Production Lot Sizing and Scheduling with Non-Triangular Sequence-Dependent Setup Times

Alistair R. Clark
Alistair.Clark@uwe.ac.uk

Masoumeh Mahdieh
Masoumeh.Mahdieh@uwe.ac.uk

Department of Engineering Design and Mathematics
University of the West of England, Bristol, BS16 1QY, Reino Unido.

Abstract

This paper considers a production lot sizing and scheduling problem with sequence-dependent setup times that are not triangular. Consider, for example, a product p that contaminates some other product r unless either a decontamination occurs as part of a substantial setup time st_{pr} or there is a third product q that can absorb p's contamination. When setup times are triangular then $st_{pr} \leq st_{pq} + st_{qr}$ and there is always an optimal lot sequence with at most one lot per product per period (AM1L). However, product q's ability to absorb p's contamination presents a shortcut opportunity and could result in shorter non-triangular setup times such that $st_{pr} > st_{pq} + st_{qr}$. This implies that it can sometimes be optimal for a shortcut product such as q to be produced in more than one lot within the same period, breaking the AM1L assumption in much research. This paper formulates and explains a new optimal model that not only permits multiple setups and lots per product in a period (ML), but also prohibits subtours using a polynomial number of constraints rather than an exponential number. Computational tests demonstrate the effectiveness of the ML model, even in the presence of just one decontaminating shortcut product, and its fast speed of solution compared to the equivalent AM1L model.

Key Words: Lot sizing and scheduling, Sequence-dependent setup times, Non-triangular setup times.

1 Introduction

Some manufacturing systems have to meet a regular but varying demand for products. When manufacturing capacity is limited, such demand cannot be met instantaneously from production, but from inventory accumulated previously. Lot-sizing decisions then need to be made about how much of each product to produce in each demand period and how much inventory to accumulate in order to meet demand while keeping within production capacity.

If a setup cost or time is charged to change from one product to another, then a sequence or schedule of lots also needs to be decided. If such setups are sequence-dependent (i.e., the size of the setup charge depends on the product processed immediately beforehand), as illustrated in Figure 1, then the decisions become more complex.

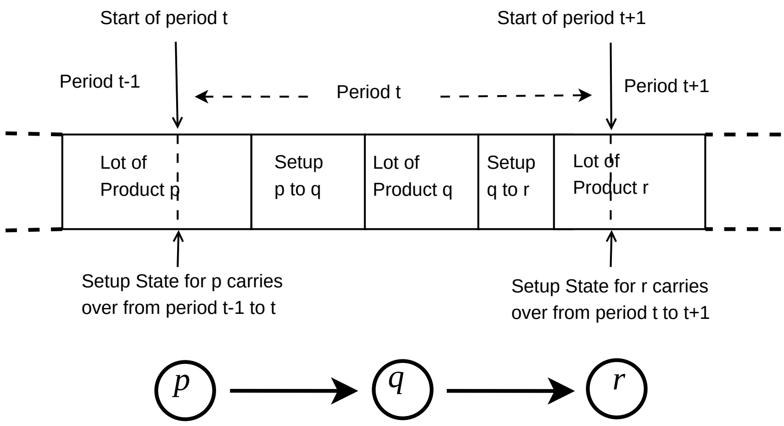

Figure 1: Production and inventory to meet demand

Many manufacturers separate lot sizing decisions from lot sequencing in order to simplify the complexity of the decision-making. However, this can result in production being less effective and more costly than it needs to be. To competitively satisfy the demand for products within available production capacity, the lot sizing and sequencing decisions should be handled simultaneously.

2 Lot Sizing and Sequencing

Research into production lot sizing and scheduling has progressed substantially over the last decades, as shown in the reviews by Drexl and Kimms (1997) and Karimi et al. (2003), recent research (Kovács et al.; 2009), and a forthcoming special issue (Clark et al.; 2011). In July 2010 at the *24th European Conference on Operational Research* (EURO10) in Lisbon, a stream on lot sizing and scheduling was organized for the first time in the history of this conference, containing seven sessions with more than 25 presentations.

In particular, much progress has been made in the area of lot sequencing when setup times are sequence -dependent (Meyr; 2000; Clark and Clark; 2000; Araújo et al.; 2007). The General Lot-sizing and Scheduling Problem (GLSP), developed by Fleischmann and Meyr (1997), minimises inventory and sequence-dependent setup costs on a single machine with finite capacity, allowing multiple setups in each single 'large-bucket' time period. The GLSP was extended by Meyr (2000) to consider sequence-dependent setup times (GLSP-ST). Toso, Morabito & Clark (2007) reformulated the GLSP-ST model to permit backlogging and non-triangular setup times, but still assumed at most one lot per product in each period.

