Proceedings of the
2nd International Workshop
on Lot Sizing

2011 Istanbul, Turkey

24 - 26 August 2011

Ozyegin University

CONTENTS

Workshop Organization

Local Organizing Committee

Meltem Denizel	Ozyegin University
Orsan Ozener	Ozyegin University
Elif Akcali	University of Florida
	Visiting at Ozyegin University

Steering Committee

Nabil Absi	(France)
Bernardo Almada-Lobo	(Portugal)
Alistair Clark	(England)
Hark-Chin Hwang	(South Korea)
Raf Jans	(Canada)
Atle Nordli	(Norway)
Safia Kedad-Sidhoum	(France)
H. Edwin Romeijn	(USA)
Hartmut Stadtler	(Germany)
Horst Tempelmeier	(Germany)
Michal Tzur	(Israel)
Albert P. M. Wagelmans	(The Netherlands)
Laurence A. Wolsey	(Belgium)

Participants

Last Name	First Name	Country
Absi	Nabil	France
Aggoune	Riad	Luxembourg
Akcali	Elif	USA
Almada-Lobo	Bernardo	Portugal
Almeder	Christian	Austria
Amorim	Pedro	Portugal
Copil	Karina	Germany
Dauzere-Peres	Stephane	France
Denizel	Meltem	Turkey
Fragkos	Ioannis	England
Gomez Urrutia	Edwin	Luxembourg
Goren	Hacer	Turkey
Gunther	Hans-Otto	Germany
Helber	Stefan	Germany
Helmrich	Mathijn Retel	The Netherlands
Hwang	Hark-Chin	Korea
Iris	Cagatay	Turkey
Jans	Raf	Canada
Kedad-Sidhoum	Safia	France
Koca	Esra	Turkey
Kucukyavuz	Simge	USA
Meyr	Herbert	Germany
Onal	Mehmet	Turkey
Ozener	Orsan	Turkey
Rapine	Christophe	France
Romero-Morales	Dolores	England
Romeijn	Edwin	USA
Seeanner	Florian	Germany
Simpson	Natalie	USA
Solyali	Oguz	North Cyprus
Tempelmeier	Horst	Germany
van den Heuvel	Wilco	The Netherlands
Van Vyve	Mathieu	Belgium
Wagelmans	Albert P.M.	The Netherlands
Wolsey	Laurence A.	Belgium

Program

Time	Tuesday, August 23, 2011
19:00-21:00	**Welcome Dinner**
	Wednesday, August 24
8:30-9:00	**Welcome**
9:00-9:30	**Opening Session**
9:30-10:50	Capacitated Lotsizing and Scheduling: tackling sequence dependent setups using column generation
	Luis Guimaraes, Diego Klabjan, **Bernardo Almada-Lobo**
	Capacitated lot sizing with parallel machines, sequence-dependent setups and common setup operators: A practical case
	Horst Tempelmeier, Sascha Herpers, **Karina Copil**
10:50-11:20	Coffee Break
11:20-12:40	Hybrid approaches for the capacitated lot sizing problem with setup carryover and backordering
	Hacer Guner Goren, Semra Tunali, Raf Jans
	A hybrid heuristic for the production-routing problem
	Yossiri Adulyasak, Jean-Franois Cordeau, **Raf Jans**
12:40-14:10	Lunch
14:10-16:10	Finding pareto optimal solutions for the multi-objective economic lot-sizing problem
	Wilco van den Heuvel, H. Edwin Romeijn, **Dolores Romero Morales**
	The economic lost sizing problem with lost sales and bounded inventory
	Hark-Chin Hwang, Wilco van den Heuvel, **Albert P.M. Wagelmans**
	A holding cost bound and a new heuristic for the economic lot-sizing problem
	Wilco van den Heuvel, Albert P.M. Wagelmans
16:10-16:40	Coffee Break
16:40-18:00	The economic lot-sizing problem with an emission constraint
	Mathijn Retel Helmrich, Wilco van den Heuvel, Albert Wagelmans, Raf Jans
	Analysis of Green Lot-Sizing Problems
	Nabil Absi, Stephane Dauzere-Peres, **Safia Kedad-Sidhoum**, Bernard Penz, Christophe Rapine

Time	Thursday, August 25
8:30-9:50	A maritime inventory routing problem
	Agostinho Agra, Henrik Andersson, Marielle Christiansen, **Laurence Wolsey**
	A Fixed-charge transportation view for solving big-bucket multi-item lot-sizing
	Mathieu Van Vyve
9:50-10:20	Coffee Break
10:20-12:20	Improvements of an integrated approach for lot-sizing and detailed scheduling
	Edwin Gomez, Riad Aggoune, Stephane Dauzere-Peres
	The block planning approach for continuous time based lot sizing and scheduling: A change of paradigm?
	Hans-Otto Gunther
	Combining the principles of Variable Neighborhood Decomposition Search and Fix&Optimize heuristic to solve multi-level lot-sizing and scheduling problems
	Florian Seeanner, Bernardo Almada-Lobo, Herbert Meyr
12:20-14:00	Lunch
14:00-15:20	Dynamic Probabilistic Lot-Sizing with Service Levels
	Samuya Goel, **Simge Kucukyavuz**
	Dynamic capacitated lot sizing with random demand subject to a backlog/waiting-time oriented δ-service level measure
	Stefan Helber, Florian Sahling, Katja Schimmelpfeng
15:20-15:50	Coffee Break
15:50-17:10	Primal and dual bounds for long-horizon lot size problems with tight capacity constraints
	Silvio Araujo, Bert De Reyck, Zeger Degraeve, **Ioannis Fragkos**, Raf Jans
	The impact of modeling on robust inventory management under demand uncertainty
	Oguz Solyali, Jean-Francois Cordeau, Gilbert Laporte
18:00-22:00	Bosphorus Dinner Cruise

Time	Friday, August 26
8:30-9:00	Discussion: Next meeting
9:00-11:00	Multilevel capacitated lot sizing and scheduling: A model overview and comparison Renate Traxler, **Christian Almeder**
	A polynomial time algorithm for the multi-stage lot-sizing problem with production capacities **Hark-Chin Hwang**, Hyun-soo Ahn, Philip Kaminsky
	New Algorithms and New Questions for the Multiple Stage Lot Sizing Problem **Natalie Simpson**
11:00-11:30	Coffee Break
11:30-12:50	Economic Lot Sizing Problem with Production Capacities and Perishable Items **Mehmet Onal**, H. Edwin Romeijn, Wilco van den Heuvel
	Integrating Lot Sizing and Scheduling with the Vehicle Routing Problem: A special look into perishable products Anderson Meneses, **Pedro Amorim**, Bernardo Almada-Lobo, Christian Almeder
12:50-14:30	Lunch

Foreword

At Ozyegin University, we are very happy to host the 2nd International Workshop on Lot-Sizing in Istanbul this year.

