants of VARI-METRIC to examine the impact of repair priorities, namely the variant based on a multi-class $M/M/k$ queue without priorities and the variant based on the multi-class $M/M/k$ priority queue. Using some numerical experiments, we will show that the inventory investment can be reduced significantly if we use repair priorities. In this section, we simply optimise stock levels for fixed priority classes and vary the priority assignment in our experiments to see the impact of proper priority setting.

We use the product structure and distribution structure as shown in Fig. 1. The network consists of one central depot and four frigates. On each frigate, two pumps are installed (A and B). A frigate is available if both pumps are in operation. The pumps have two critical parts each. For pump A, these are a valve and a piston and for pump B these are a flange and a piston. The failure rates of pump A and pump B are 10 and 15 failures per pump per year, respectively. The failure cause for
pump A is either the valve (with probability 0.3) or the piston (with probability 0.4) or there is a combination of factors, meaning that the whole pump has to be repaired or replaced (with probability 0.3). For pump B, the failure causes are 0.4 for both the flange and the piston and 0.2 for the pump itself. Repair shops are present at the depot and the frigates. Every repair job can be repaired at the frigate with probability 0.2 (simple repairs), whereas it should be forwarded to the depot with probability 0.8 (complex repairs). The order-and-ship time between depot and frigates is zero for all items. Table 1 shows the item prices and the mean repair times. The mean repair times are expressed relatively to the mean repair time of pump A. Because we will vary repair shop capacities and utilisation in our experiments, we will choose the repair time of pump A such, that a given utilisation is reached.

We vary the following parameters in our experiments:

1. The assignment of items to repair shops: (a) all items are assigned to the same repair shop, so we have a single five-types repair shop at all locations, (b) at each location one repair shop is dedicated to pumps A and B (two-types model) and one repair shop is dedicated to the valve, the flange and the piston (three-types model).

2. The assignment of items to priority classes; we consider all options, so we have $2^5 - 1 = 31$ options for a single five-types repair shop and $(2^2 - 1) + (2^3 - 1) = 21$ options for the combination of a two-types and a three-types repair shop. Note that we subtract one option per repair shop, because it does not make a difference whether all items have high priority or all items have low priority.

3. The number of servers per repair shop: $k = 1$ or 3.

4. The repair shop utilisation: $\rho = 0.8$ or 0.9.

With respect to the repair shop utilisation, note that these numbers are in line with cases as examined by other authors. Hausman and Scudder use an utilisation 0.78, whereas Pyke (1990) varies the utilisation between 0.6 and 0.96.

All repair shops in the system have the same number of servers and the same utilisation. We vary the number of servers and the utilisation in our experiments, because these parameters appeared to be critical in the finite capacity analysis by Sleptchenko et al. (2002a). Other parameters, such as failure rates, the number of echelons and indenture levels, appear to be less critical. In the same paper, it is also shown that the impact of finite capacity on estimation of the system availability is especially high if the utilisation is high. Therefore, we do not include low utilisation rates in our experiments.

In Table 2, we show the key results of our numerical experiments. The first line contains the inventory investment that is required to obtain an average availability of 95% if all items have the same priority. We show the investment corresponding to the best priority assignment in bold figures. Note that the investment reductions are only comparable within the same column and not between columns. The reason is that the item repair times vary between columns, because these times are chosen such, that the specified utilisation rate is attained for the given assignment of items to repair shops and the given number of servers $k$. Further, some of the assignments of items to priority classes are not relevant for the two-repair shop experiments, because it does not make a difference whether all items within a repair shop have high priority or all have low priority. That is, for example case LL|HLH is similar to HH|HLH because in the assembly repair shop both items have the same priority levels (both low or high) and in the subassembly repair shop priority

Table 1
The relative mean service times and item prices

<table>
<thead>
<tr>
<th>Item</th>
<th>Pump A</th>
<th>Pump B</th>
<th>Valve</th>
<th>Flange</th>
<th>Piston</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean service time $E[S]$ relative to the mean service time of pump A</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>Prices (€) $c_i$</td>
<td>5000</td>
<td>7100</td>
<td>200</td>
<td>300</td>
<td>500</td>
</tr>
</tbody>
</table>
assignment is same (HLH). Therefore, some of the cells in Table 2 are empty.