Clark, Morabito & Toso (2006) pursued an alternative approach via the Asymmetric Travelling Salesman Problem (ATSP), which has been very extensively researched (Lawler et al 1985; Carpaneto et al 1995). The adaptation of the ATSP to modelling lot-sizing and scheduling with sequence-dependent setups is not direct, since the production system is often already setup for a

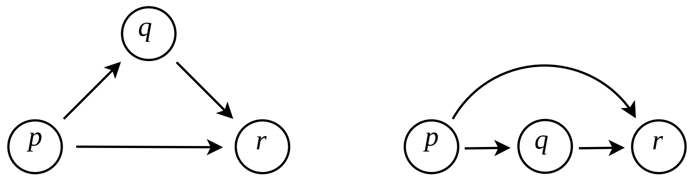

Figure 2: Triangular and Non-Triangular Setups

particular product (i.e. starting at a given city) and some products might not be produced in a given period if the demand is sufficiently small or the capacity tight (Clark et al 2006).

Clark et al. (2010) extended a method that has been found to be successful in practice for optimally solving the ATSP, namely, to quickly solve the corresponding Assignment Problem (AP) as a linear programme, identify the resulting subtours, and then resolve the AP, explicitly prohibiting these subtours using a potentially exponential number of Dantzig-Fulkerson-Johnson-type constraints adapted from Dantzig et al. (1954). The method carries on iteratively in this manner until no subtours result. It can be used heuristically (and its convergence rate sometimes accelerated) by patching the subtours into a single tour at each iteration (Karp 1979), thus providing a feasible solution (and an upper bound). Clark et al (2006) adapted the subtour elimination method to lot sequencing over multiple periods with setup carryover between periods. An extension of the method then used the patching heuristic to accelerate the time to converge to a provably optimal solution.

3 Non-triangular setup times

In some industries, e.g., animal feed supplements, some products can contaminate other products, e.g., copper is essential for pigs but kills sheep even in tiny doses. Contamination is a particular concern for the feed industry, although the problem is general and similar concerns also exist in a diverse range of other industries, such as food & beverages, and the oil industry. In the feed industry, blending equipment must be cleaned in order to avoid contamination, resulting in substantial setups that consume scarce production time. Fortunately, the amount of cleaning can be minimised by the effective sequencing of production lots.

Certain intermediate "cleansing" or shortcut products can cause non-triangular setup times. These products clean the machines whilst being processed (e.g., certain wheat mixtures) and hence reduce overall setup times. In other words, contamination cleaning can occur during value-adding production time as well as during non-productive setup time.

More precisely, "triangular" setup times occur when it is never worse to setup from product p to r directly than to setup via a third product q, so that triangular inequality $s(p,r) \leq s(p,q)$ + $s(q,r)$ always holds (as shown on the left side of Figure 2). However, in the animal feed and other industries, the contamination of a product r by a previous product p just beforehand can be often avoided by producing enough of an intermediate product q so that it absorbs p's contamination. For this to save time, the triangular inequality must not hold in this case, ie, the sum of the setup times $s(p,q)$ from p and $s(q,r)$ to r must be short enough so that $s(p,q) + s(q,r) < s(p,r)$ (as shown on the right side of Figure 2).

Existing mathematical models can be used when setup times are triangular, e.g., Meyr (2000) and Clark et al. (2010) where minimum lot-sizes are imposed to allow proper cleaning of p's contaminants, i.e., to avoid a setup from p to r via zero production of q rather than directly. However, the disobeying of the triangular inequality means that it could be optimal in certain circumstances for an intermediate shortcut product q to be produced in more than one lot within the same time-period, as shown in Figure 3. Thus the assumption of existing models (Meyr; 2000; Clark et al.; 2010) of at most one lot per product per period would not hold in such a situation. The breaking of this assumption is the key feature of the model developed below in section 4.

The GLSP models of Fleischmann and Meyr (1997) and Meyr (2000) allow non-triangular setups, as in Toso et al. (2009), but the ATSP-based model of Clark et al. (2010) assumes one lot per product per periods and so cannot allow multiple lots of shortcut products per period, as required to take advantage of non-triangular setup times. A sequence with multiple lots per period for some

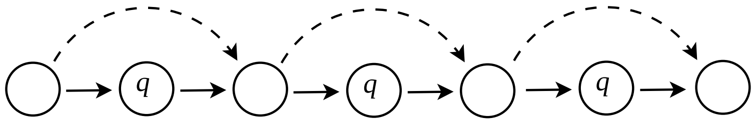

Figure 3: A Sequence of Non-Triangular Setups via a Shortcut Product q

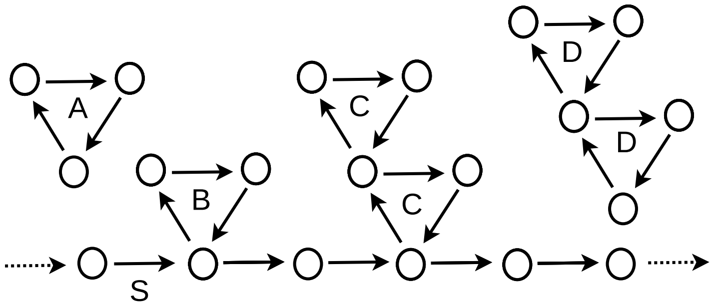

Figure 4: A main sequence and different types of subtour

products could look like that illustrated in Figure 4. Subtours connected to the main sequence S by shortcut products are possible (e.g., subtours B and C in Fig. 4).

