European Journal of Operational Research 222 (2012) 474-483

Contents lists available at SciVerse ScienceDirect

European Journal of Operational Research

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

Production, Manufacturing and Logistics

Joint optimization of level of repair analysis and spare parts stocks

R.J.I. Basten*, M.C. van der Heijden, J.M.J. Schutten

University of Twente, Faculty of Engineering Technology, P.O. Box 217, 7500 AE Enschede, The Netherlands University of Twente, School of Management and Governance, P.O. Box 217, 7500 AE Enschede, The Netherlands

A R T I C L E I N F O

ABSTRACT

Article history: Received 27 July 2011 Accepted 15 May 2012 Available online 9 June 2012

Keywords: Inventory Maintenance Level of repair analysis In the field of after sales service logistics for capital goods, generally, METRIC type methods are used to decide where to stock spare parts in a multi-echelon repair network such that a target availability of the capital goods is achieved. These methods generate a trade-off curve of spares investment costs versus backorders. Backorders of spare parts lead to unavailability of the capital goods. Inputs in the spare parts stocking problem are decisions on (1) which components to repair upon failure and which to discard, and (2) at which locations in the repair network to perform the repairs and discards. The level of repair analysis (LORA) can be used to make such decisions in conjunction with the decisions (3) at which locations to deploy resources, such as test equipment that are required to repair, discard, or move components. Since these decisions significantly impact the spare parts investment costs, we propose to solve the LORA and spare parts stocking problems jointly. We design an algorithm that finds efficient solutions. In order for the algorithm to be exact and because of its computational complexity, we restrict ourselves to two-echelon, single-indenture problems. In a computational experiment, we show that solving the joint problem is worthwhile, since we achieve a cost reduction of over 43% at maximum (5.1% on average) compared with using a sequential approach of first solving a LORA and then the spare parts stocking problem.

© 2012 Elsevier B.V. All rights reserved.

1. Introduction

In this paper, we discuss the maintenance of capital goods. Examples of capital goods are baggage handling systems at airports, radar systems on board naval vessels, MRI-scanners in hospitals, and wafer steppers that are used in the semiconductor industry. Capital goods can be defined as expensive and technologically advanced systems that are used to manufacture products or services. Because they are critical in the primary process of their users, their unavailability may lead to high costs. In other cases, their unavailability may lead to dangerous situations, e.g., in the case of naval radar systems (military mission failure) or MRI-scanners (patients that cannot be treated).

To prevent downtime, capital goods are generally repaired by replacement, which means that a defective component is quickly taken out of the system and replaced with a functioning spare part. These spare parts may be located both close to the installed base, to reduce replacement times, and at more distant locations at a higher echelon level, to use risk pooling effects: one spare part can be used for various systems at various locations. To enable an exact analysis, we consider in this paper two-echelon distribution networks, so networks consisting of one central depot and a

* Corresponding author at: University of Twente, Faculty of Engineering Technology, P.O. Box 217, 7500 AE Enschede, The Netherlands. Tel.: +31 534894007; fax: +31 534893631. number of bases (sometimes referred to as a one warehouse, multiple retailer system).

Some defective components can only be discarded and replaced with a newly purchased component. For example, mechanical parts that wear or small parts such as screws cannot be repaired economically. Other parts may be either repaired or discarded. Repairs may be performed by replacing a subcomponent for which spare parts may be stocked. To facilitate an exact analysis, we do not consider such subcomponents here (i.e., we consider singleindenture product structures). The tactical problem (solved when the product is designed or deployed in the field) of determining which components to repair upon failure and which to discard is referred to as the *level of repair analysis (LORA)* problem in the military world (see, e.g., MIL-STD-1388-1A, United States Department of Defense, 1993). To be precise, it determines:

- 1. which components to repair upon failure and which to discard;
- 2. at which locations in the repair network to perform the repairs and discards;
- 3. at which locations to deploy resources required to repair, discard, or move components.

We refer to this set of decisions as the *LORA decisions* and to the first two decisions as the *repair/discard decisions*. The goal is to achieve the lowest possible costs, consisting of both fixed costs and costs that are variable in the number of failures. Fixed costs are due to the resources. They result from the LORA decisions,





E-mail address: r.basten@utwente.nl (R.J.I. Basten).

^{0377-2217/\$ -} see front matter @ 2012 Elsevier B.V. All rights reserved. http://dx.doi.org/10.1016/j.ejor.2012.05.045

but do not depend on the annual number of failures. Examples are training of service engineers and depreciation of repair equipment. Variable costs may include transportation costs, working hours of service engineers, and usage of bulk items.

In the spare parts literature and in practice, the LORA is generally solved first and next the spare parts stocking problem is solved: which spare parts to stock at which locations in which amounts, in order to achieve a target availability of the installed base. In the context of capital goods, generally METRIC type models and methods are used. A key idea in the METRIC type models is that the focus is not directly on the maximization of the availability, but instead, the focus is on the minimization of the expected number of backorders of components at the bases. A backorder occurs if a component is requested, but cannot be delivered immediately. As a result of a backorder, a system is unavailable waiting for spares. When referring to optimality in the spare parts stocking problem, it is meant that efficient solutions are found: at the corresponding (or lower) cost levels, it is not possible to achieve a lower expected number of backorders. The METRIC type models use a system approach, which means that spare parts for all components are considered jointly so that less expensive components can be stocked at relatively high levels, whereas more expensive components can be stocked at relatively low levels.

A problem with the sequential approach of first performing a LORA and then solving a spare parts stocking problem is that the decision (in the LORA) to repair or discard is based purely on the costs to perform a repair or discard, disregarding the fact that the lead times of both decisions may differ. If the lead time for discard is much higher than the lead time for repair (i.e., if purchasing a new component takes more time than repairing a component), then choosing discard may lead to higher number of spare parts to stock to achieve the same availability of the installed base. Although discarding may seem interesting in the LORA, considering the spare parts costs may mean that repairing turns out to be better. Unfortunately, having good estimates of the spare parts costs in the LORA is difficult, since it is hard to mimic a METRIC type system approach. As a result, many authors see the integration of spare parts stocking decision into the LORA problem as an important research direction (see, e.g., Brick and Uchoa, 2009; Basten et al., 2011a).

Our contribution is that we propose an algorithm to solve the problem of LORA and spare parts stocking jointly. This so-called integrated algorithm finds efficient solutions for two-echelon (or single-echelon), single-indenture problems. Using this algorithm, we show in a numerical experiment that solving the two problems jointly opposed to solving them sequentially leads to a cost reduction of over 43% at maximum and about 5.1% on average. Since a few percent cost reduction is already worth millions over the life time of capital goods, this means that it is worthwhile to consider the problem of LORA and spare parts stocks jointly. For general multi-echelon, multi-indenture problem instances it will be necessary to develop another algorithm, because of the computational complexity of our algorithm.