From Table 2, we see that proper priority setting may reduce the inventory investment requirement substantially (12%-73%). The investment reduction decreases with the number of servers and increases with the repair shop utilisation rate. However, the impact of the assignment of items to repair shops is dominant. The potential investment reduction is particularly high if items with clearly different characteristics share the same repair capacity. In that case, the increase in stock investment for (cheap) low priority items is much less than the decrease in stock investment for (expensive) high priority items. The latter is illustrated in Table 3 that shows the modification in stock levels per item at both the depot and each frigate if the optimal priority assignment is used compared to a non-priority system. We only present the results for 1 server and utilisation 0.9, but the results for the other cases are similar.

<table>
<thead>
<tr>
<th>Priority per item (L = low, H = high)</th>
<th>One five-class repair shop</th>
<th>One two-class and one three-class repair shop</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\rho = 0.8$</td>
<td>$\rho = 0.9$</td>
</tr>
<tr>
<td></td>
<td>$k = 1$</td>
<td>$k = 3$</td>
</tr>
<tr>
<td>No priorities</td>
<td>326</td>
<td>697</td>
</tr>
<tr>
<td>L L H L L L</td>
<td>346</td>
<td>732</td>
</tr>
<tr>
<td>L L H H H</td>
<td>404</td>
<td>882</td>
</tr>
<tr>
<td>L L L H H</td>
<td>429</td>
<td>943</td>
</tr>
<tr>
<td>L L L H H</td>
<td>420</td>
<td>917</td>
</tr>
<tr>
<td>L L H H H</td>
<td>449</td>
<td>987</td>
</tr>
<tr>
<td>L L L H H</td>
<td>574</td>
<td>1318</td>
</tr>
<tr>
<td>L L H H H</td>
<td>625</td>
<td>1490</td>
</tr>
<tr>
<td>H L L H H</td>
<td>235</td>
<td>473</td>
</tr>
<tr>
<td>H L H H H</td>
<td>244</td>
<td>494</td>
</tr>
<tr>
<td>H L H H H</td>
<td>310</td>
<td>633</td>
</tr>
<tr>
<td>H L H H H</td>
<td>329</td>
<td>680</td>
</tr>
<tr>
<td>H L L H H</td>
<td>308</td>
<td>641</td>
</tr>
<tr>
<td>H L H H H</td>
<td>333</td>
<td>684</td>
</tr>
<tr>
<td>H L L H H</td>
<td>449</td>
<td>1006</td>
</tr>
<tr>
<td>H L H H H</td>
<td>498</td>
<td>1151</td>
</tr>
<tr>
<td>L L H L L L</td>
<td>236</td>
<td>468</td>
</tr>
<tr>
<td>L L H L L L</td>
<td>252</td>
<td>495</td>
</tr>
<tr>
<td>L L H L L L</td>
<td>317</td>
<td>650</td>
</tr>
<tr>
<td>L L H L L L</td>
<td>351</td>
<td>714</td>
</tr>
<tr>
<td>L L L H L H</td>
<td>318</td>
<td>652</td>
</tr>
<tr>
<td>L L H L L H</td>
<td>352</td>
<td>726</td>
</tr>
<tr>
<td>L L H H H H</td>
<td>495</td>
<td>1148</td>
</tr>
<tr>
<td>L L H H H H</td>
<td>559</td>
<td>1417</td>
</tr>
<tr>
<td>H H L L L L</td>
<td>125</td>
<td>148</td>
</tr>
<tr>
<td>H H H L L L</td>
<td>132</td>
<td>201</td>
</tr>
<tr>
<td>H H L H L L</td>
<td>189</td>
<td>290</td>
</tr>
<tr>
<td>H H H H L H</td>
<td>204</td>
<td>333</td>
</tr>
<tr>
<td>H H L L H H</td>
<td>171</td>
<td>251</td>
</tr>
<tr>
<td>H H H L H H</td>
<td>183</td>
<td>286</td>
</tr>
<tr>
<td>H H L H H H</td>
<td>293</td>
<td>543</td>
</tr>
<tr>
<td>Savings (%)</td>
<td>62%</td>
<td>55%</td>
</tr>
</tbody>
</table>
We see that the stock levels of high priority items decrease and those of low priority items increase, as expected. The main cause for the huge investment reduction if all items share the same repair capacity is the difference in item costs. For example, the number of pumps B on stock decreases from 4 (frigates) $\times 13+1$ (at depot) = 53 pumps to $(3 \times 2+1 \times 3)+0 = 9$ pumps. The investment reduction is $44 \times 7100 \ € = 312.400 \ €$. Similarly, we see that we save $230.000 \ €$ on the other high priority item, pump A. Although the stock levels of the low priority items increase significantly, these are all cheap, so that we need only $45.500 \ €$ additional investment in valves, flanges and pistons. As a consequence, the investment reduction is $496.600 \ €$.