Thus an exact formulation must allow connected subtours but exclude disconnected subtours (e.g., subtours A and D in Fig. 4). Menezes et al. (2010) developed such a formulation using an iterative model and method based on a potentially exponential number of Miller-Tucker-Zemlin subtour elimination constraints (Miller et al.; 1960).

This paper formulates and tests a new exact model for lot sizing and sequencing with non-triangular setups. The model is developed in section 4 using a polynomial number of multi-commodity-flow-type constraints adapted from Claus (1984), and then computationally tested in section 5. The model is generalised in section 6 to include period-overlapping setup operations and again tested computationally. The paper concludes in section 7 with a discussion of the model's value and flags remaining challenges and opportunities for future research.

4 Modelling multiple lots per product per period

The following indices are used:

p, q, r Product families, from $\{1,...,P\}$ where P = the number of families.

t Time period, from $\{1,...,T\}$ where T = the number of periods (e.g., days or weeks) in the scheduling horizon.

The input data required by the model are:

Cap_t Available capacity time in each period t.

u_p Time needed to produce one batch of each product p.

ml_p Minimum lot size of product p.

h_p Inventory holding cost per period for product p.

g_p Backlog cost per period for product p.

co_t Unit cost of machine time in period t.

st_{pq} Setup time needed to changeover from product p to product q.

d_{pt} Forecast of demand for product p at the end of period t.

I_{p0} Inventory of product p at the start of the scheduling horizon.

p^* The product already setup when period 1 starts (the initial setup state).

The decisions output by the model are:

I^+_{pt} Inventory of product p at the end of period t, non-negative.

I^-_{pt} Backlogs of product p at the end of period t, non-negative.

x_{pt} Total size of all lots of product product p in period t (an integer number of batches).

y_{pqt} Number of times that production is to be changed over from product p to product q in period t.

z_{pt} Number of times that product p is in a setup state in period t.

α_{pt} = 1 if p is the product already setup when period t starts (the setup state), = 0 otherwise. Thus the model allows the setup state at the start of a period to be carried over from the previous period. Note that $t \in \{1, ..., T+1\}$ and that $\alpha_{p^*,1} = 1$.

$slack_t$ Number of hours of slack capacity in period t.

The objective function (1) minimizes primarily backlogs via heavy penalties, then the costs of inventory, while maximizing slack capacity (if backlogs and inventory are readily zeroed by an excess of capacity):

$$\text{Minimise} \quad \sum_{p,t} \left(h_p\, I^+_{pt} \;+\; g_p\, I^-_{pt}\right) \;-\; \sum_{t} co_t\; slack_t \;+\; 0.01 \sum_{p,t} z_{pt} \tag{1}$$

Unnecessary capacity-eating setups are prevented by maximizing slack capacity in (1). The last term $[0.01 \sum_{p,t} z_{pt}]$ is simply a mathematical device to eliminate any excessive zero-time setups. The value of the coefficient 0.01 may need adjusting depending on the values of the other terms in (1).

Constraints (2) balance inventory, backlogs, production and demand over consecutive weeks, as previously shown in Figure **??**:

$$I^+_{p,t-1} \;-\; I^-_{p,t-1} \;+\; x_{pt} \;-\; d_{pt} \;=\; I^+_{pt} \;-\; I^-_{pt} \quad \forall\, p,\; t \tag{2}$$

The capacity constraints (3) take into account setup and production times, and calculate any capacity slack:

$$\sum_{p} u_p\, x_{pt} \;+\; \sum_{p,q} st_{pq}\; y_{pqt} \;+\; slack_t \;=\; Cap_t \quad \forall\, t \tag{3}$$

Constraints (4) ensure that a product can be produced in a period only if the machine is setup for it at some time in period t:

$$x_{pt} \;\leq\; \left(\min \left\{ \frac{Cap_t}{u_p}, \sum_{\tau=1}^{T} d_{p\tau} - I^+_{p0} + I^-_{p0} \right\} \right) z_{pt} \quad \forall\, p,\; t \tag{4}$$

The coefficient of z_{pt} in (4) is an upper bound on the value of x_{pt}, calculated as the minimum of (a) the amount of product p that can be produced if period t were entirely dedicated to its production, and (b) the effective demand for product p over all periods $t = 1, ..., T$ (given that backlogs of demand may have to be produced as well as current and future demand).

