

Economic lot-sizing with remanufacturing: complexity and efficient formulations

Mathijn J. Retel Helmrich¹, Raf Jans², Wilco van den Heuvel¹, and
Albert P.M. Wagelmans¹

¹Erasmus School of Economics, Erasmus University Rotterdam,
P.O. Box 1738, 3000 DR Rotterdam, Netherlands, E-mail:
retelhelmrich@ese.eur.nl

²HEC Montréal, 3000, Chemin de la Côte-Sainte-Catherine, Montréal,
QC H3T 2A7, Canada

23 August 2010

ECONOMETRIC INSTITUTE REPORT EI 2010-71

Abstract

Within the framework of reverse logistics, the classic economic lot-sizing problem has been extended with a remanufacturing option. In this extended problem, known quantities of used products are returned from customers in each period. These returned products can be remanufactured, so that they are as good as new. Customer demand can then be fulfilled both from newly produced and remanufactured items. In each period, we can choose to set up a process to remanufacture returned products or produce new items. These processes can have separate or joint set-up costs. In this paper, we show that both variants are \mathcal{NP} -hard. Furthermore, we propose and compare several alternative MIP formulations of both problems. Because ‘natural’ lot-sizing formulations provide weak lower bounds, we propose tighter formulations, namely shortest path formulations, a partial shortest path formulation and an adaptation of the (l, S, WW) -inequalities for the classic problem with Wagner-Whitin costs. We test their efficiency on a large number of test data sets and find that, for both problem variants, a (partial) shortest path type

formulation performs better than the natural formulation, in terms of both the LP relaxation and MIP computation times. Moreover, this improvement can be substantial.

Keywords: Lot-sizing, remanufacturing, reverse logistics, reformulations

1 Introduction

Reverse logistics (see Dekker et al. (2004)) is a field that has emerged during the last decades. It studies situations in which there is not only a product flow towards the customers, but products and materials are also returned to the manufacturer and these may be reused in production processes. Within the framework of reverse logistics, the classic economic lot-sizing problem has been extended with a remanufacturing option. After showing that the economic lot-sizing problem with remanufacturing is \mathcal{NP} -hard, we shall propose several alternative formulations. Computational tests show that these improved formulations have better LP relaxations and MIP computation times, and allow the solution of larger problem instances than conventional formulations.

As in the classic problem, we face a deterministic demand from customers in a number of discrete time periods. In each period, we must decide to set up a production process or not, and if so how much to produce. In order to find a production plan with minimal costs, we must find the optimal balance between set-up, holding and production costs. In the problem extended with a remanufacturing option, known quantities of used products are returned from customers in each period. There is no demand for these returned products themselves (or ‘returns’ in short), but they can be remanufactured, so that they are as good as new. Customer demand can then be fulfilled from two sources, namely newly produced and remanufactured items. Since both can be used to serve customers, they are referred to as ‘serviceables’. We are to determine in which periods to set up a production process to remanufacture returned products and in which to set up a production process to manufacture new items. Thus, the traditional trade-off between set-up, holding and production costs is extended with remanufacturing costs and holding costs for returns. This setting is similar to the one described by Teunter et al. (2006), although we do not assume zero production and remanufacturing costs.

Practical situations in which the flow of returns can be modeled deterministically are described in Golany et al. (2001) and Beltrán and Krass (2002). Golany et al. (2001) mention that the demand for and returns of packaging and shipping materials (such as pallets or containers) are known, since the shipments in which they are used, are

planned in advance. Beltrán and Krass (2002) discuss catalogue retailing, in which ‘the proportion of each period’s sales that come back as returns, and the timing of these returns are often quite stable (...) making it possible to forecast returns in each period quite accurately’.

As in Teunter et al. (2006), we consider two variants of lot-sizing with remanufacturing. In the first variant, manufacturing new products and remanufacturing used products take place in two separate processes, each with its own set-up costs. We call this problem ELSRs (Economic Lot-Sizing with Remanufacturing and Separate set-ups). In the second variant, the manufacturing and remanufacturing process have one joint set-up cost, for instance because manufacturing and remanufacturing operations are performed on the same production line. We call this problem ELSRj (Economic Lot-Sizing with Remanufacturing and Joint set-ups).

ELSRj with time-invariant costs can be solved in $\mathcal{O}(T^4)$ time with the dynamic programming algorithm proposed in Teunter et al. (2006). However, in this paper we will show that ELSRj is \mathcal{NP} -hard in general. Moreover, we will prove that ELSRs is \mathcal{NP} -hard even if all costs are time-invariant.

Because of their complexity, it makes sense to look at good mixed integer programming (MIP) formulations of both problems, which is what we do in this paper. A first formulation with a ‘natural’ choice of variables was presented in Teunter et al. (2006) and will serve as our benchmark. We shall see, however, that such a formulation contains so-called ‘big M ’ constraints. It is generally known (Pochet and Wolsey, 2006) that these big M constraints in the natural lot-sizing formulation often lead to a bad LP-relaxation and hence high running times. Consequently, we propose several new, alternative formulations of the lot-sizing problem with remanufacturing. The first reformulation is based on a shortest path type formulation, as first proposed by Eppen and Martin (1987) for the capacitated lot-sizing problem (without remanufacturing). The second reformulation is a partial shortest path reformulation. This reformulation has fewer variables than the full shortest path reformulation, while preserving the quality of the LP-relaxation as much as possible. This idea was used by Van Vyve and Wolsey (2006) for the classic lot-sizing problem. The last formulation is based on the (l, S, WW) -inequalities, as introduced by Pochet and Wolsey (1994) for the single-item uncapacitated lot-sizing problem with Wagner-Whitin costs. In order to assess and compare their performances, we will subject all the formulations to a large number of computational tests.

To the best of our knowledge, no-one has ever presented and tested a good MIP formulation for the economic lot-sizing problem with remanufacturing. Previous work generally used heuristics or solved restricted versions of the problem. Van den Heuvel

(2006) solves ELSRs with a genetic algorithm that uses dynamic programming to solve subproblems in which the production periods are given. Teunter et al. (2006) present heuristics for both ELSRs and ELSRj. These heuristics are modifications of the well-known Silver-Meal, Least Unit Cost and Part Period Balancing heuristics (see Silver et al. (1998)). Recently, Schulz (2009) proposed an improvement of the modified Silver-Meal heuristic for ELSRs. Exact dynamic programming algorithms were developed by Pan et al. (2009) for several special cases of the capacitated lot-sizing problem with production, disposal and remanufacturing. This includes lot-sizing with uncapacitated production and capacitated remanufacturing and no final inventory of returns, for which their algorithm runs in exponential time. With this algorithm, they solve instances with up to 14 periods. Richter and Sombrutzki (2000) study a ‘reverse Wagner-Whitin model’ with time-invariant costs in which there is an abundance of returns. As such, manufacturing items is not necessary, but may result in a production plan with lower costs. The problem is solved with an algorithm similar to Wagner and Whitin’s. This model and algorithm are extended in Richter and Weber (2001) with variable (re-) manufacturing costs. In the case of time-invariant costs and demand inputs, they find an ‘optimal switching point’ between remanufacturing and manufacturing. Golany et al. (2001) study the lot-sizing problem with remanufacturing in which it is possible to dispose returned products. They show that the problem is \mathcal{NP} -hard for general concave costs, but solvable as a transportation problem in $\mathcal{O}(T^3)$ if all costs are linear. The same setting is studied in Yang et al. (2005). They extend the \mathcal{NP} -hardness result to the time-invariant costs case and develop a heuristic that runs in polynomial time. Piñeyro and Viera (2009) study a similar model with a disposal option, but the concave costs are restricted to fixed-plus-linear costs for (re-) manufacturing and disposing, and holding costs are assumed linear. They construct a tabu search procedure for this problem, as well as several inventory policies that run in $\mathcal{O}(T^2)$. Beltrán and Krass (2002) also consider a setting where disposal of returns is possible, but they assume that remanufacturing returned items is not necessary, i.e., returns can directly be used to satisfy demand. For this setting, they develop a dynamic programming algorithm that runs in $\mathcal{O}(T^3)$ time.

The remainder of this paper is organized as follows. The next section presents a formal definition of ELSRs and ELSRj by giving a first, ‘natural’ MIP formulation. In Section 3, we show that both ELSRs and ELSRj are \mathcal{NP} -hard in general. All of our reformulations are presented in Section 4. These formulations are put to the test in Section 5 and Section 6 concludes this paper, with some suggestions for further research.

2 The original formulation

2.1 Separate set-ups

We can formulate the lot-sizing problem with remanufacturing as a mixed integer program. A first, 'natural' formulation is based on the following decision variables:

x_t^m is the number of items manufactured in period t ;

x_t^r is the number of items remanufactured in period t ;

y_t^m is 1 if the manufacturing process is set up in period t ; 0 otherwise;

y_t^r is 1 if the remanufacturing process is set up in period t ; 0 otherwise;

I_t^s is the inventory of serviceables at the end of period t ;

I_t^r is the inventory of returns at the end of period t .