Note that the reduction in pump investment is also high compared to the increase in investment in cheap items, because the repair times of pumps are shorter in these examples (see Table 1). Therefore, the delay of low priority items is not too much. Intuitively, it is clear that the item costs and the mean repair times are important parameters influencing investment reduction because of priority setting. Giving high priority to items with a relatively small repair time has a similar impact as using the shortest processing time (SPT) rule: the average number of items in the system decreases.

### 4. Integral priority assignment and stock optimisation

Finding an optimal assignment of items to priority classes and determining the corresponding optimal stock levels is a very complex problem because of the large number of integer decision variables and the complex relation between stock levels and backorders. As discussed in the previous sections, the latter relation includes complex queuing models for each finite capacity repair shop. Therefore, we will discuss two heuristic optimisation approaches in this section:

- nested optimisation, i.e. we optimise the stock levels for each priority assignment considered;
- sequential priority assignment and stock optimisation (decomposition).

<table>
<thead>
<tr>
<th>Repair shop structure</th>
<th>Location+→Item</th>
<th>Depot</th>
<th>Pump A</th>
<th>Pump B</th>
<th>Valve</th>
<th>Flange</th>
<th>Piston</th>
</tr>
</thead>
<tbody>
<tr>
<td>One 5-class repair shop</td>
<td>No priorities</td>
<td>1 10</td>
<td>13 21</td>
<td>13 21</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>One 2-class and one 3-class repair shop</td>
<td>Optimal priorities</td>
<td>0 2 3</td>
<td>2 2 4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note: The mark /C3 means that one frigate gets one additional item on stock in the optimal solution to obtain an average system availability of at least 0.95.

### Table 3: Comparison of stock allocation between the optimal priority system and a non-priority system (c = 1, $\rho = 0.9$)

<table>
<thead>
<tr>
<th>Frigates</th>
<th>Priority distribution</th>
<th>Investment levels</th>
<th>LLL HHH HHH</th>
</tr>
</thead>
<tbody>
<tr>
<td>No priorities</td>
<td>0 0 8 15 21 2 2* 9 15 22 190000 HLLL HHH</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Optimal priorities</td>
<td>1 4 2 24 3 4 25* 3 33 5 892000 HHHH HGH</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The mark * means that one frigate gets one additional item on stock in the optimal solution to obtain an average system availability of at least 0.95.
4.1. Nested optimisation

A straightforward method to optimise priority assignment and stock levels is to extend VARI-METRIC with priority assignment. VARI-METRIC subsequently adds stock for those items at those locations that give the highest improvement in the goal function (mean backorders) per invested dollar. It is difficult to mix the stock level decision with priority assignment, because changing priorities requires a new stock optimisation. However, we can construct a layer of priority assignment around the VARI-METRIC stock optimisation procedure. This leads to the following nested optimisation heuristic.

4.1.1. Nested stock optimisation and priority assignment

1. **Initialisation**: Set all item priorities at all repair shops equal, at the low level; optimise stock levels for equal priorities according to Sleptchenko et al. (2002a).
2. **Priority improvement**: Examine the impact of assigning high priority to a single item that is currently low priority for all repair shops while keeping the same priority for all other items:
   2.1. Re-optimise the stock levels for each modified priority assignment and calculate the investment reduction resulting from the modified priority assignment using the extended VARI-METRIC algorithm as discussed in Section 3.1.
   2.2. Select the priority modification causing the highest inventory reduction; if this reduction is positive, implement the new priority assignment and repeat step 2; otherwise, keep the previous solution as the optimal one and stop.