Note that constraint (4) is valid, but loose as z_{pt} need only be 1, not ≥ 2. Constraint (19) will tighten and replace (4) when extra binary variables for subtour-elimination are introduced below.

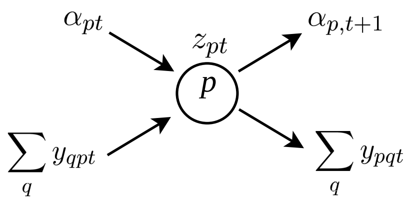

Figure 5: Node flow modelled by constraints (8) and (9)

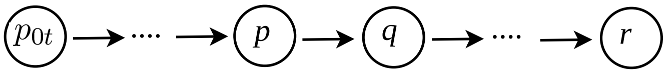

Figure 6: A path from carryover product $p_{0t} = p|(\alpha_{pt} = 1)$ to product r

Constraints (5) impose a minimum lot size of ml_q on each of product q's z_{qt} lots in order to force proper cleaning of a previous product p's contaminants:

$$x_{qt} \ \geq \ ml_q z_{qt} \quad \forall\, t,\ q \tag{5}$$

Without constraints (5), a model solution could incorrectly schedule a setup from p to a third product r via zero production of q rather than directly. Note that the total lot size x_{qt} can be split into separate z_{qt} lots, each of which is at least ml_q units in size.

Constraints (6) prohibit setups between the same product:

$$y_{ppt} \ = \ 0 \quad \forall\, p,\ t \tag{6}$$

Constraints (7) ensure that there is exactly one product in a setup state between each period:

$$\sum_p \alpha_{pt} \ = 1 \quad \text{for } t = 2, ..., T+1 \tag{7}$$

We have left until last the consideration of the ATSP-related constraints for sequencing products. Constraints (8) and (9) relate the α_{pt} and z_{pt} setup state variables to the y_{pqt} changeover variables, to and from a product respectively (Figure 5).

$$\alpha_{pt} \ + \ \sum_q y_{qpt} \ = \ z_{pt} \quad \forall\, p,\ t \tag{8}$$

$$\sum_q y_{pqt} \ + \ \alpha_{p,t+1} \ = \ z_{pt} \quad \forall\, p,\ t \tag{9}$$

The initial optimal solution to the model specified by expressions (1) to (9) will consist of a single sequence starting with product $p|\{\alpha_{pt} = 1\}$ and ending with $p|\{\alpha_{p,t+1} = 1\}$ (possibly with embedded connected subtours), and maybe one or more disconnected subtours, as previously illustrated in Figure 4. Recall that subtours connected to the main sequence S are permitted (e.g., subtours B and C in Fig. 4), but disconnected subtours must be prohibited (e.g., subtours A and D in Fig. 4).

The paper by Öncan et al. (2009) reviews and analytically compares of many ATSP formulations, highlighting the tightness of the *multi-commodity-flow* (MCF) formulation by Claus (1984) which is the inspiration for the formulation that prohibits diconnnected subtours *a priori*.

First define additional binary decision variables a^r_{pqt} as follows. Let $a^r_{pqt} = 1$ if the arc $p \rightarrow q$ is on a path from carryover product $p_{0t} = p|(\alpha_{pt} = 1)$ to product r within period t's sequence of lots, otherwise = 0, as shown in Figure 6.

The arc $p \rightarrow q$ must be part of a solution, so that value of a^r_{pqt} is constrained as follows:

$$a^r_{pqt} \ \leq \ y_{pqt} \quad \forall\, p,\ q,\ r,\ t \tag{10}$$

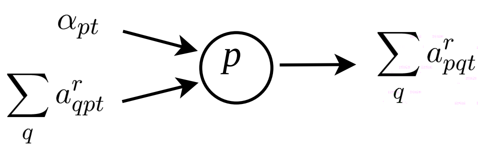

Figure 7: The path from p_{0t} to r traverses only those products p for which $z^{01}_{pt} = 1$

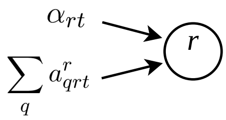

Figure 8: The path from p_{0t} must reach product r

To use a^r_{pqt} to prohibit subtours, a further binary decision variables z^{01}_{pt} are needed. Let z^{01}_{pt} =1 if product p is ever in a setup state in period t, otherwise 0. Note that $z^{01}_{pt} = 1 \Leftrightarrow z_{pt} \geq 1$ and that $z^{01}_{pt} = 0 \Leftrightarrow z_{pt} = 0$. This is enforced by the following constraints:

$$z_{pt} \;\geq\; z^{01}_{pt} \quad \forall\, p,\; t \tag{11}$$

$$z_{pt} \;\leq\; ZUB_p z^{01}_{pt} \quad \forall\, p,\; t \tag{12}$$

where ZUB_p is a prespecified upper bound (UB) on the value of z_{pt}, calculated in the computational tests below as the size of the ordered set $\{(i,j)|st_{ij} \geq st_{ip} + st_{pj}\}$, which will often be 1 for known non-shortcut products.