The notation that is used for the parameters in each period t , is as follows:

d_t is the customer demand, where $D_{i,j} := \sum_{t=i}^j d_t$;

r_t is the amount of returns, where $R_{i,j} := \sum_{t=i}^j r_t$;

h_t^s and h_t^r are the unit holding costs for serviceables and returns, respectively;

K_t^m and K_t^r are the set-up costs for manufacturing and remanufacturing, respectively;

p_t^m and p_t^r are the unit production costs for manufacturing and remanufacturing, respectively.

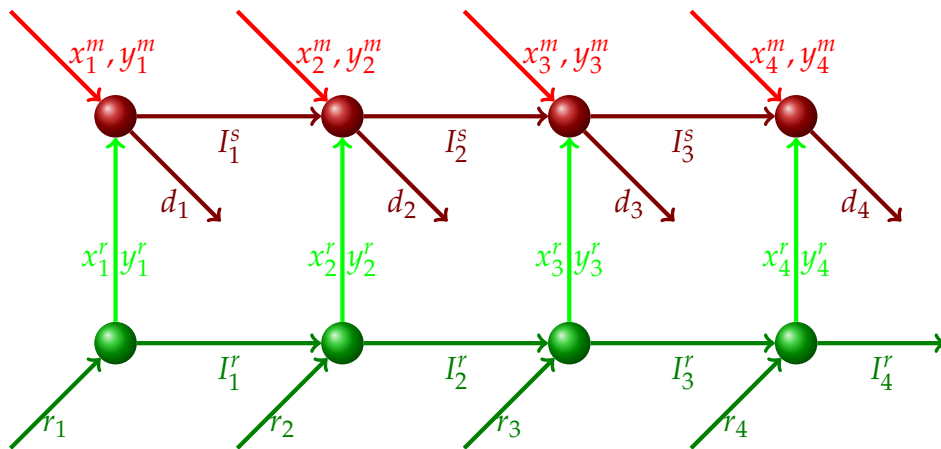


Figure 1: Network flow representation of ELSRs

A network flow representation of this problem and its variables and parameters is given in Figure 1. We are now ready to present a first, ‘natural’ formulation of the lot-sizing problem with remanufacturing and separate set-ups. This formulation is similar to the ones in Teunter et al. (2006), Yang et al. (2005) and Piñeyro and Viera (2009), and will serve as our benchmark.

$$\min \sum_{t=1}^T (K_t^m y_t^m + p_t^m x_t^m + h_t^s I_t^s + K_t^r y_t^r + p_t^r x_t^r + h_t^r I_t^r) \quad (1)$$

s.t.

$$I_t^s = I_{t-1}^s + x_t^m + x_t^r - d_t \quad t = 1, \dots, T \quad (2)$$

$$I_t^r = I_{t-1}^r - x_t^r + r_t \quad t = 1, \dots, T \quad (3)$$

$$x_t^m \leq D_{t,T} y_t^m \quad t = 1, \dots, T \quad (4)$$

$$x_t^r \leq D_{t,T} y_t^r \quad t = 1, \dots, T \quad (5)$$

$$x_t^m, x_t^r, I_t^s, I_t^r \geq 0 \quad t = 1, \dots, T \quad (6)$$

$$y_t^m, y_t^r \in \{0, 1\} \quad t = 1, \dots, T \quad (7)$$

$$I_0^s = I_0^r = 0 \quad (8)$$

We shall refer to this formulation as ‘Original’. It also serves as our (formal) definition of the economic lot-sizing problem with remanufacturing and separate set-ups (ELSRs).

The objective (1) is to minimize the sum of set-up costs of the production and remanufacturing processes, production and remanufacturing costs, and holding costs for serviceables and returns. (2) and (3) are inventory balance constraints for serviceables and returns, respectively. (4) and (5) are set-up forcing constraints for the manufacturing and remanufacturing processes. The last constraints (8) assume zero initial inventories of both serviceables and returns, without loss of generality.

2.2 Joint set-ups

For the problem variant with joint set-ups, we give a similar formulation. The notation is the same as before, but now we have only one set-up variable, y_t , and one parameter to denote the set-up costs, K_t .

$$\min \sum_{t=1}^T (K_t y_t + p_t^m x_t^m + h_t^s I_t^s + p_t^r x_t^r + h_t^r I_t^r) \quad (9)$$

s.t. (2), (3), (6), (8),

$$x_t^m + x_t^r \leq D_{t,T} y_t \quad t = 1, \dots, T \quad (10)$$

$$y_t \in \{0, 1\} \quad t = 1, \dots, T \quad (11)$$

We shall also refer to this formulation as ‘Original’. As before, it serves as our (formal) definition of the economic lot-sizing problem with remanufacturing and joint set-ups (ELSRj). The interpretation of the formulation is similar to the separate set-ups case.

3 Complexity results

3.1 Lot-sizing with remanufacturing and separate set-ups

Richter and Sombrutzki (2000) and Richter and Weber (2001) show that some special cases of the ELSRs problem can be solved in polynomial time. However, Richter and Sombrutzki (2000, p. 311) mention that “There are probably no simple algorithms to solve that general model ...”. In this section, we will show that the ELSRs problem is indeed \mathcal{NP} -hard in general. In the proof, we will use a reduction from the well-known \mathcal{NP} -complete PARTITION problem (see problem [SP12] in Garey and Johnson (1979)).

Problem PARTITION: Given n positive integers a_1, \dots, a_n . Does there exist a set $S \subset N = \{1, \dots, n\}$ such that $\sum_{i \in S} a_i = \sum_{i \in N \setminus S} a_i = A$? (Note that we may assume without loss of generality that $a_i < A$ for $i = 1, \dots, n$.)

Theorem 1 *The ELSRs problem is \mathcal{NP} -hard for time-invariant cost parameters.*

Proof Given an instance of PARTITION, we construct an instance of the ELSRs problem with $T = n$ periods as follows. For $t = 1, \dots, T$, let $d_t = a_t$, $K_t^m = K_t^r = 1$, $p_t^m = 1$, $p_t^r = 0$, $h_t^s = 3$ and $h_t^r = 0$. Furthermore, let $r_1 = A$ and $r_t = 0$ for $t = 2, \dots, T$. Clearly, this reduction can be done in polynomial time. We will show that the answer to PARTITION is positive if and only if the ELSRs instance has a solution with a cost of at most $T + A$.

Assume that we have a solution for the ELSRs instance with a cost of at most $T + A$. First, we show that we may restrict ourselves to a solution where no serviceables are held in stock. To that end, let t be the first period with serviceables in stock, so that t is a manufacturing or remanufacturing period. Now decreasing the number of items being (re)manufactured by one in period t and increasing the number of items being (re)manufactured by one in period $t + 1$ will reduce the total cost by at least 1. By repeating this process we end up with a solution without serviceables in stock and cost at most $T + A$.

Because at most A items can be remanufactured and all demand has to be satisfied, we incur at least a variable cost of A for manufactured items and we incur exactly a cost of A if all returns are remanufactured. Moreover, since no serviceables are held in stock and demand is positive, every period is a manufacturing or remanufacturing

period. So if there is both remanufacturing and manufacturing in at least one period, then the total setup costs will exceed T . Because the total cost is at most $T + A$, the total amount remanufactured equals A and demand in each period is satisfied by either manufacturing or remanufacturing (and not both). Therefore, the remanufacturing periods (or the manufacturing periods) form the set S .

Conversely, let S be the set for which $\sum_{i \in S} a_i = \sum_{i \in N \setminus S} a_i = A$. It is easy to verify that by remanufacturing a_t items in each period $t \in S$ and manufacturing a_t items in each period $t \in N \setminus S$, all demand is satisfied and total costs equal $T + A$. \square

Note that from a practical point of view, the ELSRs problem instance in the proof has reasonable assumptions on the cost parameters. Since remanufacturing adds value to an item, it is reasonable to assume that holding serviceables is at least as costly as holding returns (i.e., $h_i^s \geq h_i^r$). Furthermore, if remanufacturing is motivated economically, then the assumption that the unit remanufacturing cost equals at most the unit manufacturing cost (i.e., $p_i^m \geq p_i^r$) is also reasonable. Finally, in practice it is likely that the total amount of demand will be larger than the total amount of returns (i.e., $\sum_{t=1}^T d_t \geq \sum_{t=1}^T r_t$).

Note that the solution for the PARTITION instance and the optimal cost of the ELSRs instance are independent of the ordering of a_1, \dots, a_n (as in the \mathcal{NP} -completeness proof for the capacitated lot-sizing problem (Florian et al., 1980)). This gives the following corollary:

Corollary 2 *The ELSRs problem remains \mathcal{NP} -hard in the case of increasing (or decreasing) demand over time and time-invariant cost parameters.*

3.2 Lot-sizing with remanufacturing and joint set-ups

Although the lot-sizing problem with remanufacturing and joint set-ups can be solved in $\mathcal{O}(T^4)$ time with the algorithm presented in Teunter et al. (2006) when all costs are time-invariant, we show that ELSRj is \mathcal{NP} -hard in general.