It is rather interesting that although lot-sizing has always been a central research topic within several related disciplines such as Industrial Engineering, Operations Research, and Operations Management, a special interest group had not been formed until last year when the 1st workshop was organized by Ecole Nationale Superieure des Mines de Saint-Etienne in Gardanne, France. The well deserved success of this breakthrough paved the way to the 2nd International Workshop here in Istanbul. Hopefully, more will follow in the future.

The main objective of this workshop is to bring together researches from all over the world to exchange ideas on lot-sizing and discuss related research problems that we come across in different environments and in many different forms. The recent advances in production and supply chain research, in closed loop supply chains and logistics, present more complicated and challenging dimensions to be considered and it is a pleasure to see all these different topics addressed by our community in a timely and rigorous manner.

During this workshop, many dedicated participants from all over the world will be looking at over 20 exciting and thought provoking talks that will unquestionably generate much needed discussion and a great deal of new research ideas, hence, enhance our ongoing work.

We would like to acknowledge the courage and vision of the organizers of the first workshop, and express our appreciation. Special thanks are due to Nabil Absi, Bernardo Almada-Lobo Stephane Dauzere-Peres, and Safia Kedad-Sidhoum for their efforts.

I wish us all a fruitful workshop and hope that you all enjoy your stay in Istanbul.

Meltem Denizel
Chair, Organizing Committee

Extended Abstracts

Analysis of Green Lot-Sizing Problems

Nabil Absi Ecole des Mines de Saint-Etienne, CMP - Site Georges Charpak, Gardanne, 13541, France.
absi@emse.fr

Stephane Dauzere-Peres Ecole des Mines de Saint-Etienne, CMP - Site Georges Charpak, Gardanne, 13541, France.
dauzere-peres@emse.fr

Safia Kedad-Sidhoum Laboratoire d'Informatique de Paris 6, 75 252 Paris Cedex 05, France.
safia.kedad-sidhoum@lip6.fr

Bernard Penz Laboratoire G-SCOP, INPG - Université Joseph Fourier - Grenoble, France.
bernard.penz@grenoble-inp.fr

Christophe Rapine Laboratoire G-SCOP, INPG - Université Joseph Fourier - Grenoble, France.
christophe.rapine@g-scop.fr

1   Introduction

One of the main objectives of green logistics is to evaluate the environmental impact of different distribution and production strategies to reduce the energy usage in logistics activities. Although, in the last decades, there has been a strong interest on green logistics and environmental aspects, current logistics practices are still most often not in line with these interests. The classical production and distribution models focus on the cost minimization subject to operational constraints. Considering new green logistics objectives will lead to new problems which can lead to new combinatorial optimization models [3]. [2] insists on the potential impact of operational decisions on carbon emissions and the need for Operations Management research that incorporates carbon emission concerns. The authors point the fact that the contribution of Operational Research in this area is almost absent.

2

In this work, we address multi-sourcing lot-sizing problems with carbon emission constraints. These new constraints are induced from a maximum allowed carbon dioxide emission coming from legislations, green taxes or initiatives of companies. Contrary to the paper of [2], where a global limit of carbon emission is taken into account, we consider a maximum unitary environmental impact allowed for each item. The multi-sourcing lot-sizing problem can be defined as the problem faced by a company that has to determine over a given planning horizon of T periods, when, where and how much to produce an item to satisfy a deterministic time-dependent demand. Different production locations and transportation modes are available to satisfy a given demand. We consider M different supplying modes, where a mode corresponds to the combination of a production location and a transportation mode. No capacity constraints are imposed. We study four types of carbon emission constraints:

- Periodic carbon emission constraint,

- Cumulative carbon emission constraint,

- Rolling carbon emission constraint,

- Global carbon emission constraint.

The fourth class of constraints has similar drawbacks than the ones of the constraint used in [2]. Mathematical formulations have been proposed for each class of problems [1]. The objective function consists in minimizing the total supplying costs (fixed and variable) as well as the inventory costs.

We show that the uncapacitated multi-sourcing lot-sizing problem with periodic carbon emission constraints can be solved using a polynomial dynamic programming algorithm. We prove that the uncapacitated multi-sourcing lot-sizing problems with cumulative carbon emission constraints, rolling carbon emission constraints or global carbon emission constraints are $\mathcal{NP}$-hard. We will also discuss some of the perspectives of this work.

2   Complexity results

In this section, we discuss two types of carbon emission constraint: The periodic and the cumulative cases.

2.1   The uncapacitated single-item lot-sizing problem with periodic carbon emission constraints

We study the uncapacitated lot-sizing problem with the periodic carbon emission constraint which ensures that in each period t, the average amount of carbon emission

per product ordered does not exceed the impact limit $E_t^{\max}$. For the sake of conciseness, we denote by $\overline{e}_t^m$ the value $(e_t^m - E_t^{\max})$, where e_t^m represents the environmental impact (carbon emission) related to supplying one unit with mode m at period t. A mode m is called *ecological* or *environment friendly* in period t if $\overline{e}_t^m \leq 0$.

We establish that the ULS-PC problem can be solved in polynomial time. More precisely, we show that we can transform ULS-PC into a standard lot-sizing problem with M^2 modes, using a preprocessing step in $\mathcal{O}(M^2T)$. Thus, standard lot-sizing combinatorial algorithms can be used to solve the problem. Our analysis is based on the following dominance properties; there exists an optimal solution for the ULS-PC problem that uses at most two modes in each period: One ecological mode and possibly one non-ecological mode.