The three sets of constraints (14, 16 and 18) below will then allow connected subtours, but prohibit disconnected ones *a priori*.

Constraint (13) requires that the period t path specified by the variables $\{a^r_{pqt} \mid \forall\; p,q\}$ starts at carryover product $p_{0t} = p|(\alpha_{pt} = 1)$ and then traverses further products in the sequence, as illustrated in Figure 7.

$$\alpha_{pt} + \sum_q a^r_{qpt} \;=\; \sum_q a^r_{pqt} \qquad \forall\, r, p \neq r, t \tag{13}$$

However, constraint (13) should be enforced only when $z^{01}_{pt} = z^{01}_{rt} = 1$, but not when either is zero, i.e., only when the setup states for p and r both occur during period t. Thus constraint (14) replaces (13):

$$\alpha_{pt} + \sum_q a^r_{qpt} + 2 - z^{01}_{pt} - z^{01}_{rt} \;\geq\; \sum_q a^r_{pqt} \quad \forall\, r, p \neq r, t \tag{14}$$

Constraint (15) ensures that the period t path specified by the variables $\{a^r_{qrt} \mid \forall\; p,q\}$ reaches product r (Figure 8):

$$\alpha_{rt} + \sum_q a^r_{qrt} \;=\; 1 \qquad \forall\, r, t \tag{15}$$

But constraint (15) should be imposed only when the setup state is configured for r at least once during period t (i.e., only when $z^{01}_{rt} = 1$), but not when the setup state is never configured for r during period t, (i.e., when $z^{01}_{rt} = 0$). Thus constraint (16) replaces (15):

$$\alpha_{rt} + \sum_q a^r_{qrt} \;=\; z^{01}_{rt} \quad \forall\, r, t \tag{16}$$

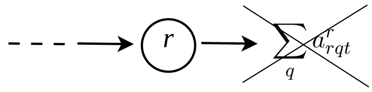

Figure 9: The path from p_{0t} must stop at product r

Constraint (17) requires that the a^r_{qrt} path from p_{0t} stops at product r (Figure 9):

$$\sum_q a^r_{rqt} = 0 \qquad \forall\, r, t \tag{17}$$

Again, this requirement can only be imposed when $z^{01}_{rt} = 1$ and never when $z^{01}_{rt} = 0$, so that constraint (18) replaces (17):

$$\sum_q a^r_{rqt} \;\leq\; 1 - z^{01}_{rt} \quad \forall\, r, t \tag{18}$$

Lastly, recall that constraint (4) is valid but too loose: z_{pt} need only be 1, and not ≥ 2. Constraint (4) can thus be tightened by replacing z_{pt} by z^{01}_{pt}:

$$x_{pt} \;\leq\; \left(\min \left\{ \frac{Cap_t \;+\; mo_t}{u_p}, \sum_{\tau=1}^{T} d_{p\tau} - I^+_{p0} + I^-_{p0} \right\} \right) z^{01}_{pt} \quad \forall\, p,\; t \tag{19}$$

Thus our model, denoted ML, for lot sizing and sequencing with non-triangular setup times and setup-state carryover between periods is specified by expressions (1-3, 5-12, 14, 16, 18, 19).

5 Computational Tests

Many models in the literature assume that there will be at most one lot per product per period. What are the pros and cons of this assumption? On the one hand, the model will be smaller with fewer variables and constraints, so we should expect faster solution times. On the other hand, the solutions with multiple lots per product per period will be excluded, so we will expect worse solutions in some cases. The computational tests in this section investigate this trade-off.

To do so, the *Multiple-lots*-per-product-per-period (ML) model can be simplified to assume that there will be at most one lot per product per period (AM1L) by merging z_{pt} and z^{01}_{pt} to be a binary variable z_{pt}. Thus constraints (11) and (12) disappear, and constraints (4) and (19) are now identical.

The aim of the tests was to assess how effectively the ML model took advantage of shortcut products to reduce the total time spent on setups, compared to the equivalent AM1L model. The tests also evaluated the consequences of less setup time on reducing demand backlogs (in the case of tight production capacity) or increasing the spare capacity (in the case of loose capacity), as well as the computing time of both models. The ML and AM1L models were both implemented in the AMPL modelling language (Fourer et al.; 2003) and solved using the Gurobi optimizer v4.0.1 (64-bit) (Gurobi Optimization Inc.; 2010) under Windows 7 on an Intel Core i5 CPU M460 at 2.53 GHz with 4Gb of RAM.