Theorem 3 *The ELSRj problem is \mathcal{NP} -hard.*

Proof We show that the lot-sizing problem with remanufacturing and separate set-ups is a special case of the problem with joint set-ups. Let an instance of ELSRs be defined as in (1)-(8). We define an instance of the lot-sizing problem with remanufacturing and

joint set-ups as follows:

$$\begin{aligned} \tilde{T} &= 2T \\ \tilde{d}_t &= \begin{cases} 0 & \text{for } t \text{ odd} \\ d_{\frac{1}{2}t} & \text{for } t \text{ even} \end{cases} \\ \tilde{p}_t^m &= \begin{cases} \infty & \text{for } t \text{ odd} \\ p_{\frac{1}{2}t}^m & \text{for } t \text{ even} \end{cases} \\ \tilde{h}_t^s &= \begin{cases} 0 & \text{for } t \text{ odd} \\ h_{\frac{1}{2}t}^s & \text{for } t \text{ even} \end{cases} \end{aligned} \quad \begin{aligned} \tilde{K}_t &= \begin{cases} K_t^r & \text{for } t \text{ odd} \\ K_t^m & \text{for } t \text{ even} \end{cases} \\ \tilde{r}_t &= \begin{cases} r_{\frac{1}{2}(t+1)} & \text{for } t \text{ odd} \\ 0 & \text{for } t \text{ even} \end{cases} \\ \tilde{p}_t^r &= \begin{cases} p_{\frac{1}{2}(t+1)}^r & \text{for } t \text{ odd} \\ \infty & \text{for } t \text{ even} \end{cases} \\ \tilde{h}_t^r &= \begin{cases} 0 & \text{for } t \text{ odd} \\ h_{\frac{1}{2}t}^r & \text{for } t \text{ even} \end{cases} \end{aligned}$$

Note that the parameters with tilde correspond to ELSRj, whereas the ones without correspond to ELSRs. An illustration of such an instance of ELSRj can be found in Figure 2. Since this problem has joint set-up costs, there is a common fixed charge

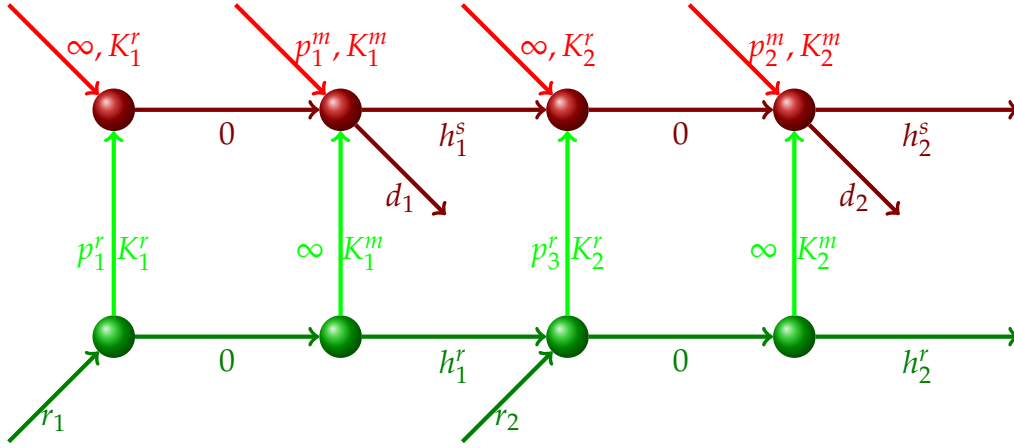


Figure 2: ELSRs as a special case of ELSRj

$(K_1^r, K_1^m, K_2^r, K_2^m, \dots)$ on *two* arcs in each period. Observe that each period t in ELSRs corresponds to a two-period pair $(2t - 1, 2t)$ in ELSRj. In the first period of such a two-period pair, the returned products become available and in the second customer demand takes place. Inventory of both serviceables and returns can be carried between two such periods without costs. Furthermore, remanufacturing will only take place in the first and manufacturing only in the second period. In accordance with this, we have chosen $\tilde{K}_{2t-1} = K_t^r$ and $\tilde{K}_{2t} = K_t^m$. Since all other parameters in the instance of ELSRj correspond directly to their counterparts in ELSRs, it is easy to see that ELSRs is indeed a special case of ELSRj. Since this reduction can clearly be performed in polynomial time, it follows that Theorem 3 holds. \square

4 Reformulations

In equations (2)-(8), we can see that the natural formulation contains two ‘big M ’ type constraints. It is generally known that these big M set-up constraints in lot-sizing often lead to a bad LP-relaxation (Pochet and Wolsey, 2006). In order to obtain better lower bounds, we propose several alternative formulations of the lot-sizing problem with re-manufacturing, namely a shortest path reformulation (in Section 4.1), a partial shortest path reformulation (in Section 4.2) and a formulation that uses an adaptation of the (l, S, WW) inequalities (in Section 4.3).

4.1 The shortest path reformulation

The formulation presented in this section is based on a shortest path reformulation, as proposed by Eppen and Martin (1987) for the capacitated lot-sizing problem. They solved a shortest path problem in a network with flow variables $z_{i,j}$ (where $i \leq j$) through which a unit flow is sent. For three periods, this network corresponds to (only) the $z_{i,j}^{sm}$ variables in Figure 3. An example of a feasible solution in this network is $z_{1,2} = \frac{1}{3}$, $z_{1,3} = \frac{2}{3}$, $z_{3,3} = \frac{1}{3}$, and $z_{i,j} = 0$ otherwise. This means that in period 1, we produce $\frac{1}{3}$ of the demand in periods 1 and 2, and $\frac{2}{3}$ of the demand in periods 1, 2 and 3. In other words: all demand in periods 1 and 2, and $\frac{2}{3}$ of the demand in period 3 are satisfied by items produced in the first period. Finally, the remaining $\frac{1}{3}$ of the demand in period 3 is produced in period 3 itself. Notice that we start with a flow of one at the first node and that in each node the inflow equals the outflow. In our example, we have a set-up in periods 1 and 3, and this corresponds exactly to the nodes with a nonzero outflow. Moreover, observe that in each period i , we can compute the production quantities as $x_i = \sum_{t=i}^T D_{i,t} z_{i,t}$. Using this relation between the x and z variables, the production and holding costs on each arc $z_{i,j}$ can be computed exactly. For the classic (single-item uncapacitated) lot-sizing problem, the LP relaxation of the shortest path formulation always gives an integer solution, i.e., the optimal solution of the classic lot-sizing problem. The problem with re-manufacturing can be viewed as having two products: serviceables and returns. A shortest path type reformulation can be applied to both.

4.1.1 Separate set-ups

When formulating the layer of serviceables as a shortest path problem, one should note that there are two sources from which demand can be fulfilled, newly produced and remanufactured products. Because both production processes have separate set-up

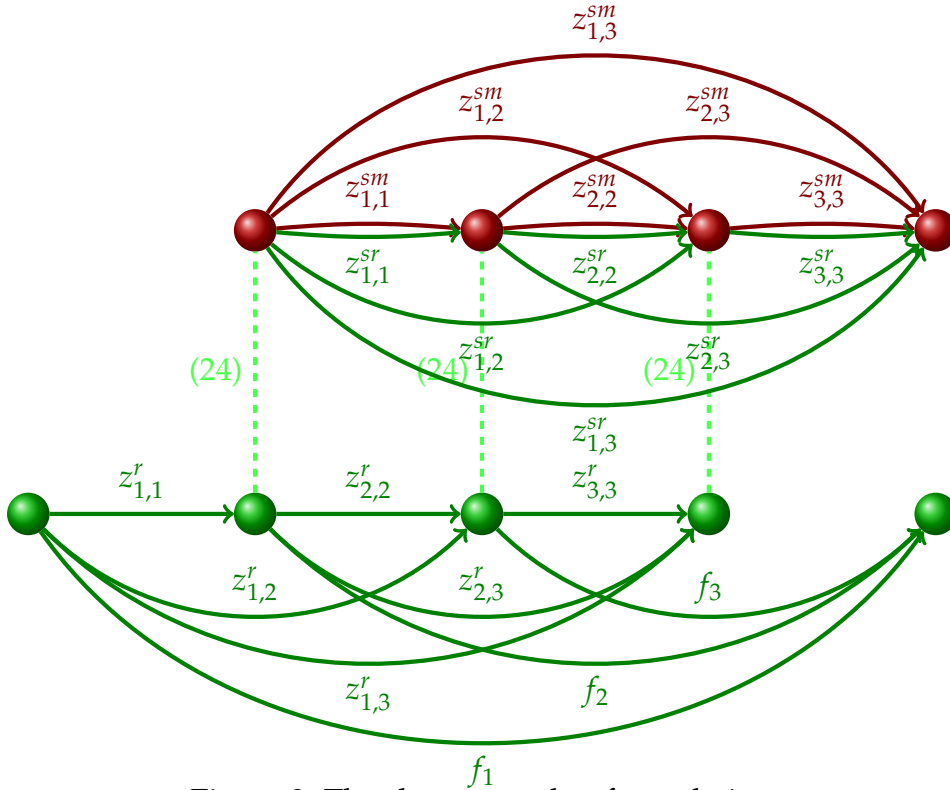


Figure 3: The shortest path reformulation

costs (and hence separate binary variables, y_t^m and y_t^r), we also need two types of flow variables (as opposed to one in Eppen and Martin's original shortest path reformulation). Call these flow variables $z_{i,j}^{sm}$ and $z_{i,j}^{sr}$. Here, $z_{i,j}^{sm}$ ($z_{i,j}^{sr}$) is defined as the fraction of demand in *each* of the periods i until j that is fulfilled by newly produced (remanufactured) items in period i .