In Section 2, we give an overview of the related literature, and we outline our model in Section 3. In Section 4, we summarize key results on spare parts stocking that we need in Section 5, where we present our algorithm. In Section 6, we design a numerical experiment and discuss its results. In Section 7, we give conclusions and recommendations for further research.

2. Literature review

We first discuss the related literature on LORA, then that on spare parts stocking, and finally one paper on the joint problem of LORA and spare parts stocking.

To the best of our knowledge, Barros (1998) presents the first LORA model. She assumes that the same decisions are taken at all locations at one echelon level and that resources required to perform repairs are uncapacitated. Furthermore, all components at one indenture level require the same resource in order to be repaired. Barros (1998) models the problem as an integer linear programming model that she solves using CPLEX, a commercial solver. Barros and Riley (2001) consider the same model as Barros (1998) does, but solve it using a branch-and-bound method. Saranga and Dinesh Kumar (2006) propose a model that differs from the model by Barros (1998) in that each component requires its own unique resource. Saranga and Dinesh Kumar (2006) solve the model using a genetic algorithm. Basten et al. (2009) propose a formulation that generalizes the models by Barros (1998) and Saranga and Dinesh Kumar (2006) in that components may require any number of resources and resources may be shared by components. Basten et al. (2011a) then generalize the model by Basten et al. (2009) in that different decisions may be taken at various locations at the same echelon level. Basten et al. (2011a) model the problem as a minimum cost flow model with side constraints, which leads to a much lower computation time compared with Basten et al. (2009). Basten et al. (2011b) model various practically relevant extensions to the LORA model, using Basten et al. (2011a) as a basis. Brick and Uchoa (2009) combine the LORA problem with the decision of which facilities to open. They assume a two-echelon network structure and effectively assume a two-indenture product structure. In the last four papers, CPLEX is used to solve the model.

A vast amount of literature exists on the spare parts stocking problem. The seminal paper in this field is the work by Sherbrooke (1968) in which he develops the METRIC model (Multi-Echelon Technique for Recoverable Item Control). He considers a two-echelon distribution network with each location using one-for-one, or (S - 1, S) replenishment. He further considers a multi-item, singleindenture problem. Sherbrooke (1968) proposes an approximate evaluation of the number of backorders at the bases and a greedy heuristic to optimize the base stock levels. Muckstadt (1973) extends the work by Sherbrooke (1968) by allowing for two indenture levels, leading to the so-called MOD-METRIC model. Graves (1985) proposes a more accurate approximation for the two-echelon, single-indenture problem, the VARI-METRIC model, which Sherbrooke (1986) extends to two indenture levels. We refer to the books by Sherbrooke (2004) and Muckstadt (2005) for extensive overviews of these and related models and methods. There are also exact evaluations of these models: Simon (1971) considers the two-echelon, single-indenture, single-item problem, which is later extended to the general multi-echelon problem by Kruse (1979). Graves (1985) gives another exact evaluation of the same model. Axsäter (1990) provides an exact evaluation and enumerative, but relatively efficient optimization of the same model, but with penalty costs instead of a service level constraint. Rustenburg et al. (2003) give both an exact and approximate evaluation for the general multi-echelon, multi-indenture problem. They also give an extensive overview of the related literature.

To the best of our knowledge, there is only one paper in the literature in which a method is presented to solve the two problems of LORA and spare parts stocking jointly: Alfredsson (1997). The author considers two-echelon, single-indenture problems. Each component requires one specific tester (resource), which is required by one component only. Furthermore, one multi-tester exists that can be used for the repair of one component. Adapters can be added in a fixed order to enable the multi-tester to be used for the repair of additional components. Alfredsson (1997)'s resources have finite capacity, so system downtime includes waiting times for resources, repair times, and waiting times for spares. We base our optimization method on the method by Alfredsson (1997). However, since we assume that resources have infinite capacity, we are able to proof that our algorithm finds efficient solutions. Furthermore, we allow for more practically realistic component-resource relations: each component may require multiple resources and resources may be shared by multiple components.

3. Model

We model the joint problem of LORA and spare parts stocking. This means that we have to decide whether defective components should be discarded or repaired, and whether to do that at the base or at the central depot. Furthermore, we should decide where to locate resources and where to stock spare parts (in which amounts). The goal is to achieve the lowest possible costs, subject to a constraint on the availability. The costs include the variable and fixed LORA costs, as mentioned in Section 1, and the spare parts holding costs. In Section 3.1, we give our assumptions and in Section 3.2, we present the mathematical model formulation.

3.1. Assumptions

We use the assumptions underlying the METRIC type models (see, e.g., Muckstadt, 2005; Sherbrooke, 2004):

- Components fail according to a Poisson process with constant rate.
- For each component at each location, an (S 1, S) continuous review inventory control policy (one-for-one replenishment) is used.
- Replacement of a defective component by a functioning component takes zero time (implying that our availability measure considers the downtime waiting for spares only).
- The repair lead time includes the time used for sending the defective component to the repair location and for diagnosing the failure cause. The repair lead times for each component at each location are i.i.d. random variables (implying that resources are uncapacitated). See Fig. 1 for an overview of the lead times in the network.
- The replenishment lead times (for discarded components) for each component are i.i.d. random variables.
- The move lead times (to move a functioning component from the depot to a base) are deterministic.
- There are no lateral transshipments between bases, or emergency shipments from the central depot.
- Components may be stocked at both echelon levels only if components are repaired at the central depot, or if they are discarded. This is a common assumption in the METRIC type models. Stocking components at the central depot while repairing at base is undesirable for two reason. First, it is then probably better to also perform the repair at the central depot, thus reducing the number of required resources. Second, it would



Fig. 1. Lead times in the network.

lead to streams of functioning components going in both directions between the base and the central depot.

To ease the presentation in the remainder of this paper and to decrease the problem size, we make some additional assumptions, which are not critical for our algorithm:

- Repairs are always successful.
- Resources may be required to enable repair only, since resources that are required to enable discard or movement do not occur frequently in practice.
- The discard option is available at the central depot only. We do not consider the discard option at each base, since newly purchased components will in practice usually arrive at the central depot.

3.2. Mathematical model

Let there be N bases; at each base there is one system installed. We then define $L_1 = \{1, ..., N\}$ as the set of bases (the locations at echelon level 1) and we indicate the central depot by 0. We further define $L = \{0\} \cup L_1$ as the set of all locations in the network. Let C be the set of all components. The set D consists of the possible decisions that can be made for each component: $D = \{$ discard, repair, move $\}$. Move means that defective components will be moved to the central depot so that they can be repaired or discarded there. We define D_l as the possible decisions at location *l*: $D_l = D \setminus \{ \text{discard} \}$ for $l \in L_1$ and $D_0 = D \setminus \{ \text{move} \}$. For each component $c \in C$, we define $\lambda_{c,l}(>0)$ as the annual expected number of failures at base $l \in L_1$ and $\lambda_{c,0} = \sum_{l \in L_1} \lambda_{c,l}$ as the total annual expected number of failures at all bases, or, equivalently, as the maximum expected number of failures that may be repaired or discarded at the central depot. Let R be the set of resources. $\Omega_r \subseteq C$ is the set of components that require resource $r \in R$ in order to be repaired.