To obtain initial insights, the performance of both models was first compared on a system with $P = 10$ products whose lot sizes and sequences were to be scheduled over a horizon of $T = 4$ demand periods. The following data were used $I_{p0} = 0.0$, $Cap_t = 100.0$, $u_p = 0.4$, $ml_p = 1.0$, $h_p = 10.0$, $co_t = 1.0$, $p^* =$ product `P1` (arbitrarily), $\forall\, p$ and t. The setup times were intially set to be $st_{pq} = (q - p)$ where $p, q \in \{1...10\}$, so that product `P2` would normally be setup immediately after `P1`. However, `P5` was then made an extreme shortcut product with zero setup times: $st_{5q} = st_{p5} = 0$. The periodic demand forecasts d_{pt} varied over product p and period t to provoke non-uniform lot-sizes and avoid lot-for-lot production. They were then randomly varied by $\pm 50\%$ within the 25 runs of each statistical experiment. To simulate loose capacity the overall demand was adjusted so that setup times could take up to 15% of capacity, i.e. 15 time units per period. Tight capacity was simulated by increasing each demand d_{pt} by 20% so that setups were left with no capacity in which to occur, provoking backorders of demand.

			Mean			Median		
P	Capacity	Meas. of Perf.	AM1L	ML	p	AM1L	ML	p
10	Loose	No. of Setups	33.0	43.12	**0.000**	33.0	44.0	**0.000**
		Setup Time	22.4	11.7	**0.000**	23.0	12.0	**0.000**
		Slack Capacity	49.2	60.0	**0.000**	49.8	60.8	**0.000**
		Inventory	131.1	118.1	**0.000**	121.5	109.0	**0.000**
		Backlogs	0.00	0.00	na	0.00	0.00	na
		CPU time	5.96	4.34	0.201	5.50	3.50	0.028
P	Capacity	Meas. of Perf.	AM1L	ML	p	AM1L	ML	p
10	Tight	No. of Setups	27.1	39.2	**0.000**	27.5	39.5	**0.000**
		Setup Time	16.0	2.6	**0.000**	16.0	2.0	**0.000**
		Slack Capacity	2.94	7.96	**0.000**	0.00	2.80	**0.002**
		Inventory	268.5	302.3	**0.001**	264.2	307.2	0.162
		Backlogs	36.8	15.8	**0.000**	25.0	0.0	**0.000**
		CPU time	6.72	8.06	0.551	7.0	5.0	0.317
P	Capacity	Meas. of Perf.	AM1L	ML	p	AM1L	ML	p
20	Loose	No. of Setups	66.4	84.0	**0.000**	66.00	84.00	**0.000**
		Setup Time	18.0	2.0	**0.000**	17.5	1.5	**0.000**
		Slack Capacity	123.0	139.0	**0.000**	121.9	137.9	**0.000**
		Inventory	240.	225.2	**0.000**	233.5	218.5	**0.000**
		Backlogs	0.00	0.00	na	0.00	0.00	na
		CPU time	3,292	623	**0.000**	3,268	431	**0.000**
P	Capacity	Meas. of Perf.	AM1L	ML	p	AM1L	ML	p
20	Tight	No. of Setups	51.5	64.9	**0.000**	50.5	64.5	**0.000**
		Setup Time	10.4	0	**0.000**	10.00	0	**0.000**
		Slack Capacity	9.1	14.34	**0.000**	0	4.00	**0.005**
		Inventory	631.3	655.0	**0.008**	672.5	702.0	0.317
		Backlogs	33.7	20.6	**0.000**	15.0	0	**0.000**
		CPU time	1.942	51	**0.000**	1,325	39	**0.000**

Table 1: Comparison of models AM1L and ML

Table 1 compares the performance of both models on 6 criteria, using a balanced analysis of variance test, and also the non-parametric Friedman test (Corder and Foreman; 2009) which is less likely to mistakenly indicate significance caused by outliers. Both tests used the data instance (i.e. the run) as a random blocking factor. Note the highly significant increase in numbers of setups spare and capacity, and decrease in total setup time and backlogs in the ML results compared to those for AM1L, particularly when capacity is tight.

For $P = 10$ products, model ML uses the shortcut product `P5` to economise on setups times, albeit with a larger number of actual setups, most or all of which take zero time making good use of `P5`. Table 1 shows that this is particulary pronounced under tight capacity where model ML reduces the total setup time by 85%, thus keeping backlogs to a minimum. This reduction in backlogs illustrates well the economic added value of mode ML over model AM1L. Note the fast solution times for $P = 10$ products using the default settings of the Gurobi 4.0.1 solver.