When formulating the layer of returns as a shortest path problem, one should note that this is exactly the classic lot-sizing problem, but with the time reversed. In the classical case, production in some period t is used to satisfy given demand in future periods $t, t + 1, \dots$. Here however, there is a given amount of returns in each period that is remanufactured in some future period t . The variable $z_{i,j}^r$ is defined as the fraction of returns in *each* of the periods i until j that is remanufactured in period j . This formulation also provides the opportunity to have a final inventory of returns, i.e., not all returns need to be remanufactured within the problem horizon. For this purpose, define f_t ($t \in \{1, \dots, T\}$) as the fraction of returns in *each* of the periods t until T that is added to the final inventory of returns at the end of period T . Following this definition, we can say that $I_T^r = \sum_{t=1}^T R_{t,T} f_t$. A shortest path reformulation with three periods is depicted in the graph in Figure 3.

Before giving the objective function and constraints, we define the following cost

parameters.

$$C_{i,j}^{sm} = p_i^m D_{i,j} + \sum_{t=i}^{j-1} h_t^s D_{t+1,j} \quad 1 \leq i \leq j \leq T \quad (12)$$

$$C_{i,j}^{sr} = p_i^r D_{i,j} + \sum_{t=i}^{j-1} h_t^s D_{t+1,j} \quad 1 \leq i \leq j \leq T \quad (13)$$

$$C_{i,j}^r = \sum_{t=i}^{j-1} h_t^r R_{i,t} \quad 1 \leq i \leq j \leq T \quad (14)$$

$$C_t^f = \sum_{j=t}^T h_j^r R_{t,j} \quad t = 1, \dots, T \quad (15)$$

We are now ready to present our shortest path formulation (SP) of ELSRs.

$$\min \left(\sum_{t=1}^T (K_t^m y_t^m + K_t^r y_t^r + C_t^f f_t) + \sum_{i=1}^T \sum_{j=i}^T (C_{i,j}^{sm} z_{i,j}^{sm} + C_{i,j}^{sr} z_{i,j}^{sr} + C_{i,j}^r z_{i,j}^r) \right) \quad (16)$$

s.t. (7) and

$$1 = \sum_{j=1}^T (z_{1,j}^{sm} + z_{1,j}^{sr}) \quad (17)$$

$$\sum_{i=1}^{t-1} (z_{i,t-1}^{sm} + z_{i,t-1}^{sr}) = \sum_{j=t}^T (z_{t,j}^{sm} + z_{t,j}^{sr}) \quad t = 2, \dots, T \quad (18)$$

$$\sum_{j=t}^T z_{t,j}^{sm} \leq y_t^m \quad t = 1, \dots, T \quad (19)$$

$$\sum_{j=t}^T z_{t,j}^{sr} \leq y_t^r \quad t = 1, \dots, T \quad (20)$$

$$1 = \sum_{j=1}^T z_{1,j}^r + f_1 \quad (21)$$

$$\sum_{i=1}^{t-1} z_{i,t-1}^r = \sum_{j=t}^T z_{t,j}^r + f_t \quad t = 2, \dots, T \quad (22)$$

$$\sum_{i=1}^t z_{i,t}^r \leq y_t^r \quad t = 1, \dots, T \quad (23)$$

$$\sum_{i=1}^t R_{i,t} z_{i,t}^r = \sum_{j=t}^T D_{t,j} z_{t,j}^{sr} \quad t = 1, \dots, T \quad (24)$$

$$z_{i,j}^{sm}, z_{i,j}^{sr}, z_{i,j}^r \geq 0 \quad 1 \leq i \leq j \leq T \quad (25)$$

Because we do not use the x -variables anymore, we have redefined the objective function as in (16). The shortest path constraints for the serviceables are given in equations (17)-(20). (17) and (18) are flow conservation constraints and (19) and (20) are set-up forcing constraints for the manufacturing and remanufacturing process, respectively.

The shortest path constraints for the returns are given in equations (21)-(23). (21) and (22) are flow conservation constraints and (23) is a set-up forcing constraint for the remanufacturing process. Constraint (24) links the z^r to the z^{sr} variables and hence the networks for serviceables and returns, which is illustrated by the dashed line in Figure 3. Finally, (25) are nonnegativity constraints.

Note that the shortest path formulation (SP) assumes nonzero demand in the first period. This can easily be overcome by excluding $z_{t,j}^{sm}$ and $z_{t,j}^{sr}$ from the summations on the left hand sides of (19) and (20) if $D_{t,j} = 0$, as in Pochet and Wolsey (2006, p. 223).

4.1.2 Joint set-ups

Because both production processes have joint set-up costs (and hence joint binary variables, y_t), we need only one type of flow variables when formulating the layer of serviceables as a shortest path problem (as opposed to two in the separate set-up case). Call these flow variables $z_{i,j}^s$. Here, $z_{i,j}^s$ is defined as the fraction of demand in *each* of the periods i until j that is fulfilled by remanufacturing or production of new items in period i . The shortest path constraints and corresponding objective function for the ELSRj problem are given in equations (26)-(34) below. Their interpretations are similar to the separate set-ups case.

$$\min \left(\sum_{t=1}^T (K_t y_t + C_t^f f_t) + \sum_{i=1}^T \sum_{j=i}^T (C_{i,j}^{sm} z_{i,j}^s + \tilde{C}_{i,j}^r z_{i,j}^r) \right) \quad (26)$$

s.t. (11) and

$$1 = \sum_{j=1}^T z_{1,j}^s \quad (27)$$

$$\sum_{i=1}^{t-1} z_{i,t-1}^s = \sum_{j=t}^T z_{t,j}^s \quad t = 2, \dots, T \quad (28)$$

$$\sum_{j=t}^T z_{t,j}^s \leq y_t \quad t = 1, \dots, T \quad (29)$$

$$1 = \sum_{j=1}^T z_{1,j}^r + f_1 \quad (30)$$

$$\sum_{i=1}^{t-1} z_{i,t-1}^r = \sum_{j=t}^T z_{t,j}^r + f_t \quad t = 2, \dots, T \quad (31)$$

$$\sum_{i=1}^t z_{i,t}^r \leq y_t \quad t = 1, \dots, T \quad (32)$$

$$\sum_{i=1}^t R_{i,t} z_{i,t}^r \leq \sum_{j=t}^T D_{t,j} z_{t,j}^s \quad t = 1, \dots, T \quad (33)$$

$$z_{i,j}^s, z_{i,j}^r \geq 0 \quad 1 \leq i \leq j \leq T \quad (34)$$

Constraint (33) links z^r to z^s . Note that the slack in this constraint is exactly the amount of products that is manufactured in period t . In the objective function (26), $C_{i,j}^{sm}$ and C_t^f are computed in the same way as in the separate set-ups case (see (12) and (15)) and $\tilde{C}_{i,j}^r$ is computed as

$$\tilde{C}_{i,j}^r = (p_i^r - p_i^m) R_{i,j} + \sum_{t=i}^{j-1} h_t^r R_{i,t} \quad 1 \leq i \leq j \leq T \quad . \quad (35)$$

4.2 The partial shortest path reformulation

The shortest path reformulations have $\mathcal{O}(T^2)$ variables and $\mathcal{O}(T)$ constraints, as opposed to the $\mathcal{O}(T)$ variables and $\mathcal{O}(T)$ constraints of the original formulation. Although $\mathcal{O}(T^2)$ variables is usually not considered an excessive amount for most applications, using the shortest path formulation in a branch-and-bound setting to solve large scale problem instances may lead to a large memory consumption. Moreover, one often has some prior knowledge about which of the flow variables will not be useful. For example, consider a problem instance in which the number of periods is large, say 75, but the set-up costs are relatively small compared to the holding costs. Now, it is unlikely that a variable as $z_{1,75}^r$ will have a value different from zero (since it would be cheaper to set up a new remanufacturing process in some period before period 75 to process the first period's returns than to keep them in stock for 74 periods). Of course, one possibility is to leave variables like $z_{1,75}^r$ out of the formulation altogether, but then the formulation is not correct anymore. We can overcome this shortcoming by using the ideas of Van Vyve and Wolsey (2006) (see also Pochet and Wolsey (2006)), which are related to a formulation proposed earlier by Stadtler (1997). Van Vyve and Wolsey (2006) describe a partial shortest path reformulation of the classic lot-sizing problem that is still correct. The basic idea is that we choose a parameter k , such that arcs covering less than k periods are reformulated with flow variables (i.e., $z_{i,j}$ only exists for $i \leq j < i + k$) and new variables are introduced to capture all arcs covering more than k periods (i.e., all $z_{i,j}$ with $j \geq i + k$ are aggregated in a new variable).