We define the following decisions variables:

 $X_{c,l,d}$ = the expected number of failures of component *c*

 $\in C$ at location $l \in L$ for which we make decision $d \in D_l$.

$$Y_{r,l} = \begin{cases} 1, & \text{if resource } r \in R \text{ is located at location } l \in L; \\ 0, & \text{otherwise.} \end{cases}$$

 $S_{c,l}$ = the number of spare parts of component $c \in C$ located at location $l \in L$.

For ease of notation, let \mathscr{X}_c be a two-dimensional vector with entries $X_{c,l,d}$.

We define three cost types. For component $c \in C$ at location $l \in L$, $vc_{c,l,d} \ge 0$) are the variable costs per expected failure if we make decision $d \in D$. Since we have chosen, without loss of generality, to minimize the total annual costs with our definition of $\lambda_{c,l}$, we define $fc_{r,l} \ge 0$) to be the annual fixed costs to locate resource $r \in R$ at location $l \in L$ and we define $hc_{c,l}(>0)$ to be the annual costs of holding one spare of component $c \in C$ at location $l \in L$.

The target availability that should be achieved is denoted by \overline{A} . With $\mathbb{E}B_{c,l}(S_{c,0}, S_{c,l}, \mathscr{X}_c)$ we denote the expected number of backorders for component $c \in C$ at base $l \in L_1$, given that there are $S_{c,0}$ and $S_{c,l}$ spare components c located at the central depot and at base l, respectively, and given the repair/discard decisions for component $c(\mathscr{X}_c)$. We will not explain here how to calculate this value, but we refer to, e.g., Graves (1985), Sherbrooke (2004), or Muckstadt (2005). Notice that the various lead times (for which we have not introduced notation) are taken into account when calculating the expected number of backorders. We define our model as follows:

$$\begin{array}{l} \text{minimize} \sum_{c \in C} \sum_{l \in L} \sum_{d \in D_l} \nu c_{c,l,d} \cdot X_{c,l,d} + \sum_{r \in R} \sum_{l \in L} f c_{r,l} \cdot Y_{r,l} + \sum_{c \in C} \sum_{l \in L} h c_{c,l} \\ \cdot S_{c,l} \end{array}$$

$$(1)$$

subject to:

$$X_{c,l,\text{repair}} + X_{c,l,\text{move}} = \lambda_{c,l} \quad \forall c \in C, \quad \forall l \in L_1$$
(2)

$$X_{c,0,\text{discard}} + X_{c,0,\text{repair}} = \sum_{l \in L_1} X_{c,l,\text{move}} \quad \forall c \in C$$
(3)

$$X_{c,l,\text{repair}} \leqslant Y_{r,l} \cdot \lambda_{c,l} \quad \forall r \in \mathbb{R}, \quad \forall c \in \Omega_r, \quad \forall l \in L$$
(4)

$$\frac{1}{N} \sum_{l \in L_1} \prod_{c \in C} \left[1 - \mathbb{E}B_{c,l}(S_{c,0}, S_{c,l}, \mathscr{X}_c) \right] \ge \overline{A}$$
(5)

 $X_{c,l,d} \ge 0 \quad \forall c \in C, \quad \forall l \in L, \quad \forall d \in D$ (6)

$$Y_{r,l} \in \{0,1\} \quad \forall r \in R, \quad \forall l \in L$$
(7)

$$S_{c,l} \in \mathbb{N}_0 = \mathbb{N} \cup \{0\} \quad \forall c \in C, \quad \forall l \in L$$
(8)

Constraints (2)-(4) are the 'LORA constraints' and define the same model as Basten et al. (2011a) use, except that they model any number of echelon levels and indenture levels. Constraint (2) assures that for each component a decision is made at the base. If a component is moved. Constraint (3) assures that a decision is made for that component at the central depot. Some options are only available if all resources are present, which is guaranteed by Constraint (4). The value $\lambda_{c,l}$ in this constraint acts as a big M. Finally, Constraint (5) is the only 'spare parts stocking constraint' and it assures that the target availability is met. Notice that having a backorder at a base means that a capital good is not available. Since the availability is a non-linear function of the repair/discard decisions and the spare parts decisions, we have a large non-linear integer optimization problem that cannot be solved using standard optimization software such as CPLEX. We therefore propose a different algorithm in Section 5. Since it uses results from the METRIC theory on spare parts stocking, we first discuss that in Section 4.

4. Spare parts stocking

As mentioned in Section 1, the goal in the METRIC type methods is to locate spare parts such that a certain target availability is achieved against the lowest possible costs. Formally (notice that the repair/discard decisions are exogenously given here):

minimize
$$\sum_{c \in C} \sum_{l \in L} hc_{c,l} \cdot S_{c,l}$$

subject to:

$$\frac{1}{N} \sum_{l \in L_1} \prod_{c \in C} \left[1 - \mathbb{E}B_{c,l}(S_{c,0}, S_{c,l}) \right] \ge \overline{A} \quad S_{c,l} \in \mathbb{N}, \quad \forall c \in C, \quad \forall l \in L$$

This is achieved indirectly, by locating spare parts such that for a given cost level, the expected number of backorders (EBO) at the bases is minimized. Since having a backorder at a base means that a capital good is not available, minimizing EBO at bases is approximately equal to maximizing the availability of the capital goods (see, e.g., Sherbrooke, 2004; Muckstadt, 2005). A greedy algorithm is used that stocks increasingly more spares. This results in a socalled EBO-costs-curve B_g (we require the subscript later). The points $b_{g,i}$ ($i \in \{1, ..., |B_g|\}$) on such a curve represent spare parts stocking solutions that result in annual spare parts holding costs, $costs(b_{g,i})$, and a certain expected number of backorders, EBO($b_{g,i}$). At each step of the greedy algorithm, it is checked whether the target availability is met and as soon as this is the case, the algorithm terminates. **Definition 1.** An *EBO-costs-curve* B_g is an ordered set of points $b_{g,i}$ such that if $costs(b_{g,i}) < costs(b_{g,j})$, then i < j (if $costs(b_{g,i}) = costs(b_{g,j})$, then i = j).

To improve the readability of the remainder of this paper, we introduce $\Delta_{g,ij} = \frac{\text{EBO}(b_{g,i}) - \text{EBO}(b_{g,j})}{\cos(b_{g,j}) - \cos(b_{g,i})}$.