We deduce that a lot-sizing problem with periodic carbon emission constraints is polynomial if and only if the corresponding lot-sizing problem without periodic carbon emission constraints is polynomial. Hence the periodic carbon emission constraint does not modify the complexity status of the problem, but only increases (in a reasonable amount) the computational time, due to the preprocessing step and the increase of the number of modes. Roughly speaking, the algorithmic complexity of the lot-sizing problem is increased by a factor M^2 due to these constraints. However, our result is restricted to linear supplying costs.

2.2 The uncapacitated single-item lot-sizing problem with cumulative carbon emission constraints

We now consider the uncapacitated lot-sizing problem with cumulative carbon emission: For each period t, the average amount of carbon emission per product ordered from the first period up to t should not exceed an impact limit $E_t^{\max}$. Similarly to the case of periodic carbon emission constrains, it is dominant to use at most two modes per period.

It seems that the situation is very similar to the case with periodic carbon emission constraints. It turns out that problem ULS-CC is in fact far more difficult to solve than the ULS-PC problem. It is possible to show that the zero inventory ordering (ZIO) property is not always a dominant solution for the problem, and that the best ZIO policy may perform arbitrarily bad. We prove that the problem is $\mathcal{NP}$-hard, even on stationary instances with unit demands. The reduction is made from a special version of the SubSetSum problem with an additional cardinality constraint on the size of the selected set.

3   Conclusion and further research directions

We believe the integration of carbon emission constraints in lot-sizing problems lead to relevant and original problems. We proposed and studied four types of carbon emission constraints and drive complexity analysis. For future research, it would be interesting to propose exact methods to solve the $\mathcal{NP}$-hard problems, or approximation algorithms for some special cases.

Different sensitivity analysis can also be conducted by different actors. In fact, the impact of introducing carbon emission constraints in a supply chain can be analyzed from different points of views. Local authorities or governments could be interested in conducting some analysis to find the best way to introduce new legislations or a succession of legislations on carbon emission without excessively penalizing manufacturers. Companies could also be interested in conducting some analysis to find out which carbon emission constraint is more relevant for them. In fact, they may be interested in displaying carbon footprints on their products while keeping their competitiveness. Some computational experiments based on the integer linear programming formulations will be presented.

References

[1] N. Absi, S. Dauzère-Pérès, S. Kedad-Sidhoum, B. Penz et C. Rapine. Lot-sizing with carbon emission constraints. In International Workshop on Lot-Sizing, Gardanne, France, August 2010.

[2] S. Benjaafar, Y. Li and M. Daskin. Carbon Footprint and the Management of Supply Chains: Insights from Simple Models. *Working Paper*, University of Minnesota, USA, 2010.

[3] A. Sbihi, R.W. Eglese. Combinatorial optimization and Green Logistics. *4OR: A Quarterly Journal of Operations Research* **5(2)** 99–116, 2007.

A Hybrid Heuristic for the Production Routing Problem

Yossiri Adulyasak, Jean-Francois Cordeau, Raf Jans
HEC Montral, Montal, Canada

1   Introduction

It is more than five decades ago that two classical problems in logistics, namely, lot-sizing and vehicle routing, were introduced in the Operations Research literature. The lot-sizing problem (LSP) involves production lot size and inventory decisions over the planning horizon, while the vehicle routing problem (VRP) involves distribution and routing decisions in order to satisfy customer demands in each period. There are a large number of studies related to these two problems, but most of them are mainly focused on each separate problem. In this paper, we consider the integration of the lot-sizing problem and the vehicle routing problem. This problem is called the Production Routing Problem (PRP). The total costs of the entire chain can be reduced when all activities are optimized simultaneously, compared to a sequential approach. This provides an opportunity to achieve greater benefits through solving integrated problems.

Some research has been done on integrating production and distribution decisions. When we consider the lot-size and distribution decisions simultaneously, we obtain the two-level lot-sizing problem with direct shipments. The Inventory Routing Problem (IRP), on the other hand, incorporates the routing decisions, but ignores the production decisions, as it is assumed that the production quantities are known in advance. The PRP is hence a generalization of both models.

We consider the PRP with one product and multiple capacitated vehicles. The objective function is to minimize minimize the total production, setup, inventory and routing costs. Constraints include demand balance equations and setup decisions at the plant and customer level, maximum inventory levels, vehicle capacities and routing constraints, including subtour elimination constraints.

Due to its complexity, the PRP has not been studied extensively and only a few papers analyse this problem. Chandra (1993) and Chandra and Fisher (1994) are the first to discuss the opportunity of coordinating lot-size decisions with distribution and routing planning. Further developments are made by Fumero and Vercellis (1999), Lei et al. (2006), Bard and Nananukul (2009, 2010), Ruokokoski et al. (2010), and

Archetti et al. (2011). Several metaheuristics are employed to solve the PRP. Boudia et al. (2007) developed a greedy randomized adaptive search procedure. Boudia and Prins (2009) presented a memetic algorithm. Tabu search procedures were proposed by Bard and Nananukul (2009) and Armentano et al. (2011).

2   The hybrid heuristic

In this paper, we present a hybrid decomposition based heuristic to solve the PRP. The basic idea is to reduce the complexity of the model by decomposing the problem into subproblems which are easier to solve. A natural way is to separate the setup and routing part that contains a large number of binary variables and complex sets of constraints, and the decisions on the production and delivery quantity which contain the set of the remaining continuous variables. The procedure comprises of two main phases. Initial solutions are created in the initialization phase and the solutions are improved iteratively in the improvement phase.

Initial solutions are generated by sequentially solving two decomposed problems, namely the production-distribution (PD) and routing (R) subproblems. The first subproblem is used to determine a production, inventory and distribution plan, and the second subproblem is to determine the routes to serve the customers according to the distribution plan obtained from the first subproblem. The solution is stored and the algorithm starts finding another initial solution with a different production setup configuration. We apply the local branching technique within the PD subproblem in order to force the model to produce a different solution from the ones we have already generated. This process is repeated until the maximum number of initial solutions is reached. The purpose of generating many initial solutions is to avoid local optima issues by observing different and various search spaces.