We apply this principle to lot-sizing with remanufacturing and separate set-ups only, although an extension to the problem with joint set-ups would be straightforward. Let k^s and k^r be the number of periods that are reformulated with flow variables in the layer of serviceables and returns, respectively. For $T = 4$ and $k^s = k^r = 2$, the partial shortest path reformulation can be represented by the graph in Figure 4.

For serviceables, $z_{i,j}^{sm}$ and $z_{i,j}^{sr}$ have the same interpretation as in the (full) shortest path formulation, but their domains are restricted to $j < i + k^s$. We define the following new

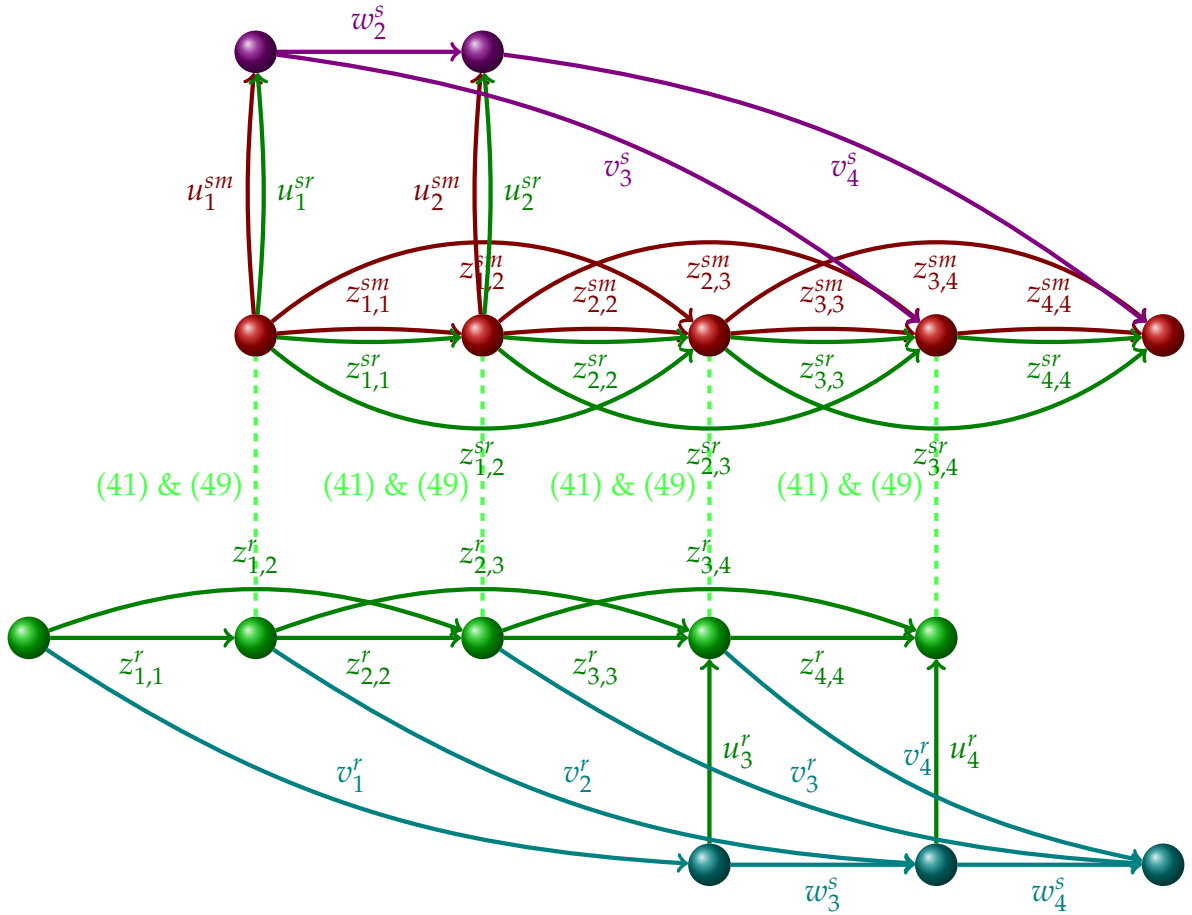


Figure 4: The partial shortest path reformulation

variables:

u_t^{sm} (u_t^{sr}) is the sum over all periods $j \geq t + k^s$, of the fractions of the cumulative demands in periods t until j , that are satisfied by items that are newly produced (remanufactured) in period t (for $t \leq T - k^s$);

v_t^s is the sum over all periods $i \leq t - k^s$, of the fractions of the cumulative demands in periods i until t , that are satisfied by items that are newly produced or remanufactured in period i (for $t \geq k^s + 1$);

w_t^s is the sum over all periods $i \leq t - 1$ and $j \geq t + k^s$, of the fractions of the cumulative demands in periods i until j , that are satisfied by items that are newly produced or remanufactured in period i (for $t = 2, \dots, T - k^s$).

For the sake of simplicity, we define $u_t^{sm}, u_t^{sr}, v_t^s, w_t^s = 0$ for all other values of t . The constraints for serviceables are given in equations (36)-(44). (36)-(39) define a shortest path problem. A unit flow through the network is ensured by (36) and (38). (39) and (37) are flow conservation constraints for the upper and second layer of nodes in

Figure 4, respectively. (40)-(42) provide lower bounds on the production and remanufacturing quantities, and inventory of serviceables, respectively. Since arcs covering more than k^s periods are aggregated, no exact amounts can be computed here. (43) and (44) are set-up forcing constraints for the manufacturing and remanufacturing process, respectively.

$$1 = \sum_{j=1}^{k^s} (z_{1,j}^{sm} + z_{1,j}^{sr}) + u_1^{sm} + u_1^{sr} \quad (36)$$

$$\sum_{i=\max\{1,t+1-k^s\}}^t (z_{i,t}^{sm} + z_{i,t}^{sr}) + v_t^s = \sum_{j=t+1}^{\min\{t+k^s,T\}} (z_{t+1,j}^{sm} + z_{t+1,j}^{sr}) + u_{t+1}^{sm} + u_{t+1}^{sr} \quad (37)$$

$$t = 1, \dots, T-1$$

$$\sum_{i=T+1-k^s}^T (z_{i,T}^{sm} + z_{i,T}^{sr}) + v_T^s = 1 \quad (38)$$

$$u_t^{sm} + u_t^{sr} + w_t^s = w_{t+1}^s + v_{t+k^s}^s \quad t = 1, \dots, T - k^s \quad (39)$$

$$x_t^m \geq \sum_{j=t}^{\min\{t+k^s-1,T\}} D_{t,j} z_{t,j}^{sm} + D_{t,t+k^s} u_t^{sm} \quad (40)$$

$$t = 1, \dots, T$$

$$x_t^r \geq \sum_{j=t}^{\min\{t+k^s-1,T\}} D_{t,j} z_{t,j}^{sr} + D_{t,t+k^s} u_t^{sr} \quad (41)$$

$$t = 1, \dots, T$$

$$I_{t-1}^s \geq \sum_{i=1}^{t-1} \sum_{j=t}^T D_{t,j} (z_{i,j}^{sm} + z_{i,j}^{sr}) + \sum_{j=t}^{\min\{t+k^s-1,T\}} D_{t,j} v_j^s \quad (42)$$

$$+ D_{t,t+k^s} w_t^s \quad t = 2, \dots, T$$

$$y_t^m \geq \sum_{j=t}^{\min\{t+k^s-1,T\}} z_{t,j}^{sm} + u_t^{sm} \quad t = 1, \dots, T \quad (43)$$

$$y_t^r \geq \sum_{j=t}^{\min\{t+k^s-1,T\}} z_{t,j}^{sr} + u_t^{sr} \quad t = 1, \dots, T \quad (44)$$

For returns, the variable $z_{i,j}^r$ has the same interpretation as in SP, but its domain is restricted to $i + k^r > j$. The other variables are:

u_t^r is the sum over all periods $i \leq t - k^r$, of the fractions of cumulative returns in periods i until t , that are remanufactured in period t (for $t \geq k^r + 1$);

v_t^r is the sum over all periods $j \geq t + k^r$, of the fractions of cumulative returns in periods t until j that are remanufactured in period j (for all t);

w_t^r is the sum over all periods $i \leq t - k^r$ and $j \geq t + 1$, of the fractions of cumulative returns in periods i until j , that are remanufactured in period j (for $t \geq k^r + 1$).