In a *single-item, single-location* problem, if increasingly more spare parts are stocked, the EBO will keep decreasing, but at a decreasing rate. In other words, the EBO is a decreasing, convex function of the number of spare parts (see, e.g., Sherbrooke, 2004; Muckstadt, 2005) (and thus of the annual spare parts holding costs if those are proportional to the number of stocked spare parts). Formally: If there are three points $b_{g,i}$, $b_{g,i}$, $b_{g,k} \in B_g$ such that i < j < k, then $\Delta_{g,i,j} \ge \Delta_{g,j,k}$. Obviously, the points $b_{g,i}$ on the constructed EBO-costs-curve are efficient solutions (meaning that it is not possible to find a point $b_{g,j}$ such that $costs(b_{g,j}) \le costs(b_{g,i})$). We formalize this well-known result without proof.

Lemma 1. Applying the greedy algorithm to a single-item, single-location spare parts stocking problem results in a convex EBO-costscurve consisting of efficient solutions.

If there are two echelon levels, the greedy algorithm functions as follows. Spare parts are first stocked at the central depot as explained above. For each spare parts level at the central depot (a point on the resulting EBO-costs-curve), spare parts are stocked at the bases. At each step in the construction of each of these EBO-costs-curves, it is checked at which base to stock the next spare part in order to get the highest backorder reduction per dollar ('biggest bang for the buck'). This results in as many EBO-costscurves as there are points on the curve at the central depot, see Fig. 2 for an example. Graves (1985) shows how to calculate the backorder levels at the bases, taking into account the backorder levels at the central depot. We introduce the lower envelope of a set of functions $f_i(x)$ as the function given by their pointwise minimum: $f(x) = \min_i f_i(x)$. Taking the lower envelope of the EBO-costscurves and removing all non-convex points (convexification) results in one convex EBO-costs-curve for the two-echelon problem, see Fig. 3 for an example. This EBO-costs-curve consists of efficient solutions (not necessarily all efficient solutions). Before formalizing this result in Lemma 3, we state Lemma 2.

Lemma 2. Taking the convexification of the lower envelope of a set of EBO-costs-curves, each consisting of efficient solutions, results in one convex EBO-costs-curve consisting of efficient solutions.

Proof. Let there be a set *G* of EBO-costs-curves. Taking the convexification of the lower envelope of these curves leads to the set $B_h = \{b_{h,j} | b_{h,j} \in \bigcup_{B_g \in G} B_g, \operatorname{costs}(b_{h,i}) < \operatorname{costs}(b_{h,j}) < \operatorname{costs}(b_{h,k}), \Delta_{h,i,j} \ge \Delta_{h,i,k}, \forall b_{h,i}, b_{h,k} \in \bigcup_{B_r \in G} B_g\}$. If there exist two points $b_{h,i}, b_{h,j} \in B_h$ such



Fig. 2. Multiple EBO-costs-curves (data from Sherbrooke (2004, Table 3.3)).



Fig. 3. Multiple EBO-costs-curves, including the convexification of the lower envelope.

that $costs(b_{h,i}) = costs(b_{h,j})$ and $EBO(b_{h,i}) = EBO(b_{h,j})$, we remove one at random. The set B_h represents a convex EBO-costs-curve, consisting of efficient solutions. \Box

Notice that we do not require the EBO-costs-curve to be related to either a single or multiple locations and to either a single or multiple components. Now we are ready to give Lemma 3. We do not provide the proof, but notice that the proof requires Lemma 2 and a slightly different version of Lemma 1.

Lemma 3. Applying the greedy algorithm to a single-item, twoechelon spare parts stocking problem results in a convex EBO-costscurve consisting of efficient solutions.

If there are *multiple components*, marginal analysis is used. We formalize how this is done in the proof of Lemmas 4, and 5 states the formal result. The idea is to start with constructing an EBO-costs-curve per component as described above. Next, an EBO-costs-curve is constructed for the total problem, with the first point having costs and EBO that are the total of the costs and EBO of the first points on each of the curves per component. Then, look at each of the EBO-costs-curves per component and add that spare part to stock that leads to the largest backorder reduction per invested dollar ('biggest bang for the buck'). Because of the convexity of the curves, such a myopic approach will lead to one convex EBO-costs-curve.

Lemma 4. Applying marginal analysis to a set *G* of convex EBO-costscurves, each consisting of efficient solutions for a spare parts stocking problem consisting of a set of components $C_{B_g}(B_g \in G)$, results in one convex EBO-costs-curve consisting of efficient solutions for the spare parts stocking problem for the set $\bigcup_{B_r \in G} C_{B_g}$.

Proof. Consider two convex EBO-costs-curves (B_1 and B_2), each consisting of efficient solutions for a spare parts stocking problem for a set of component C_{B_1} and C_{B_2} , respectively ($C_{B_1} \cap C_{B_2} = \emptyset$). These two curves are merged using marginal analysis as follows. The first point on the resulting EBO-costs-curve (B_3) is $b_{3,1}$ with $costs(b_{3,1}) = costs(b_{1,1}) + costs(b_{2,1})$ and $EBO(b_{3,1}) = EBO(b_{1,1}) + E-BO(b_{2,1})$. Next, consider the second point on each of the two original curves. Two cases can be distinguished:

- If $\Delta_{1,1,2} \ge \Delta_{2,1,2}$, then the second point on the resulting EBOcosts-curve is $b_{3,2}$ with $costs(b_{3,2}) = costs(b_{1,2}) + costs(b_{2,1})$ and $EBO(b_{3,2}) = EBO(b_{1,2}) + EBO(b_{2,1})$. Of course, $\Delta_{3,1,2} = \Delta_{1,1,2}$. In the next step, compare $\Delta_{1,2,3}$ with $\Delta_{2,1,2}$.
- Otherwise, the second point on the resulting EBO-costs-curve is $b_{3,2}$ with $costs(b_{3,2}) = costs(b_{1,1}) + costs(b_{2,2})$ and $EBO(b_{3,2}) = E-BO(b_{1,1}) + EBO(b_{2,2})$. Then, $\Delta_{3,1,2} = \Delta_{2,1,2}$. And in the next step, compare $\Delta_{1,1,2}$ with $\Delta_{2,2,3}$.

In either case, because of the convexity of both curves, it is guaranteed that both fractions that we are comparing in the next step are smaller than $\varDelta_{3,1,2}$. Therefore, proceeding in this way, we are certain to get a convex EBO-costs-curve consisting of efficient solutions for the spare parts stocking problem for the problem consisting of the set of components $C_{B_1} \cup C_{B_2}$. Notice furthermore that marginal analysis is an *associative* operator. This means that applying marginal analysis to n + 1 sets of components leads to the same result as applying it first to n sets of components. Therefore, the result holds for any number of sets of components.