This is a nested optimisation, because all stock levels are re-optimised every time the priority assignment is modified. Note that this is a heuristic procedure, because (1) VARI-METRIC is generally not an exact optimisation procedure (2) items are only moved from the low priority group to the high priority group and not vice versa; there is no guarantee that this will lead to an optimal priority assignment. Still we examine many options for priority assignment and stock levels, so that we will probably find a good (although not optimal) solution. Because an optimal solution is not possible for most practical problems (even for a non-priority system), it is impossible to give an indication for the quality of the optimisation procedure. However, on the simple case from Section 3.2, the procedure presented here gave exact optimum. Further in Section 5, we compare priority optimisation to an optimised system without priorities and analyse the investment reduction. Unfortunately, this procedure may be time consuming, because the number of stock optimisations may be excessive. Therefore, we search for a faster heuristic.

4.2. Sequential priority assignment and stock optimisation

The nested optimisation procedure considers many modifications in priority assignment. It is clear that a part of these assignments is probably not advantageous, because the item characteristics do not justify high priority. For example, items having relatively low costs and/or relatively long service times are candidates for low priority rather than high priority, as we also concluded from the numerical experiments in Section 3.2. Therefore, we can try to reduce the number of priority modifications under consideration.

A simple way to reduce the run time needed for the optimisation is to decompose the problem into priority assignment and stock optimisation. We first assign items to priority classes in each repair shop without considering stock optimisation. Next, we optimise stock levels only once for the (heuristically found) priority assignment. An a priori assignment of items to priority classes is not straightforward, because it depends on many parameters of all items sharing the same repair shop capacity, such as the total repair shop utilisation, the variation in items costs, the variation in mean service times and the variation in arrival rates. Some preliminary experiments that we do not discuss here showed that the high priority class may use 50–60% of the repair shop capacity in some cases and only 10–20% in other cases. The precise relation between the optimal assignment and all
relevant parameters is not clear. Only the number of servers seems to be less relevant for the optimal priority assignment, because given fixed utilisation rate a queueing system with a big number of servers behave similar to a system with a smaller number of servers and shorter processing time.

Next, we estimate the impact of priority assignment on investment using a goal function that (1) is a good indicator for stock investment and that (2) can easily be evaluated. It is clear that stock levels tend to be high (low) if the number of items in the repair shop are high (low). Therefore, we propose to use as goal function for priority assignment the total expected costs of all jobs in a $M/M/1$ multi-class priority queue, i.e. $\sum_{i=1}^{k} c_{i}E[N_{i}]$, where $N_{i}$ denotes the number of items of type $i$ in the repair shop and $c_{i}$ the costs of item $i$. The hypothesis is that the stock levels are approximately proportional to the number of items in the systems. Obviously, this is not true, but we hope that this indicative goal function is sufficient for priority assignment. We will test the validity of this heuristic approach in some numerical experiments (Section 5).

An advantage of this approximate goal function is that we can evaluate it quickly, because the state dimension of the $M/1$ multi-class priority queue is $k$ times as fast if the actual repair shop has $k \geq 2$ servers (only for priority assignment). Although this method might not seem to be a good one, it has shown good performance during experiments, results of which are presented in the next sections.

One option to find the optimal priority assignment for the approximate goal function is to enumerate all options and choose the priority assignment with minimum costs. This is possible for a moderate number of item classes within a repair shop, say 12 at most (giving $2^{12} - 1 = 4095$ options). However, we can also construct a heuristic from Buzacott and Shantikumar’s (1993) result on the optimal priority assignment in the multi-class $M/G/1$ model with non-preemptive priorities. They order jobs according to a priority index that is assigned to each class. They assume that each class contains one item and prove that $c_{i}/E[S_{i}]$ is the optimal priority index, where $S_{i}$ and $c_{i}$ denote the service time and the costs per time unit in the system for item $i$, respectively. That is, the item with the highest index $c_{i}/E[S_{i}]$ should have the highest priority.