Again for simplicity's sake, we define $u_t^r, v_t^r, w_t^r = 0$ for all other values of t . The constraints for returns are given in equations (45)-(51). (45)-(48) are flow conservation constraints; (49) and (50) link the partial network variables to the original remanufacturing quantity and inventory variables; (51) is a set-up forcing constraint for the remanufacturing process. Note that the networks for serviceables and returns are linked by constraints (41) and (49), which is illustrated by the dashed line in Figure 4.

$$1 = \sum_{j=1}^{k^r} z_{1,j}^r + v_1^r \quad (45)$$

$$\sum_{i=\max\{1,t-k^r\}}^{t-1} z_{i,t-1}^r + u_{t-1}^r = \sum_{j=t}^{\min\{t+k^r-1,T\}} z_{t,j}^r + v_t^r \quad t = 2, \dots, T \quad (46)$$

$$\sum_{i=T+1-k^r}^T z_{i,T}^r + u_T^r = 1 \quad (47)$$

$$v_{t-k^r}^r + w_{t-1}^r = w_t^r + u_t^r \quad t = 1 + k^r, \dots, T \quad (48)$$

$$x_t^r \geq \sum_{i=\max\{1,t-k^r+1\}}^t R_{i,t} z_{i,t}^r + R_{t-k^r,t} u_t^r \quad t = 1, \dots, T \quad (49)$$

$$I_t^r \geq \sum_{j=t+1}^T \sum_{i=1}^t R_{i,t} z_{i,j}^r + \sum_{i=\max\{1,t-k^r+1\}}^t R_{i,t} v_i^r + R_{t-k^r,t} w_t^r \quad t = 1, \dots, T \quad (50)$$

$$y_t^r \geq \sum_{i=\max\{1,t-k^r+1\}}^t z_{i,t}^r + u_t^r \quad t = 1, \dots, T \quad (51)$$

These constraints are added to the Original formulation (1)-(8) to obtain formulation PSP. Altogether, this gives a mathematical formulation with $\mathcal{O}(k^s T + k^r T)$ variables. Of course, we still need to decide upon appropriate values of control parameters k^s and k^r , such that we sufficiently reduce the number of variables without deteriorating the LP-relaxation (too much). From quantities like the EOQ, we can obtain an approximation of the time between orders (TBO). Van der Laan and Teunter (2006) found a number of approximations of the order quantities for lot-sizing with remanufacturing, from which we have derived times between orders for our model. Note that although Van der Laan and Teunter (2006) studied a stochastic setting, their formulae were derived from the analysis of a deterministic model, like ours.

The results in Van der Laan and Teunter (2006) lead to the following times between orders:

$$TBO^s = \sqrt{\frac{2\bar{K}^s}{\bar{h}^s(\bar{d} - \bar{r})}} \quad \text{and} \quad TBO^r = \sqrt{\frac{2\bar{K}^r}{\bar{h}^r \bar{r}}} \quad , \quad (52)$$

where \bar{d} , \bar{r} , \bar{h}^s , \bar{h}^r , \bar{K}^s and \bar{K}^r denote the averages of d_t , r_t , h_t^s , h_t^r , K_t^s and K_t^r , respectively.

In the computational tests in Section 5, we will use $k^s = \lceil 2 \cdot TBO^s \rceil$ and $k^r = \lceil 2 \cdot TBO^r \rceil$, as well as $k^s = \lceil 3 \cdot TBO^s \rceil$ and $k^r = \lceil 3 \cdot TBO^r \rceil$. We call these formulations PSP2 and PSP3, respectively.

4.3 The (l, S, WW) valid inequalities

A different approach to improve the MIP formulation is to add valid inequalities to the Original formulation. A well-known set of strong valid inequalities for the classic (single-item uncapacitated) lot-sizing problem consists of the (l, S, WW) inequalities (Pochet and Wolsey, 1994). We adapt them for both the returns and serviceables layer of lot-sizing with remanufacturing.

In case of separate set-up costs, the following valid inequalities are added to the Original formulation (1)-(8) to obtain our (l, S, WW) formulation:

$$I_{i-1}^s + \sum_{t=i}^j D_{t,j} (y_t^m + y_t^r) \geq D_{i,j} \quad 2 \leq i \leq j \leq T \quad (53)$$

$$I_j^r + \sum_{t=i}^j R_{i,t} y_t^r \geq R_{i,j} \quad 1 \leq i \leq j \leq T \quad . \quad (54)$$

The intuition behind (53) is as follows: if at the beginning of period i the inventory (of serviceables) is insufficient to satisfy all demand in periods i until j , then we need to set up the manufacturing or remanufacturing process in some period within this interval. Moreover, if we do not have a set-up until period t , then there should be sufficient inventory in period i to satisfy demand in periods i until $t-1$. Inequality (54) has a similar interpretation, if we view the layer of returns as a lot-sizing problem with reversed time, as we did in Section 4.1.

In case of joint set-up costs, the following valid inequalities are added to the Original formulation ((9)-(11), (2), (3), (6), (8)) to obtain our (l, S, WW) formulation:

$$I_{i-1}^s + \sum_{t=i}^j D_{t,j} y_t \geq D_{i,j} \quad 2 \leq i \leq j \leq T \quad (55)$$

$$I_j^r + \sum_{t=i}^j R_{i,t} y_t \geq R_{i,j} \quad 1 \leq i \leq j \leq T \quad . \quad (56)$$

Their interpretations are similar to the problem with separate set-ups.

5 Computational tests

5.1 Test set-up

In order to gain insight into the performance of the different formulations, we randomly generated 360 problem instances, both for ELSRs and ELSRj. The values of the

problem parameters were chosen in the following way.

The considered time horizons are 25, 50 and 75 periods. Demand was assumed to be normally distributed with mean 100 and standard deviation 50. Returns are also drawn from a normal distribution, with three different parameter settings, $(\mu = 10, \sigma = 5)$, $(\mu = 50, \sigma = 25)$ and $(\mu = 90, \sigma = 45)$. Negative demands and returns were rounded up to zero, thus creating a positive probability of having zero demand or returns. The coefficient of variation is kept constant (at $\frac{1}{2}$); previous research on lot-sizing problems (e.g. Trigeiro et al. (1989)) has indicated that varying this coefficient has little influence on the difficulty of a problem. Each of the 9 possible parameter settings is replicated 10 times, thus obtaining 90 demand-returns data sets.

All cost parameters are assumed time-invariant. Preliminary experiments showed that instances with non-stationary cost parameters were not harder to solve than their counterparts with time-invariant costs. The values of the set-up costs that are tested are 125, 250, 500 and 1000. In the ELSRs the set-up costs of the manufacturing and remanufacturing process are equal. The holding costs are 1 for all instances, for both serviceables and returns. Again, preliminary experiments showed that cases where serviceables and returns had different holding costs were not harder to solve. Production and remanufacturing costs were assumed to be zero.

We solved all problems with CPLEX 10.1 (single processor version) in the Aimms 3.9 modelling environment on a Windows XP based computer with a 3.0 GHz Intel Core 2 Duo processor (E8400) and 3.2 GB RAM. The time limit for each instance and formulation was one hour.

5.2 Results for the separate set-ups case

The results for the problem with separate set-ups can be found in Tables 1, 2 and 3. These tables give the number of instances (out of ten replications) that could be solved to optimality within the one hour time limit. They also give the average optimality gap of the MIPs, where the gap of a problem solved to optimality was counted as zero. If all instances were solved to optimality by all methods, then these rows were omitted. Furthermore, the average solution times of the MIPs are given; if an instance could not be solved to optimality within the time limit, the solution time was counted as one hour. The number of times the LP relaxation of a formulation found the integer optimal solution is also stated, unless none of the LP relaxations found any integer optimal solutions. Finally, the average LP gaps are mentioned, as a measure of the quality of the LP relaxation. We computed the LP gap as the percentage deviation of the solution of the LP relaxation with respect to the best integer solution found by

any of the formulations. The best performance among all formulations is indicated in boldface.

In general, we can see that the shortest path (SP) and partial shortest path (PSP2 and PSP3) reformulations have the best LP relaxations, in the sense that they have smaller LP gaps than the Original and (l, S, WW) formulations, in each ten-replication average. Furthermore, the LP relaxations of SP, PSP2 and PSP3 give the same solution for all instances but three, for which there was a negligible difference.