Lemma 5. Applying marginal analysis to |C| convex EBO-costs-curves, each consisting of efficient solutions for a single-item $c \in C$, twoechelon spare parts stocking problem, results in one convex EBOcosts-curve consisting of efficient solutions for the spare parts stocking problem for the total set C.

Proof. This follows directly from Lemma 4.

We formalize a well known, important result (see, e.g., Sherbrooke, 2004; Muckstadt, 2005).

Theorem 1. Given a set of components, a two-echelon network, and LORA decisions, applying the greedy algorithm and marginal analysis to the spare parts stocking problem results in a convex EBO-costs-curve consisting of efficient solutions.

Proof. This follows directly from Lemmas 3 and 5. □

Since a discrete set of solutions is found, there is usually some *overshoot* over the target availability, meaning that a point is found that corresponds to an availability level that is somewhat higher than the target availability.

We restrict ourselves in this paper to two-echelon, singleindenture problems since enumeration is required to find efficient solutions for two-indenture problems and for general multi-echelon problems. This is too time-consuming to be a realistic approach for any but very small problems (see Appendix A to see what may go wrong when using a greedy algorithm).

5. Algorithm

Consider an example problem instance, consisting of a central depot and two bases and the product structure as depicted in Fig. 4: there are four components, component c_1 requires resource r_1 in order to be repaired, component c_2 requires both resources r_1 and r_2 , component c_3 requires both resources r_2 and r_3 , and component c_4 requires no resources ($\Omega_{r_1} = \{c_1, c_2\}, \Omega_{r_2} = \{c_2, c_3\}, \Omega_{r_3} = \{c_3\}$). As a result, the decision where to install resource r_2 depends on the decision where to install resource r_3 (and vice versa).

The basic idea of our *integrated algorithm* is to decompose the problem in a smart way and to aggregate the results. By fixing a resource to certain locations in the repair network, we are able to solve subproblems independently (e.g., by fixing resource r_2 in the example). We find a convex EBO-costs-curve, consisting of



Fig. 4. Example: multiple resources per component.

efficient solutions for a certain subproblem and we use either marginal analysis or the convexification of the lower envelope to merge the results. In this way, we find one convex EBO-costscurve, consisting of efficient solutions for the original problem. Below, we proof this result, but first we explain how our algorithm functions [example between brackets]:

- 1. Consider one resource and distinguish $2^{(N+1)}$ scenarios, each representing the resource being located at a subset of the locations in the repair network; we say that the resource is *fixed* [we choose to fix resource r_2 , resulting in $2^3 = 8$ scenarios; by fixing this resource, the problem may be decomposed into two subproblems later on]. We come back to the order in which resources may be fixed after the proof.
- 2. If there are still resources that interact, i.e., there exist components that require two or more resources that have not been fixed yet, repeat Step 1 for each of the $2^{(N+1)}$ scenarios (resulting in $2^{(N+1)} \cdot 2^{(N+1)}$ scenarios). Otherwise, proceed [we proceed; notice that this would not have been the case if we would have chosen to fix either resource r_1 or r_3].
- 3. Split the problem for each scenario into a subproblem consisting of one resource and those components that require that resource, and a subproblem consisting of the components that require no resources that have not been fixed yet [there are three subproblems: (1) resource r_1 and components c_1 and c_2 , (2) resource r_3 and component c_3 , (3) component c_4].
- 4. In each subproblem, distinguish $2^{(N+1)}$ scenarios in the same way as done in Step 1. For the subproblem consisting of no resources, consider one scenario [there are $2^3 \cdot (2^3 + 2^3 + 1)$ scenarios; 2^3 scenarios for resource r_2 and then 2^3 , 2^3 , and 1 scenarios for the three subproblems, respectively].
- 5. Divide each subproblem into a subproblem per component.
- 6. For each component consider each of the possible repair/discard decisions (repair at base, repair at depot, or discard) for each of the demand streams (*N* demand streams, so 3^{*N*} combinations) and stock spare parts for each of them, resulting in 3^{*N*} EBO-curves.
- 7. Take the convexification of the lower envelope of the curves per combination of repair/discard decisions, resulting in one EBO-costs-curve per component.
- Apply marginal analysis to merge the results per component into results per scenario, resulting in one EBO-costs-curve per scenario.
- 9. Take the convexification of the lower envelope of these curves, resulting in one EBO-costs-curve per resource.
- 10. Apply marginal analysis to merge the results per resource into results per scenario, resulting in one EBO-costs-curve per scenario.
- 11. Repeat Steps 9 and 10 until one EBO-costs-curve remains for the original problem.

There are two things to notice here:

- If there are no resources involved in the original problem, the algorithm starts at Step 3 and ends at Step 8.
- In Step 6, some repair decisions may not be available due to the locations of the resources that are required.

Theorem 2. Given a two-echelon problem of LORA and spare parts stocking, consisting of one or more components and each component requiring zero or more resources, applying the integrated algorithm results in a convex EBO-costs-curve, consisting of efficient solutions.

Proof. In Steps 1–5, the problem is increasingly divided into subproblems. Lemmas 3 and 2 state that Steps 6 and 7 result in a con-



Fig. 5. Graph of three resources.

vex EBO-costs-curve, consisting of efficient solutions for one component. Taking Steps 8–10 (9 and 10 possibly multiple times) leads to a convex EBO-costs-curve, consisting of efficient solutions for the total problem, using repeatedly the results stated in Lemmas 4 and 2. \Box

Notice that for our claim of finding efficient solutions, it does not matter in which order resources are fixed (Step 1). However, it does matter for the computation time. That is why we now discuss how to decompose a general problem with shared resources into independent subproblems so that the total problem can be solved in an efficient way. To this end, we represent the interaction between the resources in a graph, which we will call a *resource graph*: a vertex represents a resource, and an edge between two vertices exists if there exists a component that uses both resources. Fig. 5 represents the resource graph for the example. Below, we first give a number of definitions, then we explain how we use the graph representation of the interaction between the resources. We assume a familiarity with basic graph theory; for further definitions, we refer to any book on general graph theory (e.g., Godsil and Royle, 2001).

A graph G = (V, E) consists of vertices $v \in V$ and edges between vertices $(v, w) \in E$. A graph G' = (V', E'), with $V' \subseteq V$ and $(v,w) \in E' \Leftrightarrow (v,w) \in E, v \in V', w \in V'$, is called an *induced subgraph* of G. A graph is connected if any two of its vertices are linked by a path (the graph consisting of one vertex is connected as well). A maximal connected subgraph G' = (V', E') is an induced subgraph of G = (V, E), such that adding any more vertices $v \in V \setminus V'$ (and the required edges $(v, w) \in E \setminus E'$ to keep an induced subgraph) leads to a disconnected graph. A maximal connected subgraph is also called a *connected component* or just component; we will use the term graph component to avoid confusion. A depth first search can be used to identify the graph components in a graph (see, e.g., Hopcroft and Tarjan, 1973). A graph is complete if an edge exists between any two vertices: if for all vertices $v_1, v_2 \in V$ with $v_1 \neq v_2$ it holds that $(v_1, v_2) \in E$, then the graph G = (V, E) is complete.