Note that our model is different, because we have two priority groups with multiple items (classes) within each priority group. Hence we should not only order items, but also decide how many items should be allocated to the high priority group. Based on the result above, a reasonable heuristic is to order items according to the index $c_{i}/E[S_{i}]$. Starting with low priority items only, we move items to the high priority class until the cost criterion cannot be improved anymore. In the numerical section, we will check the quality of this priority assignment heuristic by comparison to a full enumeration of all priority assignments.

Summarised, the sequential heuristic consists of the following steps.

4.2.1. Sequential priority assignment and stock optimisation

1. Priority assignment per repair shop,
   1.1. Initialisation:
      1.1.1. If the number of servers in the repair shop is $k \geq 2$, then modify the mean service time for priority assignment as $E[S_{i}] := E[S_{i}] / k$.
      1.1.2. Sort the items in decreasing order of their cost/service time ratio $c_{i}/E[S_{i}]$.
      1.1.3. Assign low priority to all items in the repair shop.
      1.1.4. Calculate the expected number of items $E[N_{i}]$ in the $M/M/1$ multi-class priority queue for each item type $i$.
      1.1.5. Calculate the minimum total expected costs of being in the system for all job classes, $\sum_{i=1}^{k} c_{i}E[N_{i}]$.

1.2. Iteration:
      1.2.1. Assign high priority to the low priority item $i^*$ that is highest on the ordered list.
      1.2.2. Recalculate the expected number of items $E[N_{i}]$ in the $M/M/1$ multi-class priority queue for each item...
and the corresponding total expected costs \( \sum_{i=1}^{k} c_i E[N_i] \).

1.2.3. If the total costs are less than in the previous iteration, continue with the next iteration; otherwise, assign low priority to item \( i \) and stop.

2. Optimise the stock levels and calculate the investment reduction resulting from the priority assignment using the extended VARI-METRIC algorithm from Section 3.1.

5. Numerical analysis

In this section, we will test both optimisation heuristics. Evaluation criteria are (1) the investment reduction obtained compared to a non-priority finite capacity model and (2) the computer run time requirements. We conduct three sets of experiments. In the first set, we compare the investment reduction obtained using the two heuristics for priority assignment and stock optimisation as presented in the previous section. It will turn out that sequential priority assignment and stock optimisation (see Section 4.2) is as good as nested optimisation and much faster in most of our numerical experiments. Therefore, we conduct a second set of experiments based on sequential optimisation to examine priority assignment in more detail (i.e., experiments in which many items share the same repair capacity). In the third set we validate our findings by running experiments for a completely different data set as given in Hausman and Scudder (1982).

5.1. Comparing the algorithms for priority assignment and stock optimisation

The data for our numerical experiments are similar to Section 3.2, except for the following:

- We only consider the cases with \( k = 3 \) servers, because we see from Table 2 that the results for \( k = 1 \) server are similar.
- We consider both a two-indenture and a three-indenture system, see Fig. 3; the item costs and the mean service times for the three indenture models are shown in Table 4.
- We have two scenarios for the failure causes, see Fig. 3 (the probability that an assembly failure is caused by a specific subassembly is displayed as a number at the connecting link); the first scenario is similar to Section 3.2 with an extension for the three-indenture variant; the second scenario is displayed between parentheses.

In our first set of experiments, each repair facility is dedicated to the items from the same level in the indenture structure. That is, each location has two respectively three (multi-class, multi-server) repair shops in the two- respectively three-indenture model. One repair shop is dedicated to pump A and pump B, one repair shop is dedicated to the valve, the piston and the flange and the third repair shop (only in the 3-indenture model) is dedicated to the other items.