In the improvement phase, the algorithm will try to improve the initial solutions by using a heuristic that employs the adaptive large neighborhood search (ALNS) and linear programming (LP) techniques. The ALNS is proposed by Ropke and Pisinger (2006). The basic idea of the algorithm is to destroy the current solution and repair the solution to seek for improvement. The nice feature of the ALNS framework is that it incorporates several heuristics to perform the movements to search the neighborhood and they are randomly selected with different probabilities based on the empirical scores. We use an adaptation of the ALSN framework to handle the binary variables, and the remaining continuous variables are evaluated by solving a network flow model that is embedded into the transformation operators of the ALSN. The heuristic procedure terminates when it reaches the maximum number of iterations or the maximum number of explored node candidates.

3 Computational results

In order to evaluate the efficiency of our algorithm, we have performed experiments using two benchmark test sets. The problems in the first set have 6 periods and 14, 50 or 100 customers. This test set was used by Archetti et al. (2011). The results indicate that our new approach outperforms on average the solutions of Archetti et al. (2011) both in terms of solution quality and CPU time for the problems with 50 and 100 customers, but not for the class with 14 customers. The average improvement after 100 iterations is 0.3% and 0.2% for the two larger classes. For the largest class, our algorithm (after 100 iterations) is approximately 4 times as fast as the Archetti et al. algorithm, whereas the CPU times for the middle class are comparable. The average deterioration is 0.8% for the smallest class.

The problems in the second test set have all 20 periods and 50 to 200 customers. This test set has been used by Boudia et al. (2007), Boudia and Prins (2009), Bard and Nananukul (2009) and Armentano et al. (2011). The previous best solutions were obtained by the tabu search heuristic of Armentano et al. (2011). For all three classes (50, 100 and 200 customers), our algorithm outperforms the Armentano et al. heuristic in terms of solution quality and CPU time. When looking at the solutions after 100 iterations of our algorithm, we obtain improvements of 0.6%, 5.0% and 6.3% for the three classes. These solutions are obtained in less time than the solutions of Armentano et al.

References

[1] Archetti, C., Bertazzi, L., Peletta, G., and Grazia Speranza, M. (2011). Analysis of the maximum level policy in a production-distribution system. Computers and Operations Research,38, 1731-1746.

[2] Armentano, V. A., Shiguemoto, A. L., and Lkketangen, A. (2011). Tabu search with path relinking for an integrated production-distribution problem. Computers and Operations Research, 38, 1199-1209.

[3] Bard, J. F., and Nananukul, N. (2009). Heuristics for a multiperiod inventory routing problem with production decisions. Computers and Industrial Engineering, 57(3), 713-723.

[4] Bard, J. F., and Nananukul, N. (2010). A branch-and-price algorithm for an integrated production and inventory routing problem. Computers and Operations Research, 37(12), 2202-2217.

[5] Boudia, M., Louly, M. A. O., and Prins, C. (2007). A reactive GRASP and

path relinking for a combined production-distribution problem. Computers and Operations Research, 34(11), 3402-3419.

[6] Boudia, M., and Prins, C. (2009). A memetic algorithm with dynamic population management for an integrated production-distribution problem. European Journal of Operational Research, 195(3), 703-715.

[7] Chandra, P. (1993). A Dynamic Distribution Model with Warehouse and Customer Replenishment Requirements. The Journal of the Operational Research Society, 44(7), 681-692.

[8] Chandra, P., and Fisher, M. L. (1994). Coordination of production and distribution planning. European journal of operational research, 72(3), 503-517

[9] Fumero, F., and Vercellis, C. (1999). Synchronized Development of Production, Inventory, and Distribution Schedules. Transportation Science, 33(3), 330-340.

[10] Lei, L., Liu, S., Ruszczynski, A., and Park, S. (2006). On the integrated production, inventory, and distribution routing problem. IIE Transactions, 38(11), 955 - 970.

[11] Ropke, S., and Pisinger, D. (2006). An Adaptive Large Neighborhood Search Heuristic for the Pickup and Delivery Problem with Time Windows. Transportation Science, 40(4), 455-472.

[12] Ruokokoski, M., Solyal, O., Cordeau, J.-F., Jans, R., and Sral, H. (2010). Efficient Formulations and a Branch-and-Cut Algorithm for a Production-Routing Problem, Working paper: HEC Montral.

A maritime inventory routing problem: Time discrete formulations and valid inequalities

Agostinho Agra University of Aveiro
aagra@ua.pt

Henrik Andersson Norwegian University of Science and Technology
henrik.andersson@iot.ntnu.no

Marielle Christiansen Norwegian University of Science and Technology
marielle.christiansen@iot.ntnu.no

Laurence Wolsey Catholic University of Louvain
laurence.Wolsey@uclouvain.be

1   Introduction

We consider a single commodity maritime inventory routing problem in which the production rates and consumption rates are time varying. There is a heterrogeeous fleet and multiple production/consumption ports with limited storage capacity. In this extended abstract we formulate the problem as a pure single commodity fixed charge network flow problem. This formulation can then be strengthened by the addition of valid inequalities and solved with the choice of appropriate branching rules.

We now describe the network. The commodity enters the time-phased network at the production ports, flows along the routes selected by the vessels, and leaves the network at the consumption ports. Figure 1 gives an example of the fixed charge arcs corresponding to the progress of a vessel between ports (S arcs) and a simplified representtion of the movement of a vessel though a port. The fixed charge network corresponding to a consumption (unloading) port is a standard lot-sizing network. The main question concerns the interface between thew two networks. To be precise we impose certain rules describing the behavior of a vessel at a port:

In each period the vessel is either in waiting mode (W) or in operating (loading/unloading) mode (O), and
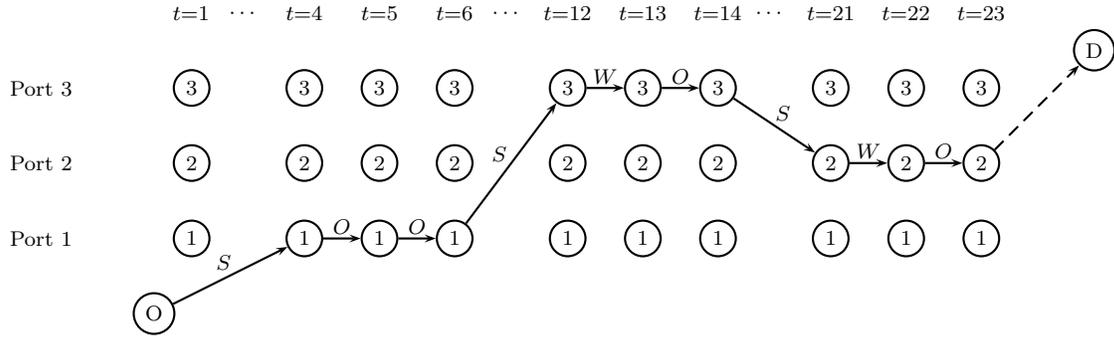
Figure 1: Example of the movement of a ship in a time expanded network. The arc labels are O for operating, W for waiting and S for sailing.

i) there is at least one period in operating mode,

ii) once in operating mode, the vessel remans in operating mode till it quits the port.

In Figure 2 we indicate the network for the passage of a single vessel at an unloading port. The thick black arcs indicate the fixed charge arcs that are active. The last row of nodes correspond to a standard lot-sizing (unloading) network with unloading taking place in periods 3 and 4. The upper two rows of nodes correspond to the vessel arriving at the port, spending period 2 in waiting mode, periods 3 and 4 in operating (unloading) mode and then departing in period 5.
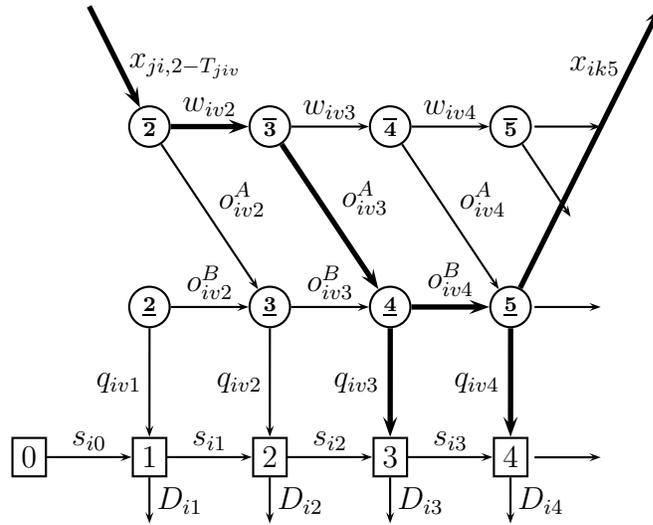
Figure 2: Discharge operation at port i for ship v in the extended network.

2 Formulation

To model the problem as a mixed integer linear program, the following notation is introduced:

N is the set of production and consumption ports with indices i and j,

T the set of time periods with index t,

V the set of ships with index v.

The data, excluding costs, are:

B_{it} the berth capacity in number of ships at port i in time period t,

D_{it} the consumption at port i in period t,

P_{it} the production at port i in period t,

J_i is 1 if i is a loading port and -1 if i is a discharge port,

K_v the capacity of ship v,

Q_v the upper bound on the amount ship v loads/discharges per time period,

$\overline{S}_{it}$ the upper bound on the inventory level at port i at the end of time period t,

$\underline{S}_{it}$ the lower bound on the inventory level at port i at the end of time period t,

S_i^0 the inventory level in port i at the beginning of the planning horizon,

$o(v)$ the initial position for ship v,

$d(v)$ the artificial end node for ship v,

T_{ijv} the sailing time from port i to port j for ship v.

We define the following variables:

o_{ivt} is 1 if ship v operates in port i in time period t, 0 otherwise,

x_{ijvt} is 1 if ship v sails from port i to port j in time period t, 0 otherwise,

w_{ivt} is 1 if ship v is waiting outside port i in time period t, 0 otherwise,

q_{ivt} the quantity loaded/discharged in time period t at port i by ship v,

s_{it} inventory level in port i at the end of time period t,

o_{ivt}^A indicates whether ship v starts to operate at port i in period t,

o_{ivt}^B indicates the succeeding operations at that port.

The model constraints include:

i) the fixed charge network describing the trajectory/path of the vessels

$$\sum_{j \in N \cup \{d(v)\}} x_{o(v)jvt} = 1, \qquad \forall v \in V, t \in T, \qquad (1)$$

$$\sum_{i \in N \cup \{o(v)\}} x_{id(v)vt} = 1, \qquad \forall v \in V, t \in T, \qquad (2)$$

$$\sum_{j \in N \cup \{o(v)\}} x_{jiv,t-T_{jiv}} + w_{iv,t-1} = w_{ivt} + o_{ivt}^A \qquad \forall v \in V, i \in N, t \in T, \qquad (3)$$

$$o_{iv,t-1}^A + o_{iv,t-1}^B = o_{ivt}^B + \sum_{j \in N \cup \{d(v)\}} x_{ijvt}, \qquad \forall v \in V, i \in N, t \in T, \qquad (4)$$

$$x_{ijvt}, w_{ivt}, o_{ivt}^A, o_{ivt}^B \in \{0, 1\}, \qquad (5)$$

12

where (3) and (4) correspond to path balance constraints at the nodes of the first two layers in Figure 2, and to which we need to add corresponding flow variables and flow conservation constraints.

ii) there are the port operations:

$$\sum_{v \in V} o_{ivt} \leq B_{it}, \qquad \forall i \in N, t \in T, \qquad (6)$$

$$0 \leq q_{ivt} \leq Q_v o_{ivt}, \qquad \forall v \in V, i \in N, t \in T, \qquad (7)$$

$$s_{i,t-1} + \sum_{v \in V} q_{ivt} = D_{it} + s_{it}, \qquad \forall i \in N : J_i = -1, t \in T, \qquad (8)$$

$$s_{i,t-1} + P_{it} = \sum_{v \in V} q_{ivt} + s_{it}, \qquad \forall i \in N : J_i = 1, t \in T, \qquad (9)$$

$$\underline{S}_{it} \leq s_{it} \leq \overline{S}_{it}, s_{i0} = S_i^0, \qquad \forall i \in N, t \in T, \qquad (10)$$

$$o_{ivt} \in \{0, 1\}, \qquad \forall v \in V, i \in N, t \in T. \qquad (11)$$

with (8) for an unloading port and (9) for a loading port,
and finally coordination constraints between the vessel and port networks;

$$o_{ivt}^A + o_{ivt}^B = o_{ivt}, \qquad \forall v \in V, i \in N, t \in T. \qquad (12)$$

The talk contains a discussion of valid inequalities for the model, branching strategies and presentation of computational results on a set of small to medium-sized instances.