We now define a *resource group* as the set of all resources represented by the vertices in one graph component in the resource graph. Let $|R^{\text{group}}|$ be the number of resources in such a resource group. Since fixing the locations for one resource leads to $2^{(N+1)}$ scenarios, we know that the number of possible combinations of locations for the resources in a resource group is $2^{(N+1)}$. The goal is to decompose any resource group in such a way that the smallest number of scenarios remains.¹ Notice that decomposition of the resource group is useful only if it is represented by a graph component that is not complete. We decompose the problem using a recursive approach on the graph representation of the resource group as follows.

¹ Notice that some scenarios may consist of many components, whereas others may consist of a few components only. One may want to incorporate this in the search for the best way of decomposing the resource group, but for sake of simplicity, we do not do that.

- Check whether the graph component, representing a resource group, is complete. If so, the number of scenarios is 2^{(N+1)^{|Rgroup|}}. Otherwise, go to Step 2.
- 2. For each vertex in the graph component, representing a resource, remove the vertex from the graph component.² Using a depth-first search, find the graph components in the new subgraph. For each of the new graph components, go to Step 1. The total number of scenarios is 2^(N+1) times the summation of the number of scenarios in each new graph components. Go to Step 3.
- 3. Over all the vertices that can be removed, choose the vertex that leads to the smallest number of scenarios. This vertex represents the resource that should be fixed.

6. Computational experiment

In this section, we use an extensive computational experiment to compare solving the joint problem using the integrated algorithm with solving the two problems sequentially. In Section 6.1, we give the most important characteristics of the problem instances that we use; a more extensive explanation of how the problem instances are generated can be found in Appendix B. Section 6.2 gives the results. We have implemented our work in Delphi 2007 and solve problems instances on an Intel Core 2 Duo P8600@2.40 GHz, with 3.5 GB RAM, under Microsoft Windows XP SP 3.

6.1. Generator

In order to obtain insights into the impact of using a joint approach instead of a sequential approach and to obtain insights into the various parameter settings, we have chosen to eliminate the impact of a-symmetrical networks. We thus restrict ourselves to symmetrical repair networks only, which means that the parameters are identical for all bases (e.g., $\lambda_{c,l} = \lambda_{c,k}$ for all $l, k \in L_1$). With a similar reasoning, we take the same decisions at all bases (e.g., $S_{c,l} = S_{c,k}$ for all $l, k \in L_1$). Notice that this strategy is optimal for symmetrical networks, except that the overshoot increases, see Section 4.

There are seven parameters that get two different values or ranges in our set of problem instances, the other parameters get a fixed value or range (see Tables 1 and 2 for the exact values). We generate ten problem instances per parameter combination to avoid basing conclusions on one exceptional case only. In total this leads to $10 \times 2^7 = 1280$ problem instances in the computational experiment.

6.2. Results

Table 3 gives an overview of the results: compared with solving the two problems sequentially, solving them using the integrated algorithm results in a cost reduction of 5.07% on average and 43% at maximum. The two key reasons why cost reductions result are:

- Some components that require resources in order to be repaired, are repaired in the solution of the integrated approach, whereas they are discarded in the solution of the sequential approach. As the repair lead time is considerably less than the resupply lead time in our experiments, we need less spare parts if we repair. In rare cases, components are repaired at the bases instead of at the central depot in order to reduce the lead time.
- Components that do not require resources in order to be repaired, are always repaired at the base in the sequential

Table 1	
Fixed values.	
Denementen	

Parameter	Value (s)
# Bases	5
# Components	100
# Resources	10

Table		2

Values that vary over a range.

Parameter	Range (s)
Annual demand of component	[0.01; 0.1]
Net cost of component	[1000; 10,000] and [1000; 100,000]
Discard costs	[75%; 125%]
Repair costs	[25%; 75%]
Move costs	[1%; 1%]
Annual holding costs	[20%; 20%] and [20%; 40%]
Annual cost of resource	[10,000; 100,000] and [10,000; 500,000]
Repair time (in years)	[0.5/52; 4/52] and [2/52; 4/52]
Move time (in years)	[2/365;4/52] and [1/52;4/52]
# Components per resource	[2;3] and [2;6]

solution, because there is no reason to induce move costs in order to repair the components at the central depot (notice that usually some resources will be required for a repair, but only the expensive, specific resources are considered in the LORA problem since other resources will be available anyhow). However, in our model (and generally in the METRIC type models), spare parts may only be stocked at locations where the spare parts are repaired or downstream in the network, which means that if repairs are performed at the bases, spare parts may only be stocked there. In the integrated solution, quite some of these repairs are performed at the central depot so that risk pooling effects can be used by stocking spare parts there.

The second reason seems to be the most important reason why cost reductions are achieved. To see this, look at the results for the problem instances in which each resource is required by 2–3 components ([2;3]) in Table 4. Using the integrated approach, more repairs are performed at the central depot (echelon level 2) and in total, compared with using the sequential approach. However, the increase in the total number of repairs is only minor (the increase in installed number of resources is also minor; this is not shown in the tables); the main difference results from changing the repair location from echelon level 1 to echelon level 2 for many components.

Since there are more components that do not require any resource when 2–3 components require each resource, than when 2–6 components require each resource (the number of resources is not changed), the repair strategy of many more components is changed in the former case than in the latter case, leading to an average cost reduction of 9.2% and 0.9%, respectively.

Another interesting thing to notice is that the cost reduction that may be achieved is especially large if the move lead time is low, see Table 5. If this lead time is drawn from a uniform distribution on the interval [2/365; 4/52], the average cost reduction is 7.6%, whereas it is only 2.5% when the interval [1/52; 4/52] is used. The reason is that if the move lead time is low, then the increase in lead time is small when a component is repaired at echelon level 2 (in the solution of the integrated algorithm) instead of at echelon level 1 (in the solution of the sequential approach). This means that the advantage of being able to use risk pooling effects more easily outweighs the disadvantage of a higher lead time.

The effects of the other parameter settings are much smaller. The next biggest effect results from a change in the repair lead time, which is shown in Table 5. Two last remarks:

² Do not remove a vertex that is connected to one other vertex only, since that does not help in decomposing the graph component.

Table 3

Overview of the results.

Algorithm used	Optimization ti	me in seconds	Cost reduction co	mpared with sequential	Average availability (%)
	Average	Maximum	Average	Maximum	
Sequential	0.02	0.13	-	-	95.14
Integrated	0.11	0.64	5.07%	43.26%	95.11

Table 4

% Of demand that is repaired.