Summarised, we vary three parameters: the number of indentures, the failure cause probabili-

![Fig. 3. The indenture structure with two scenarios for the failure cause probabilities.](image)

Table 4

<table>
<thead>
<tr>
<th>Item</th>
<th>Pump A</th>
<th>Pump B</th>
<th>Valve</th>
<th>Flange</th>
<th>Piston</th>
<th>Stem</th>
<th>Gasket</th>
<th>Rod</th>
<th>Ring</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean service time ( E[S] ) relative to the mean service time of pump A</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>Prices (€) ( c_i )</td>
<td>5000</td>
<td>7100</td>
<td>200</td>
<td>300</td>
<td>500</td>
<td>120</td>
<td>60</td>
<td>50</td>
<td>80</td>
</tr>
</tbody>
</table>
ties and the repair shop utilisation rates (i.e., we have $2^3 = 8$ cases). We optimise all cases using 4 algorithms:

1. Inventory optimisation without repair priorities.
2. Nested priority assignment and inventory optimisation.
3. Sequential priority assignment and inventory optimisation in two variants:
   3.1. enumeration of all options in the priority assignment step;
   3.2. priority assignment based on the priority index $c_i/E[S_i]$.

In Table 5, we present for the various experiments the stock investment required to obtain an average system availability of 95%, the relative reduction using repair priorities and the algorithm run time (based on a PC with Pentium-III 800 MHz processor). Similar to Table 2, investment reductions are only comparable within the same row and not between rows, because the item repair times vary between rows (they are chosen such, that the specified utilisation rate is attained for the given assignment of items to repair shops).

We see that the sequential algorithm performs only slightly worse than the nested algorithm in terms of investment reduction, whereas it is much faster. Therefore, we can see that our assumption about minimisation of the total expected item cost within each repair shop is a good one and the sequential optimisation seems to be an attractive heuristic. The difference between both variants of the sequential algorithm is minor, since the number of item types per repair shop is always small (at most 4).

### 5.2. Investigating priority assignment

To examine the impact of the priority assignment algorithm in more detail, we need to run experiments in which more item types share the same repair shop capacity. Therefore, we focus on the three-indenture models. We only include both variants of the sequential algorithm, because the nested procedure requires excessive run times (as can also be seen from Table 5). To keep run times small, we focus on repair shops with $k = 1$ server.

We take scenario two for the failure cause probabilities and a repair shop utilisation of $\rho = 0.90$. We consider four variants for the assignment of items to repair shops:

1. three repair shops, one for each indenture level (i.e. with 2, 3 and 4 item types respectively);
2. two repair shops, one for indenture level 1 and 2 and one for indenture level 3 (i.e., with 5 and 4 item types respectively);
3. two repair shops, one for indenture level 1 and one for indenture levels 2 and 3 (i.e., with 2 and 7 item types respectively);
4. one repair shop for all item types (i.e., 9 item types sharing the same capacity).

We show the results of these experiments in Table 6. For both variants of priority assignment, we give the investment reduction compared to a non-priority system, the run time needed for the optimisation algorithm and the assignment of priorities to item types, H(igh) or L(ow). Note, that the “enumeration” is done only within repair shops. That is, the number of operations in this algorithm is $O(N^2m)$, where $N$ is the number of repair shops and $m$ is the maximum number of item types within one repair shop. Therefore, in cases with 3–4 item types per repair shop the run time is very short, although it is the “enumeration” case.

We see that the investment reduction is almost the same for both enumeration of priorities and ordering based on the index $c_i/E[S_i]$. Sometimes the assignment of item types to priority classes is different, but then the investment reduction is almost the same. However, priority assignment using ordering is much faster, particularly when many item types share the same repair shop capacity. Even for repair shops modelled as $M/M/1$ priority queues with 9 classes, the run time is still relatively small (17 seconds). Therefore we conclude that the sequential heuristic based on the priority index is a suitable heuristic for large models.

Further, we observe that most investment reduction is possible if the two pumps share repair capacity with many cheaper item types. Both pumps have relatively short repair times, and so
repair priority for these pumps diminish the number of pumps in repair (and so the required inventories) considerably. Because the pumps are also much more expensive than the other item types, this yields a large investment reduction. Hence repair priorities are especially attractive if many item types with completely different characteristics (item costs, service times) share repair shops with a high utilisation.