Primal and dual bounds for long-horizon lot size problems with tight capacity constraints

Silvio Araujo   Universidade Estadual Paulista
saraujo@ibilce.unesp.br

Bert De Reyck   University College London
bdereyck@london.edu

Zeger Degraeve   London Business School
Zdegraeve@london.edu

Ioannis Fragkos   University College London
i.fragkos@ucl.ac.uk

Raf Jans   HEC Montreal
raf.jans@hec.ca

1   Introduction

The aim of this paper is to design a fast heuristic for the capacity constrained lot size problems with setup times (CLST) that provides good solutions and a strong lower bound to assess their quality. Per-period Dantzig-Wolfe decompositions of two strong reformulations of the CLST are proposed. The main advantage of these decompositions is that they provide lower bounds which are stronger than those obtained by most other approaches. Along the lines of [8] we propose a novel fast subgradient-based hybrid scheme that combines Lagrange relaxation and column generation. This scheme outperforms simplex-based column generation, Lagrange relaxation and subgradient-based column generation (in which the restricted master programs are solved with subgradient optimization).

The new hybrid scheme is embedded in a branch-and-price framework, designed specifically to obtain good feasible solutions fast. To achieve this, we recover a primal solution of the restricted master using the volume algorithm [1], and branch on the resulting fractional setup variables. Moreover, we integrate in a customized fashion

recent MIP-based heuristic approaches, such as the relaxation induced neighborhoods and selective dives [2], with existing ones such as the forward/backward smoothing heuristic [3]. Extensive computational experiments show that the branch-and-price heuristic performs very well against other competitive approaches, especially in long-horizon problems.

2 Lower bounds from a hybrid column generation approach

We study and extend the per-period decomposition of the shortest path formulation of the CLST, as proposed in [3]. The authors suggest an improved lower bound, but their computational experiments reveal that it is too time consuming to be used in practice. We apply a novel transformation on the master linking constraints. The transformed master has reduced dual space and the convergence of column generation and lagrange relaxation is orders of magnitude faster.

A related lagrange relaxation was proposed recently by [5] for the CLST without setup costs. The authors employ the facility location formulation, originally proposed in [6] for facility location problems. We devise a Danzig-Wolfe decomposition of the facility location formulation, and show how the inclusion of valid inequalities in the subproblem corresponds to adding dual optimal inequalities in the restricted master. For all formulations, we demonstrate the performance of a hybrid scheme that combines column generation and lagrange relaxation and computes the lower bound fast.

3 Heuristic solutions and branch-and-price

We develop a branch-and-price algorithm in a heuristc manner, with the aim to compute good feasible solutions. At each node of the tree we employ a successive rounding heuristic that uses the smoothing subroutine of [3]. We perfom guided dives, i.e., fix the variables that are the same in the root node and in the best feasible solution. The volume algorithm recovers an approximate primal solution used for branching. After a predefined node limit, a dynamic programming based heuristic is invoked, and the search space shifts to the neighborhood of the best feasible solution.

4 Computational results

Table 1 compares the time efficiency of several decomposition methods when they are applied to 7 CLST instances taken from [3]. All methods solve the period decomposition of the facility location formulation (the results for the shortest path formulation

are similar). Subscript 1 refers to the period subproblem, whereas subscript 2 refers again to the period subproblem amended by valid inequalities. HB refers to the combined lagrange relaxation/column generation method and ACG to approximate column generation, in which the master is solved with lagrange relaxation.

The first column shows the time that the most competitive approach (HB2) needs to compute the bound. The other columns show how slower each method is, and the last column shows the maximum violation of the approximate primal solution recovered by the volume algorithm.

Dataset	HB2(s)	HB1	LR1	LR2	ACG1	ACG2	JD	HB2 viol
G30	0.18	2.0	8.9	3.1	9.6	9.3	8.0	0.032
G30b	0.20	1.3	11.9	3.8	7.1	6.7	8.2	0.049
G53	1.60	0.9	4.6	1.2	3.5	3.5	5.7	0.016
G57	7.60	2.2	5.8	2.3	3.8	3.9	3.4	0.023
G62	0.55	1.1	7.2	2.6	15.3	12.6	681.0	0.029
G69	2.43	2.2	9.6	2.4	12.4	14.5	79.9	0.025
G72	15.77	2.3	6.9	1.6	12.0	11.1	18.4	0.031

Table 1: Computational performance of different implementations.

Table 2 shows the median gap computed by CPLEX12.2 and our branch-and-price heuristic (BPH). We used the instances from [3] modified as follows. First, item demands were replicated to make each problem 60 periods long. Second, the capacity constraints were tightened by reducing the period capacity to 95% of its nominal value. Finally, problems in which CPLEX could find a solution immediately were deemed too easy were excluded from the dataset.

The algorithm has been assessed extensively against other competing approaches, such as those at [3], [5] and [6]. For the sake of brevity we list here the results against CPLEX 12.2, which outperforms all other approaches. The performance of BPH is much better at the root node, while after 150 seconds the two approaches produce similar results. It is worth noting that BPH delivers almost always a better lower bound than CPLEX, which could not find a feasible solution at the root node for two instances.

5 Conclusions

We develop a novel branch-and-price heuristic that finds good feasible solutions, and is particularly effective to long-horizon problems. A tight lower bound is efficiently

Method	Root	100s	150s
CPLEX 12.2	7.069	1.85	1.72
BPH	3.634	1.78	1.77

Table 2: Median Gap (%) - 30 instances.

calculated with a hybrid column generation/lagrange relaxation scheme and is used to assess the primal solution quality. Computational results show that our approach outperforms other recent methods and compares favorably with state-of-the-art commercial software.