Table 1 also shows the results with twice as many products ($P = 20$), two extreme shortcut products (`P5` and `P15`), double the capacity per period, but T=4 still. The demand and setup times for products `P11` to `P20` simply replicate those for `P1` to `P10`. The models were allowed to run for a maximum of one hour. Note the predictably longer solution times and that under tight capacity model M1 solves much faster than AM1L. This is an counter-intuitive result given that model M1 has more binary and integer variables than AM1L and so might be assumed to be more combinatorial complex.

6 Modelling period-overlapping setup operations

The model can be generalized to allow setup operations to overlap periods, i.e., to permit a setup to begin in a period and end in the next period. Intuition suggests *a priori* that it is unlikely to alter lot sequences, but may be advantageous when capacity is tight and lot sizing decisions need more flexibility to reduce backlogs.

Consider the following additional decision variables:

OLS_{pqt} = 1 if an overlapping last setup at the end of period t is from product p to product q, but = 0 otherwise.

S_t is the amount of setup time that overlaps into period $t+1$, having begun at the end of period t. S_0 is known and fixed, being the amount of setup time still required in period 1 of any setup operation that started at the end of the previous period 0 but has not yet finished.

The value of S_t must be zero if there is no overlapping last setup at the end of period t:

$$S_t \;\leq\; \sum_{pq} st_{pq}\, OLS_{pqt} \quad \forall\, t \tag{20}$$

At most one setup $p \to q$ can overlap from period t to $t+1$:

$$\sum_{pq} OLS_{pqt} \;\leq\; 1 \quad \forall\, t \tag{21}$$

The value of OLS_{pqt} must be zero if $p \to q$ is not a setup initiated in period t:

$$OLS_{pqt} \;\leq\; y_{pqt} \quad \forall\, p,\, q,\, t \tag{22}$$

The capacity constraints (3) now become:

$$\sum_{p} u_p\, x_{pt} \;+\; \sum_{p,q} st_{pq}\, y_{pqt} \;+\; S_{t-1} \;-\; S_t \;+\; slack_t \;=\; Cap_t \quad \forall\, t \tag{23}$$

If $OLS_{qpt} = 1$, then product p cannot be produced as the last lot in period t and the value of z_{pt} must be reduced by 1 to reflect this. Thus constraints (9) now become:

$$\sum_{q} y_{pqt} \;+\; \alpha_{p,t+1} \;=\; z_{pt} \;+\; \sum_{q} OLS_{qpt} \quad \forall\, p,\, t \tag{24}$$

Thus the model for lot sizing and sequencing with non-triangular setup times, setup-state carryover between periods, and period-overlapping setup operations, is specified by expressions (1-2, 5-12, 14, 16, 18-24).

The model was applied to the 25 instances of data set with 10 products and tight capacity. The results did not show any significant benefit. The 25 AM1L solutions did not change at all and the supposedly longer solution time is due to sampling variability (p-value = 0.90). Only 2 of the 25 ML solution instances decreased the number of setups (by about 5%), but the solutions did not change at all on the other 5 criteria and the seemingly10% faster solution time can be ascribed to sampling variability (p-value = 0.26).

7 Conclusions and Future Research

This paper has developed a new model for lot sizing and sequencing with a polynomial number of constraints that can handle the multiple lot per product per period that arise in the presence of non-triangular sequence-dependent setup times. The computational tests validated and confirmed that the multiple-lots feature of the model enables more efficient production than when the formulation is restricted to single lots per product per period. The model can also be faster to solve than in the latter case, despite being more complex computationally, maybe because for some problem instances (such as our tests above) there is an outstanding optimal ML solution that is quickly identified whereas an optimal AM1L solution may not be so clearly superior and hence more difficult to find.

The computational tests also show that future research must also include the development of faster solution methods for large instances, possibly via exact methods such as (1) Lagrangian Relaxation coupled with decomposition into single periods where the submodels can be solved very rapidly, or via heuristic methods such as (2) *Relax-&-Fix* methods of various types (Ferreira et al.; 2009), (3) depth-first heuristics (Zhang; 2000), or (4) local branching (Fischetti and Lodi; 2003). Future work will also computationally compare the model against a functionally-equivalent GLSP model and Menezes et al. (2010)'s iterative method with Miller-Tucker-Zemlin subtour elimination constraints.

Given that the demand forecasts usually change as time advances from one period to the next, the question arises as to whether it is worthwhile to schedule over even a medium term horizon, let alone a long-term one. Frequent rescheduling (Haase and Kimms; 1999) implies that firm schedules should really only be specified for the immediate to short term over which demand forecasts will not change (much), while approximate or aggregate planning (rather than scheduling should be carried out for medium to long term. This poses interesting (and not trivial) research challenges about how to perform planning that result in effective and efficient short term schedules (Clark; 2003).