5.3. Validation of findings

Because the structure of the data is similar in the previous experiments, our findings may be case-dependent. Therefore, we examine whether we find similar results using a different data set. To this end, we use the data as provided by Hausman and Scudder (1982) on engines at a commercial airline. They construct a simulation model for a
three indenture, single location model with a single repair facility. All engine failures are caused by failure of one of the 23 components (third indenture items). The repair facility consists of 10 machines. Each item type passes some, but generally not all, machines following an item-specific route. The processing time at each machine is deterministic and differs for each item/machine combination. Obviously, this model is different from ours. Therefore, we only use the data from the repair shop by Hausman and Scudder (1982). We model the repair facility by a number of multi-server repair shops that are dedicated to a subset of item types. We use the total machine times from Hausman and Scudder as mean repair times and we assume that these repair times are exponentially distributed.

We consider five variants for the assignment of item types to repair shops, see Table 7. As we have mentioned in the introduction, our method to analyse priority queues (Slepetchenko et al., 2002b) is able to handle a limited number of item classes and servers only. Therefore, we are not able to consider a single repair shop for all item types, which would require a 10-server priority queue, where both priority classes have totally 23 subclasses (components). Then the state space becomes too large, so new and fast approximations have to be developed to handle such large repair shops.

In all experiments, we assume that engines fail with a frequency of 0.50 per day. In Table 8, we present the remaining data for our experiments, i.e. the mean repair times, prices and probabilities

Table 6
Priority assignment, investment reduction and algorithm run times for various repair shop structures

<table>
<thead>
<tr>
<th>Repair shop structure</th>
<th>Priority assignment</th>
<th>Investment reduction (run time)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Three repair shops: 2, 3 and 4 item types</td>
<td>enumeration</td>
<td>18.5% (23 s.)</td>
</tr>
<tr>
<td>Two repair shops: 5 and 4 item types</td>
<td>enumeration</td>
<td>58.0% (34 s.)</td>
</tr>
<tr>
<td>Two repair shops: 2 and 7 item types</td>
<td>enumeration</td>
<td>17.7% (13 s.)</td>
</tr>
<tr>
<td>One repair shop: 9 item types</td>
<td>enumeration</td>
<td>65.9% (56 m. 15 s.)</td>
</tr>
</tbody>
</table>

Table 7
Repair shop assignments and characteristics (numbers of servers and utilisation rates)

| Comp.11 | Comp.12 | Comp.13 | Comp.14 | Comp.21 | Comp.22 | Comp.23 | Comp.24 | Comp.25 | Comp.31 | Comp.32 | Comp.33 | Comp.34 | Comp.35 | Comp.36 | Comp.37 | Comp.38 | Comp.39 | Comp.40 | Comp.41 | Comp.42 | Comp.43 | Comp.44 | Comp.45 | Comp.46 | Comp.47 | Comp.48 | Comp.49 | Comp.50 | Comp.51 | Comp.52 | Comp.53 | Comp.54 | Comp.55 |
|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|
| Var. 1  |   k = 2, ρ = 0.84 | k = 2, ρ = 0.89 | k = 2, ρ = 0.91 | k = 2, ρ = 0.89 | k = 2, ρ = 0.83 |
| Var. 2  |   k = 3, ρ = 0.86 | k = 2, ρ = 0.89 | k = 2, ρ = 0.91 | k = 2, ρ = 0.89 | k = 2, ρ = 0.83 |
| Var. 3  |   k = 2, ρ = 0.84 | k = 2, ρ = 0.73 | k = 2, ρ = 0.68 | k = 2, ρ = 0.83 | k = 2, ρ = 0.83 |
| Var. 4  |   k = 1, ρ = 0.79 | k = 1, ρ = 0.90 | k = 1, ρ = 0.88 | k = 1, ρ = 0.86 | k = 1, ρ = 0.71 | k = 1, ρ = 0.69 | k = 1, ρ = 0.74 | k = 1, ρ = 0.62 |
| Var. 5  |   k = 3, ρ = 0.84 | k = 3, ρ = 0.73 | k = 2, ρ = 0.85 | k = 2, ρ = 0.91 |
that a failure is caused by a specific component. The key results are presented in Table 9.