References

[1] Barahona, F., R. Anhbil. The volume algorithm: producing primal solutions with a subgradient method, Math. Programming, 83, 101-111 (1998)

[2] Danna, E., E. Rothberg, C. Le Pape. Exploring relaxation induced neighborhoods to improve MIP solutions. Math. Programming Ser. A, 102 71-90 (2005)

[3] Trigeiro, W., L. J., Thomas, J. O., McClain. Capacitated lot sizing with set-up times. Management Science. 35 (3), 353-366, (1989)

[4] Jans, R., Z., Degraeve. Improved lower bounds for the capacitated lot sizing problem with setup times. Operations Research Letters. 32, 185-195, (2004)

[5] Sural, H., M., Denizel. L. N., Van Wassenhove. Lagrangean relaxation based heuristic for lot sizing with setup times. European Journal of Operational Research 195, 51-63, (2009)

[6] Bilde, O., J., Krarup. Sharp Lower Bounds and Efficient Algorithms for the Simple Plant Location Problem. Ann. Discrete Math.1, 79-88, (1977).

[7] Pimentel, C.M.O., F.P. Alvelos, J.M. Valrio de Carvalho. Comparing danzig-wolfe decompositions and branch-and-price algorithms for the multi-item capacitated lot sizing problem. Opt. Meth. Soft. 25 299-319, (2009)

[8] Huisman, D., R. Jans, M. Peeters, A.P.M. Wagelmans. Combining column generation and lagrangian relaxation. G. Desaulniers, J. Desrosiers, M. Solomon, eds. Column Generation. Springer, New York, 247-270 (2005)

Dynamic Probabilistic Lot Sizing with Service Level Constraints

Saumya Goel and Simge Küçükyavuz
The Ohio State University
{kucukyavuz.2, goel.43}@osu.edu

1 Introduction

Many inventory control models rely on the assumption that demand is known for all successive time periods apriori with certainty. This is a very restrictive assumption because the future demand can be influenced by many factors, most of which cannot be quantified ahead of time, like recession, inflation etc.

Recently, Beraldi and Ruszczyński [1] studied a static, probabilistic version of the lot sizing problem with a target service level and gave a branch-and-bound algorithm for solving it. Demand is assumed to follow a finite and discrete distribution and the order schedule is determined at the beginning of the planning horizon. This is a restrictive model because it assumes that the order schedule can not be updated during the planning horizon based on how the demands unfold over time. In reality, many order schedules have the flexibility that the order level can be updated depending on the demands observed in the past periods. Dynamic lot-sizing models address this flexibility.

In this paper, we consider an uncapacitated single product multi-stage probabilistic lot sizing problem in a dynamic setting. That is, one need not be bound by future decisions that are made at the start of the planning horizon, because the order schedule can be updated in each time period. Based on what demands we have seen in the previous time periods, we determine a order schedule that minimizes the total expected cost while satisfying both the flow balance and service level constraints.

We evaluate the overall probability of stocking out over the horizon, instead of maintaining the service level for each period individually. In other words, we consider joint chance constraints instead of the much easier case of individual chance constraints which call for quantile-based linear reformulations. Although joint chance constraints are more difficult, they are more appropriate because they ensure a high service level over the entire planning horizon.

The optimal solutions for Dynamic Probabilistic Lot-Sizing (DPLS) models are more flexible than those for Static Probabilistic Lot-Sizing (SPLS) models. This can

18

be attributed to the fact that SPLS is a restricted version of DPLS and the feasible region for SPLS is contained in the feasible region for DPLS. Although DPLS problems are more difficult than SPLS problems in terms of computational time, there is value gained in making the system adaptive. However, there might be certain applications where this flexibility is not possible and in those cases the more expensive but simpler SPLS models are relevant. Bookbinder and Tan [2] present a hybrid "static-dynamic" approach which uses a heuristic based algorithm to yield approximate results.

2   Stochastic Programming Model

In DPLS, the order decision in period t takes the observed demands and costs in periods $1, \ldots, t-1$ into account. This gives rise to what we refer to as the *dynamic* probabilistic lot-sizing problem (DPLS). Let Γ_{t-1}, be the random vector representing the demands and costs in periods $1, \ldots, t-1$, for $t \in [2, n]$, where n is the length of the finite planning horizon. Let ξ_t be the random variable representing the cumulative demand until time t. Let $x_t(\Gamma_{t-1})$ be the decision variable at stage $t \in [2, n]$, whose value is determined after the random variables, Γ_{t-1}, are observed, and x_1 be the initial order quantity. Then the service level constraint can be modeled with the chance constraint:

$$P \begin{pmatrix} x_1 & \geq & \xi_1 \\ x_1 + x_2(\Gamma_1) & \geq & \xi_2 \\ x_1 + x_2(\Gamma_1) + x_3(\Gamma_2) & \geq & \xi_3 \\ \vdots & \vdots & \vdots \\ x_1 + x_2(\Gamma_1) + x_3(\Gamma_2) + \cdots + x_n(\Gamma_{n-1}) & \geq & \xi_n \end{pmatrix} \geq \tau,$$

where τ represents the threshold probability of meeting the demand on time over the planning horizon.

The deterministic equivalent of the stochastic program for DPLS contains the so-called *mixing set* as its substructure, for which strong valid inequalities are proposed in [3, 5, 4]. We implement a branch-and-cut algorithm to solve the deterministic equivalent of DPLS. Our computational experiments illustrate that mixing cuts are effective in solving the DPLS problems.