# Components per resource	Sequential		Integrated	Integrated		
	Ech. 1 (%)	Ech. 2 (%)	Total (%)	Ech. 1 (%)	Ech. 2 (%)	Total (%)
[2;3]	78.6	0.1	78.8	62.7	16.8	79.5
[2;6]	67.1	1.1	68.2	65.5	4.3	69.8

Table 5

Results for the move and repair lead times.

Lead time	Setting	Cost reduction	
		Average (%)	Maximum (%)
Move	[2/365;4/52]	7.62	43.26
	[1/52;4/52]	2.52	18.07
Repair	[0.5/52; 4/52]	4.33	35.82
	[2/52; 4/52]	5.81	43.26

Table 6

Cost reduction of joint approach compared to sequential approaches.

	No estimate	Alternative 1	Alternative 2
	(%)	(%)	(%)
Average cost reduction Maximum cost reduction	5.07 43.26	5.02 43.26	5.34 43.26

- We have tried to improve the sequential approach by incorporating an estimate of the spare parts holding costs in the LORA building block, but have not been able to do this in such a way that the results of the sequential approach did actually improve. We compare the results for two such alternative sequential approaches with the sequential approach that we have used (without an estimate) in Table 6 (notice that the lower the cost reduction that the joint approach achieves, the better the performance of the sequential approach). See Appendix C for an explanation of these alternative approaches.
- The optimization times that we show in Table 3 are low, but notice that due to the computational complexity of our algorithm (see Section 5), they would increase drastically when the problem would become more complex, e.g., when bases are not identical. However, as mentioned in Section 6.1, we have chosen to focus on symmetrical networks only so that we can clearly see how repair strategies change as a result of using the joint approach instead of the sequential approach, and to see the impact of various parameter settings.

7. Conclusions and further research

We have presented an algorithm that finds optimal solutions for the joint problem of LORA and spare parts stocking. Solving the problem using our integrated algorithm leads to a cost reduction of 5.07% on average and more than 43% at maximum in our numerical experiment compared with solving the two problems sequentially, which is done usually in both the literature and in practice. Therefore, we conclude that it is worthwhile to solve this problem integrally, especially if the lead time between the echelon levels is low. The cost reduction is achieved mainly by performing repairs at the central depot instead of at the bases (for parts that do not require resources) so that spare parts may be stocked at the central depot and thus risk pooling effects may be used. Performing repairs instead of discards and performing repairs at base instead of at the central depot has a much smaller effect on the achieved cost reduction.

For further research, it would be interesting to extend the model to multi-echelon, multi-indenture problems and find good heuristic procedures. The integrated algorithm that we have presented will not find efficient solutions anymore, but may still serve as a benchmark.

Acknowledgments

The authors thank the anonymous reviewers for their comments, which improved the original paper. The authors also gratefully acknowledge the support of the Innovation-Oriented Research Programme 'Integral Product Creation and Realization (IOP IPCR)' of the Netherlands Ministry of Economic Affairs, Agriculture and Innovation. The first author gratefully acknowledges the support of The Lloyd's Register Educational Trust, an independent charity working to achieve advances in transportation, science, engineering and technology education, training and research worldwide for the benefit of all.

Appendix A. Counter example optimality single-site, multiindenture problem

To understand the calculations in the example below, a basic knowledge of METRIC models is required. We refer to the books by Sherbrooke (2004) or Muckstadt (2005) for a detailed explanation. We also require some notation: Let $P{X_c \le k}$ denote the probability that over the repair or replenishment lead time of component *c*, the number of demands for component *c* is less than or equal to *k*. These Poisson probabilities are easily calculated.

Consider a single stock point where spare parts are stocked for a component (c_1) and its two subcomponents $(c_2 \text{ and } c_3)$. So, we aim to construct a curve of EBO of component c_1 versus total spare parts costs resulting from stocking components c_1 , c_2 , and c_3 . The failure rate per unit time of component c_1 is 4. Each failure can be repaired by replacing either one of the two subcomponents. Out of the four failures, on average 1.5 are caused by component

 c_2 and 2.5 are caused by component c_3 . Replacement of the defective subcomponent (so, repair of component c_1) takes zero time. Repair of components c_2 and c_3 both take 1 unit time. If no spare parts are stocked of any of the three components, we will thus have expected number of backorders of component c_1 of $1 \cdot 1.5 + 1 \cdot 2.5 + 0 \cdot 4 = 4$.

Consider two possible spare parts stocking solutions for the subcomponents, having the same holding costs per unit time (the holding costs per unit time for component c_2 are two third of those for component c_3):

- 1. Stock 1 spare component c_2 and 4 spare components c_3 , resulting in expected number of backorders for the subcomponents of 0.8939 (0.7231 and 0.1708 for component c_2 and c_3 , respectively). These backorders delay repairs of the parent component c_1 .
- 2. Stock 4 spare components c_2 and 2 spare components c_3 , resulting in expected number of backorders for the subcomponents of 0.8935 (0.0242 and 0.8694 for component c_2 and c_3 , respectively).

Constructing a convex EBO-curve would lead to removal of solution 1 (since it has a higher EBO than solution 2 and the same costs). If we do not stock any spare components c_1 , this is fine. However, if we stock one spare component c_1 , solutions 1 and 2 would lead to an expected number of backorders of component c_1 of 0.39 and 0.43, respectively. Solution 1 clearly dominates solution 2. This shows that generally, it is not possible to solve a subproblem for subcomponents first and use the resulting convex EBO-curve to solve the problem for the parent component. Instead, we should enumerate all possible solutions for the two subcomponents, construct an EBO-curve for the parent for each resulting solution, and take the lower envelope of the resulting curves. The computation time clearly explodes if the number of subcomponents per component increases.

A similar problem occurs in the case of general multi-echelon problems (more than two echelon levels). Although these results are often mentioned, both for the multi-indenture and general multi-echelon problems, we have never seen an example such as we have provided here.

Appendix B. Problem instances generator

We explain how we generate the problem instances that we use in our experiments. For each parameter that we use to generate these instances, we use the default setting in the text. Tables 1 and 2 give a complete overview of the possible settings. Some values are set to a certain value, others are drawn from a given distribution. These random values are the same for all settings of the other parameters. We use a full factorial design and we generate 10 problem instances for each combination of parameters to decrease the risk of basing conclusions on one odd problem instance. As a result, there are 1280 problem instances in total.