Acknowledgements: Our thanks go to Socorro Rangel of UNESP for a critical reading of an earlier version of this paper, and to the anonymous referees who also made valuable suggestions for improving it. This research was partly supported by a *Global Research Award* from the *Royal Academy of Engineering*, London, and an FP7 Marie Curie *International Research Staff Exchange Scheme* (IRSES) grant from the European Commission.

References

Araújo, S. A., Arenales, M. N. and Clark, A. R. (2007). Joint rolling-horizon scheduling of materials processing and lot-sizing with sequence-dependent setups, *Journal of Heuristics* **13**(4): 337–358.

Clark, A. R. (2003). Optimization approximations for capacity constrained material requirements planning, *International Journal of Production Economics* **84**(2): 115–131.

Clark, A. R., Almada-Lobo, B. and Almeder, C. (2011). Editorial: Lot sizing and scheduling - industrial extensions and research opportunities, special issue on lot sizing and scheduling, *International Journal of Production Research* **49**(9): 2457–2461.

Clark, A. R. and Clark, S. J. (2000). Rolling-horizon lot-sizing when setup times are sequence-dependent, *International Journal of Production Research* **38**(10): 2287–2308.

Clark, A. R., Morabito, R. and Toso, E. A. V. (2010). Production setup-sequencing and lot-sizing at an animal nutrition plant through ATSP subtour elimination and patching, *Journal of Scheduling* **13**(2): 111–121.

Claus, A. (1984). A new formulation for the travelling salesman problem, *SIAM Journal on Algebraic and Discrete Methods* **5**: 21–5.

Corder, G. W. and Foreman, D. I. (2009). *Nonparametric Statistics for Non-Statisticians: A Step-by-Step Approach*, Wiley-Blackwell.

Dantzig, G., Fulkerson, R. and Johnson, S. (1954). Solution of a large-scale traveling-salesman problem, *Operations Research* **2**: 393–410.

Drexl, A. and Kimms, A. (1997). Lot sizing and scheduling - survey and extensions, *European Journal of Operational Research* **99**: 221–235.

Ferreira, D., Morabito, R. and Rangel, S. (2009). Solution approaches for the soft drink integrated production lot sizing and scheduling problem, *European Journal of Operational Research* **196**: 697–706.

Fischetti, M. and Lodi, A. (2003). Local branching, *Mathematical Programming, Series B* **98**: 23–47.

Fleischmann, B. and Meyr, H. (1997). The general lotsizing and scheduling problem, *OR Spektrum* **19**(1): 11–21.

Fourer, R., Gay, D. M. and Kernighan, B. W. (2003). *AMPL - A Modeling Language for Mathematical Programming*, second edn, Duxbury Press / Brooks-Cole Publishing Company, USA. http://www.ampl.com/.

Gurobi Optimization Inc. (2010). Gurobi optimizer version 3.0.0. http://www.gurobi.com.

Haase, K. and Kimms, A. (1999). Lot sizing and scheduling with sequence dependent setup costs and times and efficient rescheduling opportunities, *International Journal of Production Economics* **66**: 159–169.

Karimi, B., Fatemi Ghomia, S. M. T. and Wilson, J. M. (2003). The capacitated lot sizing problem: a review of models and algorithms, *Omega* **31**: 365–378.

Kovács, A., Brown, K. N. and Tarim, S. A. (2009). An efficient MIP model for the capacitated lot-sizing and scheduling problem with sequence-dependent setups, *International Journal of Production Economics* **118**(1): 282 – 291.
URL: *http://www.sciencedirect.com/science/article/B6VF8-4T9CCSG-3/2/8ec751c0f847ed8624677b88cf991987*

Menezes, A., Clark, A. and Almada-Lobo, B. (2010). Capacitated lot-sizing and scheduling with sequence-dependent, period-overlapping and non-triangular setups, *Journal of Scheduling* . in press.

Meyr, H. (2000). Simultaneous lot-sizing and scheduling by combining local search with dual optimization, *European Journal of Operational Research* **120**: 311–326.

Miller, C. E., Tucker, A. W. and Zemlin, R. A. (1960). Integer programming formulations and traveling salesman problems, *Journal of ACM* **7**: 326–329.

Öncan, T., Altinel, K. and Laporte, G. (2009). A comparative analysis of several asymmetric traveling salesman problem formulations, *Computers and Operations Research* **36**(3): 637–654.

Toso, E. A. V., Morabito, R. and Clark, A. R. (2009). Lot-sizing and sequencing optimisation at an animal-feed plant, *Computers and Industrial Engineering* **57**: 813–821.

Zhang, W. (2000). Depth-first branch-and-bound versus local search: A case study, *AAAI/IAAI Proc. 17th National Conf. on Artificial Intelligence (AAAI-2000)*, Austin, Texas, pp. 930–935. Webpage: citeseer.ist.psu.edu/470650.html.