In this table, we show the investment reduction, the assignment of priorities to item types and the price/service time ratio. The variants for repair shop assignment are shown using vertical lines. That is, all components between two successive vertical lines share the same repair shop. Note that

Table 9
Investment reduction using repair priorities with priority assignment for the Hausman and Scudder (1982) example (H = high priority, L = low priority)

| Invest. Reduc. | Comp. 11 | Comp. 12 | Comp. 13 | Comp. 14 | Comp. 21 | Comp. 22 | Comp. 23 | Comp. 31 | Comp. 32 | Comp. 33 | Comp. 34 | Comp. 35 | Comp. 36 | Comp. 37 | Comp. 41 | Comp. 42 | Comp. 43 | Comp. 44 | Comp. 51 | Comp. 52 | Comp. 53 | Comp. 54 | Comp. 55 |
|---------------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|
| Var. 1        | 17.9%   | H H L   | H L I   | H L L   | H L L   | H L L   | H L L   | H L L   | H L L   | H L L   | H L L   | H L L   | H L L   | H L L   | H L L   | H L L   | H L L   | H L L   | H L L   | H L L   | H L L   | H L L   | H L L   | H L L   |
| Var. 2        | 19.9%   | H H H   | H H L   | H H L   | H H L   | H H L   | H H L   | H H L   | H H L   | H H L   | H H L   | H H L   | H H L   | H H L   | H H L   | H H L   | H H L   | H H L   | H H L   | H H L   | H H L   | H H L   | H H L   | H H L   |
| Var. 3        | 20.7%   | H H H   | H H H   | H H H   | H H H   | H H H   | H H H   | H H H   | H H H   | H H H   | H H H   | H H H   | H H H   | H H H   | H H H   | H H H   | H H H   | H H H   | H H H   | H H H   | H H H   | H H H   | H H H   | H H H   |
| Var. 4        | 11.1%   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   |
| Var. 5        | 14.7%   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   | H L H   |
| Price / S.Time| 9.6 3.1 | 10.3 3.8 | 5.3 4.0 | 3.7 2.9 | 6.1 7.4 | 6.8 6.6 | 8.0 6.8 | 13.2 3.2 | 17.7 16.3 | 6.2 5.6 | 8.2 5.1 |
we do not distinguish between the two variants of sequential optimisation, enumeration and based on price service time ratio, because they give the same results in all five experiments.

We see that the investment reduction is not excessive, because the price/service time ratios of item types sharing the same repair capacity do not vary too much. Still, we find 10–20% investment reduction, similar to the non-extreme cases from the previous sections. As we expected, high priority is always assigned to item types with the highest price/service times ratios in the same repair shop. These results confirm our findings from the previous sections.

6. Conclusions and further research

We have shown that repair priorities theoretically provide an opportunity to reduce investments in the supply networks for repairable spare part, especially if the repair shop utilisation is high and many item types with different costs and repair times share the same repair capacity. We have presented algorithms for priority assignment and stock optimisation and we have shown that a sequential algorithm, where we optimise consequently the priority assignments based on the price/service times ratio and the stock levels for the spare parts given the assigned priorities, performs very well in terms of the investment reduction and algorithm run times. We summarised our preferred sequential heuristic at the end of Section 4.2.

The obtained results give us several interesting directions for further research, from both points of view: practical and theoretical. First, it is interesting to find out to what extent the investment reductions that we found can actually be achieved in practical circumstances. This requires additional case studies above to the cases referred to in this paper. In such cases, it has to be examined amongst others:

- Which utilisation rates are realistic in practice?
- What kind of shop one would like to have: one with very different item characteristics mixed in the same shops and with high utilisation or rather one with specialisation of shops and with lower utilisation?

From the theoretical point of view, the following directions of research are possible:

- Developing of faster algorithms to analyse multi-class, \( M/M/k \) priority queues, so that larger problem instances can be solved than we presented in this paper.
- Developing of methods to analyse more than two priority classes, and analysis of their advantages.
- Analysis of other priority disciplines: non-preemptive priorities, dynamic priorities, etc.

References


Sleptchenko, A., van Harten, A., van der Heijden, M.C., 2002b. Analyzing multi-class, multi-server queueing systems with preemptive priorities, Working paper, University of Twente, Faculty of Technology and Management, Enschede, the Netherlands, Submitted for publication.