Table 1: Separate set-ups, 25 periods

set-up costs	average returns															
	10					50					90					
	Original	SP	PSP2	PSP3	(l, S, WW)	Original	SP	PSP2	PSP3	(l, S, WW)	Original	SP	PSP2	PSP3	(l, S, WW)	
125	avg. sol. time MIP (s)	0.1	0.1	0.1	0.1	0.2	0.3	0.4	0.5	0.7	0.4	0.2	1.1	1.3	1.5	0.5
	avg. LP gap (%)	85	0.99	0.99	0.99	20	82	5.9	5.9	5.9	19	47	9.6	9.6	9.6	13
250	avg. sol. time MIP (s)	0.4	0.1	0.1	0.2	0.5	1.1	0.5	0.9	0.9	1.1	0.6	1.2	2.2	2.5	1.1
	avg. LP gap (%)	82	0.88	0.88	0.88	17	81	5.5	5.5	5.5	16	56	9.0	9.0	9.0	12
500	avg. sol. time MIP (s)	0.5	0.1	0.1	0.2	0.6	1.5	0.5	0.9	0.9	1.2	0.5	0.8	1.6	1.6	1.0
	avg. LP gap (%)	77	0.85	0.85	0.85	14	79	4.2	4.2	4.2	14	63	7.7	7.7	7.7	11
1000	avg. sol. time MIP (s)	0.1	0.0	0.1	0.1	0.4	1.2	0.3	0.7	0.7	1.0	0.5	0.7	1.4	1.3	0.7
	integer solutions LP	0	2	2	2	0	0	0	0	0	0	0	0	0	0	0
	avg. LP gap (%)	72	0.15	0.15	0.15	10	75	3.6	3.6	3.6	11	65	6.1	6.1	6.1	9.1

Table 2: Separate set-ups, 50 periods

set-up costs	average returns															
	10					50					90					
	Original	SP	PSP2	PSP3	(l, S, WW)	Original	SP	PSP2	PSP3	(l, S, WW)	Original	SP	PSP2	PSP3	(l, S, WW)	
125	avg. sol. time MIP (s)	12	0.4	0.7	0.8	6	606	426	252	311	296	152	325	194	218	106
	avg. LP gap (%)	91	1.7	1.7	1.7	21	89	7.0	7.0	7.0	20	45	7.3	7.3	7.3	11
250	solved to optimality	3	10	10	10	10	3	10	10	9	7	8	10	10	10	9
	avg. MIP gap (%)	1.7	0	0	0	0	1.7	0	0	0.03	0.5	0.4	0	0	0	0.1
	avg. sol. time MIP (s)	3272	0.5	1.1	1.1	369	2859	843	989	1090	2000	1122	811	805	920	1006
	avg. LP gap (%)	89	1.0	1.0	1.0	18	88	6.3	6.3	6.3	17	56	7.8	7.8	7.8	11
500	solved to optimality	0	10	10	10	10	3	10	10	10	10	6	9	9	9	9
	avg. MIP gap (%)	5.1	0	0	0	0	1.4	0	0	0	0	0.99	0.21	0.18	0.24	0.20
	avg. sol. time MIP (s)	3600	0.5	1.3	1.3	335	3144	53	91	100	729	1576	390	450	517	607
	avg. LP gap (%)	86	1.1	1.1	1.1	18	86	4.7	4.7	4.7	14	64	7.7	7.7	7.7	11
1000	solved to optimality	0	10	10	10	10	1	10	10	10	10	9	10	9	9	9
	avg. MIP gap (%)	3.6	0	0	0	0	3.4	0	0	0	0	0.48	0	0.25	0.13	0.19
	avg. sol. time MIP (s)	3600	0.4	1.0	1.1	401	3582	25	45	58	283	724	424	592	610	517
	avg. LP gap (%)	83	0.67	0.67	0.67	14	83	3.8	3.8	3.8	12	69	6.2	6.2	6.2	9.3

When we look at performance in terms of optimality gap and computation time, we see that the shortest path reformulation gives the best results in most cases. Notice that PSP2 (with $k^s = \lceil 2 \cdot TBO^s \rceil$ and $k^r = \lceil 2 \cdot TBO^r \rceil$) gives better results than PSP3 (with $k^s = \lceil 3 \cdot TBO^s \rceil$ and $k^r = \lceil 3 \cdot TBO^r \rceil$) in almost all cases, which could be explained from the fact that both formulations have the same LP relaxation (in all but three tested

Table 3: Separate set-ups, 75 periods

set-up costs		average returns														
		10					50					90				
		Original	SP	PSP2	PSP3	(<i>l, S, WW</i>)	Original	SP	PSP2	PSP3	(<i>l, S, WW</i>)	Original	SP	PSP2	PSP3	(<i>l, S, WW</i>)
125	solved to optimality	9	10	10	10	10	0	0	1	0	0	3	2	2	3	3
	avg. MIP gap (%)	0.17	0	0	0	0	2.9	2.5	1.5	1.7	1.7	1.7	2.4	1.9	2.0	1.3
	avg. sol. time MIP (s)	985	1.4	1.4	1.6	176	3600	3600	3504	3600	3600	2855	3268	3255	3277	2993
	avg. LP gap (%)	94	1.3	1.3	1.3	21	88	7.5	7.5	7.5	20	46	7.7	7.7	7.7	11
250	solved to optimality	0	10	10	10	0	0	0	0	1	0	2	2	1	1	2
	avg. MIP gap (%)	8.0	0	0	0	1.4	5.5	2.1	1.9	2.0	3.1	3.1	2.4	2.5	2.4	2.2
	avg. sol. time MIP (s)	3600	3.2	4.6	5.0	3600	3600	3600	3600	3596	3600	3231	3151	3263	3280	3247
	avg. LP gap (%)	92	1.2	1.2	1.2	18	89	6.3	6.3	6.3	17	56	7.9	7.9	7.9	11
500	solved to optimality	0	10	10	10	0	0	3	2	2	0	0	1	1	1	1
	avg. MIP gap (%)	12.2	0	0	0	3.0	7.1	1.1	1.2	1.3	3.3	4.7	2.8	2.7	3.0	2.9
	avg. sol. time MIP (s)	3600	2.6	4.6	5.3	3600	3600	2982	3034	2671	3600	3600	3555	3528	3472	3364
	avg. LP gap (%)	90	0.97	0.97	0.97	16	89	5.0	5.0	5.0	15	65	8.1	8.1	8.1	11
1000	solved to optimality	0	10	10	10	0	0	6	5	5	0	0	3	2	2	2
	avg. MIP gap (%)	14.9	0	0	0	3.7	10.5	0.4	0.4	0.5	2.9	5.0	2.0	2.2	2.4	2.4
	avg. sol. time MIP (s)	3600	1.4	3.7	4.3	3600	3600	2239	2518	2765	3600	3600	2985	3309	3249	3340
	avg. LP gap (%)	88	0.74	0.74	0.74	15	87	4.3	4.3	4.3	13	71	6.5	6.5	6.5	9.3

instances), but PSP2 has fewer variables. We also did some experiments with other choices for k^r and k^s , but this did not lead to improvements in the performance.

PSP2 outperforms SP if the number of periods is large (50 or 75) and the set-up costs are relatively low. It is not surprising that the partial shortest path reformulation has an advantage over the full reformulation (SP) under these circumstances, because relatively low set-up costs imply a small time between orders. In combination with a large horizon, this means that PSP2 has much fewer variables than SP. Of course, one may wonder why PSP2 does not always perform better than SP, since their LP relaxations give the same solution. In some problem instances, the time between orders may be large compared to the horizon, in which case there is little gain in using an approximate reformulation, because it will contain nearly all of the flow (z) variables and have several additional variables (the u , v , w variables). Otherwise, the difference in performance between PSP2 and SP may be attributable to the CPLEX solver, which may choose a different cutting (and/or branching) strategy, for instance because it might not recognize PSP2 as a shortest path formulation.

The performance of the shortest path type reformulations (SP, PSP2 and PSP3) is best when the return rate is low. This is not surprising, because if there are no returns at all, then we know that the LP relaxation of SP always gives the optimal (integer) solution.

The (l, S, WW) formulation provides the smallest MIP gaps and computation times if the return rate is high, the set-up costs are low and the horizon is not short (50 or 75 periods). The Original formulation only gives the fastest results for some of the simplest instances, with only 25 periods and low set-up costs. It should be noted that

the performance of both the Original and (l, S, WW) formulation can go down quite dramatically when the set-up costs are higher. When there are 50 periods for example, Original solves all 30 instances in $4\frac{1}{2}$ minutes on average if the set-up costs are 125, but if the set-up costs are 1000, Original can solve only 10 out of 30 instances within the one hour time limit.

5.3 Results for the joint set-ups case

Tables 4 and 5 present the results for the problem with joint set-ups. All formulations of all instances with a horizon of 25 periods were solved by CPLEX within a quarter of a second. Those results are therefore omitted.

When we compare the results for ELSRj with those for ELSRs, we see that ELSRj is easier to solve than ELSRs. This was to be expected, because the problem with separate set-ups has twice as many integer variables as the problem with separate set-ups. In fact, formulation SP was able to solve all instances of ELSRj within a reasonable amount of time, which was the reason why we did not test a partial shortest path reformulation for ELSRj.