We use a two-echelon repair network that is completely symmetrical in the cost factors, the demand rates, and the throughput times. It consists of a central depot and five bases. The product structure consists of 100 components. The annual demand for a component is drawn from a uniform distribution on the interval [0.01; 0.1]. The component's price is drawn from a shifted exponential distribution with shift factor 1000 and rate parameter 7/ (10,000 – 1000). As a result, we do not have components with a price below 1000, since they are typically discarded by default. Furthermore, there are considerably more cheap components than expensive ones. On average 1% of the components get a value larger than 10,000, but we draw a new price for these components to

avoid odd problem instances. Using the calculated prices, we calculate the variable costs as follows:

- Repair costs as a fraction of the component price are drawn from a uniform distribution on the interval [0.25; 0.75].
- The discard costs as a fraction of the component price are drawn from a uniform distribution on the interval [0.75; 1.25]. These costs include the costs of purchasing a new component and the disposal costs or residual value of the defective component.
- The move costs as a fraction of the component price are 1%.
- The annual costs of holding one spare part of a component are 20% of the gross component price.

For each component, we draw a repair time from a uniform distribution on the interval [2/52; 4/52]. The discard time, the time it takes to order a component and receive it at the central depot, is drawn from a uniform distribution on the interval [1/10; 1/2]. Both the discard and the repair times vary over the components, but are the same at both echelon levels. The replenishment lead time between the two echelon levels is drawn from a uniform distribution on the interval [2/362; 4/52]. This value is the same for all components.

There are ten resources and their annual costs are drawn from a shifted exponential distribution with shift factor 10,000 and rate parameter 7/(100,000 - 10,000). We draw a new price for the resources that get a price higher than 100,000 to avoid odd problem instances. The number of components that requires a certain resource is drawn from a uniform distribution on the range [2;3]. These components are chose randomly.

Appendix C. Alternative sequential solutions

In our numerical experiment, we compare our joint approach with a sequential approach. The LORA building block in this sequential approach does not incorporate the spare parts holding costs that may result from a LORA decision. To improve the results of the sequential approach, we have therefore tried to incorporate an estimate of the spare parts holding costs in the LORA. We have done this by adding an estimate to the variable costs in two different ways:

- 1. Our first estimate covers the average pipeline only, so this can be seen as a lower bound on the actual spare parts requirements. Consider a repair/discard decision for a certain component, e.g., repair at the central depot. Take the total lead time that is related to this decision. For our example, this is the repair lead time at the central depot plus the lead time to move a repaired component back to the base. The estimate of the spare parts holding costs per failure for the component and repair/ discard decision is now this total lead time times the annual holding costs of this spare part.
- 2. For our second estimate we assume that the total target *un*availability over all components may be distributed evenly over all components. In a system approach, such as METRIC, more expensive parts will achieve a lower component target availability than less expensive parts. As a result, our estimate can be seen as an upper bound on the actual spare parts requirements.

If the availability target is, say, 0.95, then this target is achieved if each component achieves an availability of $0.95^{(\frac{1}{|C|})}$ (with |C|being the number of components in each system). The per component target availability can be achieved as follows: take the total lead time for a repair/discard option (as explained above) and calculate the average pipeline by multiplying this lead time by the failure rate. The expected number of backorders (EBOs) is equal to this average pipeline if there are no spare parts in the network. Keep adding spares at the base until the EBO is reduced such that 1-EBO is below the component target availability. Multiplying the resulting number of spare parts by the annual holding costs of these spare parts and dividing that by the failure rate gives our estimate of the spare parts holding costs per failure for the component and repair/discard decision.

References

- Alfredsson, P., 1997. Optimization of multi-echelon repairable item inventory systems with simultaneous location of repair facilities. European Journal of Operational Research 99, 584–595.
- Axsäter, S., 1990. Simple solution procedures for a class of two-echelon inventory problems. Operations Research 38 (1), 64–69.
- Barros, L.L., 1998. The optimization of repair decisions using life-cycle cost parameters. IMA Journal of Mathematics Applied in Business and Industry 9, 403–413.
- Barros, L.L., Riley, M., 2001. A combinatorial approach to level of repair analysis. European Journal of Operational Research 129 (2), 242–251.
- Basten, R.J.I., Schutten, J.M.J., van der Heijden, M.C., 2009. An efficient model formulation for level of repair analysis. Annals of Operations Research 172 (1), 119–142.
- Basten, R.J.I., van der Heijden, M.C., Schutten, J.M.J., 2011a. A minimum cost flow model for level of repair analysis. International Journal of Production Economics 133 (1), 233–242.
- Basten, R.J.I., van der Heijden, M.C., Schutten, J.M.J., 2011b. Practical extensions to a minimum cost flow model for level of repair analysis. European Journal of Operational Research 211, 333–342.

- Brick, E.S., Uchoa, E., 2009. A facility location and installation or resources model for level of repair analysis. European Journal of Operational Research 192 (2), 479– 486.
- Godsil, C., Royle, G., 2001. Algebraic Graph Theory. Springer, New York (NY).
- Graves, S.C., 1985. A multi-echelon inventory model for a repairable item with onefor-one replenishment. Management Science 31 (10), 1247–1256.
- Hopcroft, J., Tarjan, R.E., 1973. Efficient algorithms for graph manipulation. Communications of the ACM 16 (6), 372–378.
- Kruse, W.K., 1979. An Exact N Echelon Inventory Model: The Simple Simon Method. Technical Report 79-2, US Army Research Office.
- Muckstadt, J.A., 1973. A model for a multi-item, multi-echelon, multi-indenture inventory system. Management Science 20 (4), 472–481.
- Muckstadt, J.A., 2005. Analysis and Algorithms for Service Parts Supply Chains. Springer, New York (NY).
- Rustenburg, W.D., van Houtum, G.J., Zijm, W.H.M., 2003. Exact and approximate analysis of multi-echelon, multi-indenture spare parts systems with commonality. In: Shanthikumar, J.G., Yao, D.D., Zijm, W.H.M. (Eds.), Stochastic Modelling and Optimization of Manufacturing Systems and Supply Chains. Kluwer, Boston (MA), pp. 143–176.
- Saranga, H., Dinesh Kumar, U., 2006. Optimization of aircraft maintenance/support infrastructure using genetic algorithms – level of repair analysis. Annals of Operations Research 143 (1), 91–106.
- Sherbrooke, C.C., 1968. METRIC: a multi-echelon technique for recoverable item control. Operations Research 16 (1), 122–141.
- Sherbrooke, C.C., 1986. vari-metric: improved approximations for multi-indenture, multi-echelon availability models. Operations Research 34, 311–319.
- Sherbrooke, C.C., 2004. Optimal inventory modelling of systems, .. Multi-echelon Techniques, second ed. Kluwer, Dordrecht (The Netherlands).
- Simon, R.M., 1971. Stationary properties of a two-echelon inventory model for low demand items. Operations Research 19 (3), 761–773.
- United States Department of Defense, 1993. MIL-STD-1388-1A Logistics Support Analysis (Notice 4).