Table 4: Joint set-ups, 50 periods

set-up costs		average returns								
		10			50			90		
		Original	SP	(l, S, WW)	Original	SP	(l, S, WW)	Original	SP	(l, S, WW)
125	avg. sol. time MIP (s)	0.0	0.0	0.2	0.0	0.1	0.3	1.4	4.9	3.9
	integer solutions LP	0	9	1	0	0	0	0	0	0
	avg. LP gap (%)	89	0.009	0.91	85	1.0	1.4	42	3.3	3.5
250	avg. sol. time MIP (s)	0.8	0.0	0.2	0.1	0.1	0.4	2.1	4.8	3.9
	integer solutions LP	0	10	4	0	1	0	0	0	0
	avg. LP gap (%)	87	0	0.70	85	0.48	0.97	53	3.5	4.0
500	avg. sol. time MIP (s)	35	0.0	0.2	1.9	0.0	0.3	3.4	3.2	3.7
	integer solutions LP	0	10	6	0	6	3	0	0	0
	avg. LP gap (%)	84	0	0.68	83	0.11	0.47	61	3.1	3.7
1000	avg. sol. time MIP (s)	168	0.0	0.2	16.8	0.0	0.2	6.2	0.9	2.4
	integer solutions LP	0	10	6	0	7	5	0	0	0
	avg. LP gap (%)	80	0	0.45	81	0.009	0.16	66	2.0	2.4

The results for joint set-ups show roughly the same pattern as for the separate set-ups case. The shortest path formulation has the best LP relaxation in terms of LP gaps, compared to the Original and (l, S, WW) formulations. Moreover, the optimal solution of the LP relaxation of SP is often integer. When the average returns are low (10), it even finds an integral optimum in 79 out of 80 test instances. The LP relaxation of (l, S, WW) also finds integer solutions, although not as often as SP. The LP relaxation of SP does worsen when the average returns are higher, but the average LP gap is always smaller than for the LP relaxations of Original and (l, S, WW) .

Table 5: Joint set-ups, 75 periods

set-up costs		average returns								
		10			50			90		
		Original	SP	(l, S, WW)	Original	SP	(l, S, WW)	Original	SP	(l, S, WW)
125	avg. sol. time MIP (s)	0.1	0.0	0.8	0.1	0.2	1.3	36	69	38
	integer solutions LP	0	10	2	0	0	0	0	0	0
	avg. LP gap (%)	74	0	0.48	85	1.0	1.5	42	3.4	3.5
250	avg. sol. time MIP (s)	40	0.1	0.8	0.4	0.2	1.3	48	57	54
	integer solutions LP	0	10	6	0	0	0	0	0	0
	avg. LP gap (%)	91	0	0.32	87	0.42	0.83	53	3.3	3.6
500	solved to optimality	0	10	10	10	10	10	9	10	10
	avg. MIP gap (%)	2.1	0	0	0	0	0	0.05	0	0
	avg. sol. time MIP (s)	3600	0.1	0.8	58	0.2	1.4	678	248	136
	integer solutions LP	0	10	5	0	0	0	0	0	0
	avg. LP gap (%)	88	0	0.20	86	0.22	0.46	63	3.7	4.0
1000	solved to optimality	0	10	10	5	10	10	8	10	10
	avg. MIP gap (%)	9.4	0	0	0.6	0	0	0.2	0	0
	avg. sol. time MIP (s)	3600	0.1	0.8	2934	0.1	1.1	1686	61	38
	integer solutions LP	0	10	8	0	5	3	0	0	0
	avg. LP gap (%)	85	0	0.04	85	0.08	0.16	69	2.3	2.5

Looking at the computation times of the MIPs, we see again that the shortest path reformulation gives the fastest results in most cases. If the average returns are higher, however, (l, S, WW) often has shorter computation times when the horizon is long (75 periods) and Original has shorter computation times for 50 periods and low set-up costs. The original formulation also gives slightly faster results in a few other cases with low set-up costs. However, if the set-up costs grow, then the performance of Original goes down, similar to what we have seen in the separate set-up case. This is especially clear when the number of returns is low. For example: when there are 75 periods, Original solves all instances within 0.1 second if the set-up costs are 125, but if set-up costs are 1000, none of the problems can be solved to optimality within one hour and the average MIP gap is 9.4%, while SP can solve all instances within 0.1 second.

6 Conclusion and further research

In this paper, we have considered two variants of the economic lot-sizing problem with remanufacturing. As we have shown, both the problem with joint and with separate set-up costs for the production and remanufacturing process are \mathcal{NP} -hard. We have proposed several MIP formulations of these problems and tested their efficiency on a wide variety of test instances and found that, for both problem variants, SP (our shortest path formulation) performs better than the Original and (l, S, WW) formulations, especially in terms of the quality of the LP relaxation. The computation times and MIP gaps are also smaller in the vast majority of test instances. When the return rate is high

though, faster results may be obtained by (l, S, WW) (for a large horizon) or Original (for a shorter horizon). A partial shortest path formulation (PSP2) exhibits many features of SP, such as the quality of the LP relaxation, while having fewer variables and needing less computer memory.

It would be worthwhile to see what the consequences are if the test problems were solved with another solver than CPLEX (that exploits the problem structure in a different way than CPLEX does) and see to what extent the differences in performance between SP and PSP2 persist. Other avenues for further research include extending the shortest path reformulations with production capacities, which should be quite straightforward, since Eppen and Martin (1987) introduced their shortest path reformulation of the lot-sizing problem without remanufacturing in the context of production capacities. Another extension involves changing the assumption that remanufactured products are as good as new to a situation with a separate demand for new and remanufactured products, where new products can serve as substitutes for remanufactured ones. A similar setting was studied by Piñeyro and Viera (forthcoming), who solved the problem with tabu search. Formulations similar to the ones presented in our paper could be used to solve this extended problem to optimality. Another track worth exploring is using the solution of the LP relaxation of SP in a heuristic, e.g. a rounding or relax-and-fix heuristic. Since this formulation gives good results for ELSRs and especially ELSR_j, we would expect such a heuristic to give good feasible solutions in a short amount of time.

References

- J. L. Beltrán and D. Krass. Dynamic lot sizing with returning items and disposals. *IIE Transactions*, 34(5):437–448, 2002.
- R. Dekker, M. Fleischmann, K. Inderfurth, and L. N. van Wassenhove. *Reverse Logistics: Quantitative Models for Closed-Loop Supply Chains*. Springer-Verlag, Berlin Heidelberg, 2004.
- G. D. Eppen and R. K. Martin. Solving multi-item capacitated lot-sizing problems using variable redefinition. *Operations Research*, 35(6):832–848, 1987.
- M. Florian, J. K. Lenstra, and A. H. G. Rinnooy Kan. Deterministic production planning: algorithms and complexity. *Management Science*, 26:669–679, 1980.
- M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W.H. Freeman and Company, New York, 1979.

- B. Golany, J. Yang, and G. Yu. Economic lot-sizing with remanufacturing options. *IIE Transactions*, 33:995–1003, 2001.
- Z. Pan, J. Tang, and O. Liu. Capacitated dynamic lot sizing problems in closed-loop supply chain. *European Journal of Operational Research*, 198:810–821, 2009.
- P. Piñeyro and O. Viera. Inventory policies for the economic lot-sizing problem with remanufacturing and final disposal options. *Journal of Industrial and Management Optimization*, 5:217–238, 2009.
- P. Piñeyro and O. Viera. The economic lot-sizing problem with remanufacturing and one-way substitution. *International Journal of Production Economics*, forthcoming. doi:10.1016/j.ijpe.2010.01.007.
- Y. Pochet and L.A. Wolsey. Polyhedra for lot-sizing with wagner-whitin costs. *Mathematical Programming*, 67:297–324, 1994.
- Y. Pochet and L.A. Wolsey. *Production Planning by Mixed Integer Programming*. Springer, 2006.
- K. Richter and M. Sombrutzki. Remanufacturing planning for the reverse Wagner/Whitin models. *European Journal of Operational Research*, 121:304–315, 2000.
- K. Richter and J. Weber. The reverse Wagner/Whitin model with variable manufacturing and remanufacturing cost. *International Journal of Production Economics*, 71:447–456, 2001.
- T. Schulz. A new silver-meal based heuristic for the single-item dynamic lot sizing problem with returns and remanufacturing. FEMM Working Paper 38, Otto-von-Guericke-University Magdeburg, November 2009.
- E. A. Silver, D. F. Pyke, and R. Petersen. *Inventory management and production planning and scheduling*. Wiley, New York, 3rd edition, 1998.
- H. Stadtler. Reformulations of the shortest route model for dynamic multi-item multi-level capacitated lotsizing. *OR Spektrum*, 19:87–96, 1997.
- R. H. Teunter, Z. P. Bayındır, and W. van den Heuvel. Dynamic lot sizing with product returns and remanufacturing. *International Journal of Production Research*, 44(20):4377–4400, 2006.
- W. W. Trigeiro, L. J. Thomas, and J. O. McClain. Capacitated lot sizing with setup times. *Management Science*, 35(3):353–366, 1989.

- W. van den Heuvel. *The economic lot-sizing problem: New results and extensions*. PhD thesis, Erasmus University Rotterdam, 2006.
- E.A. van der Laan and R.H. Teunter. Simple heuristics for push and pull remanufacturing policies. *European Journal of Operational Research*, 175:1084–1102, 2006.
- M. Van Vyve and L.A. Wolsey. Approximate extended formulations. *Mathematical Programming, Series B*, 105:501–522, 2006.
- J. Yang, B. Golany, and G. Yu. A concave-cost production planning problem with remanufacturing options. *Naval Research Logistics*, 52(5):443–458, 2005.