3   Numerical Example

In order to highlight the difference between these static and dynamic PLS models, let us consider a small test case with 5 time periods and 5 scenarios as given in Figure 1. Let $\nu(i, j)$ represent the node of the scenario tree for period i and scenario j. Each outcome of the random variables is represented by a scenario path and

19

non-anticipative nodes are circled in a group. That is, $\nu(1,1) = \nu(1,2) = \cdots = \nu(1,5)$ (because at the beginning of the first period we did not see any demand), $\nu(2,1) = \nu(2,2); \nu(2,3) = \nu(2,4); \nu(3,3) = \nu(3,4)$. Let the probability of occurrence of each scenario be $\pi = (0.30, 0.10, 0.05, 0.20, 0.35)$. The values in nodes represent the demand, order cost respectively at that time in that scenario, for example demand at time 2 in scenario 1 is 73 units with unit order cost of 48. Let the holding cost be 10 percent of the order cost and the target service level τ be 85 percent.
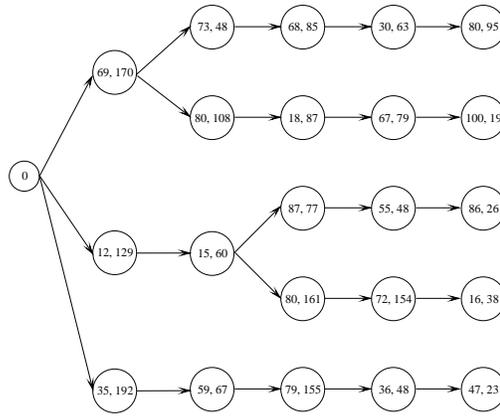
Figure 1: Scenario tree representation

We solved this small example with the SPLS and DPLS models. We report the optimal order quantities for SPLS and DPLS models in Table 1. The optimal cost given by SPLS model is 29493.3 which is higher than the optimal cost given by the DPLS model, 24095.5. As a result, significant cost savings are achieved when order quantities are determined based on the demand history. In contrast, the SPLS model is a restrictive version of DPLS model where the order quantities are decided ahead of time and are independent of the scenario path realized. The observed service level in SPLS model is 100% which is much higher than target service level of 85% whereas in DPLS models, the observed service level is 90%.

Table 1: Quantities produced in DPLS and SPLS models, $z^* = \{0,1,1,0,0\}$

Time	DPLS					SPLS
	Scenario					
	1	2	3	4	5	
1	69	69	69	69	69	69
2	80	80	110	110	104	141
3	61	0	0	0	0	0
4	110	0	0	0	36	30
5	0	185	76	16	47	94

4   Acknowledgment

References

[1] Beraldi, P. and Ruszczyński, A., A branch and bound method for stochastic integer programs under probabilistic constraints, *Optimization Methods and Software*, 17:359–382, 2002.

[2] Bookbinder, J. and Tan, J., Strategies for the probabilistic lot-sizing problem with service level constraints, *Management Science*, 34:1096–1108, 1988.

[3] Günlük, O. and Pochet, Y, Mixing mixed-integer inequalities, *Mathematical Programming*, 90:429–457, 2001.

[4] Küçükyavuz, S., On mixing sets arising in chance-constrained programming, to appear in *Mathematical Programming*, 2010.

[5] Luedtke, J., Ahmed, S., and Nemhauser, G., An integer programming approach for linear programs with probabilistic constraints, *Mathematical Programming*, 122:247–272, 2010.

Improvements of an Integrated Approach for Lot Sizing and Detailed Scheduling

Edwin David Gomez Urrutia Public Research Centre Henri Tudor; École Nationale Supérieure des Mines de Saint-Étienne
edwin.gomez@tudor.lu; gomez-urrutia@emse.fr

Riad Aggoune Public Research Centre Henri Tudor
riad.aggoune@tudor.lu

Stéphane Dauzère-Pérès École Nationale Supérieure des Mines de Saint-Étienne
dauzere-peres@emse.fr

1   Introduction

Traditionally, lot-sizing and scheduling decisions are taken separately, following a natural hierarchical order which consists in first defining the production targets at the tactical level, and then determining resource sequencing to realize the production plan. Because mathematical lot-sizing models include aggregate capacity constraints, there is no guarantee that the proposed production plan is feasible at the operational level. Integrated approaches have been developed to provide feasible production plans. Dauzère-Pérès and Lasserre [2] and Ouenniche *et al.* [4] study the impact of sequencing decisions on the multi-item lot-sizing problem. Stadtler [5] studies the multi-level single-machine Proportional Lot Sizing and Scheduling Problem (PLSP) with zero lead times, incorporating period overlapping setup times and batch size constraints. Li and Ierapetritou [3] propose a rolling horizon method with production capacity consideration. We worked on the approach proposed in Wolosewicz *et al.* [7] (see also Aggoune *et al.* [1]) which includes a Lagrangian heuristic to determine a feasible production plan and a Tabu search procedure to improve the plan. Our work consists in improving the efficiency of the approach through several modifications performed in different parts.

2   Integrated approach

The scheduling problem is represented by a conjunctive graph, where nodes correspond to operations and arcs to precedence constraints between two operations (in the routing of an item or on a resource in the sequence). In order to meet deadlines, the last operation of each path must be completed before its due date. Hence, the sum of processing and setup times of all operations in a path must not exceed the due date of the last operation of this path. And this must be true for all paths of the graph. These constraints can be seen as capacity constraints.

This integrated approach is composed of two principal modules. The first one looks for the best feasible production plan associated to a fixed sequence, and the second one aims at modifying this sequence to improve it. The new sequence is then taken by the first module and the procedure continues iteratively.

The first module is implemented through a Lagrangian relaxation, which aims at decomposing an optimization problem into a number of easy-to-solve subproblems dualizing capacity constraints. The goal of the Lagrangian relaxation is to give an optimal production plan without considering capacity constraints, using the algorithm proposed in [6]. Since the production quantities are not necessarily feasible, a smoothing procedure is then implemented in order to satisfy capacity constraints associated to a given sequence. The Lagrangian relaxation and the smoothing procedure are realized iteratively in the Lagrangian heuristic, to find the best feasible production plan for a fixed sequence.

Tabu search is used to implement the second module. Here, the fixed sequence is modified by choosing an arc to be swapped, using the Lagrangian relaxation information.

3   Improvements

We performed different modifications of the former approach, and several ideas were tested. In this abstract, we present the most relevant modifications provided to our method. These modifications and others, as well as some discarded ideas will be presented in the workshop, as well as numerical results that are significantly better than the ones obtained with the previous approach.

3.1   Lagrangian heuristic

In [7], Lagrangian relaxation could be applied twice at each iteration of the tabu search. The first Lagrangian relaxation computes only the lower bound. If the resulting lower bound is smaller than the best lower bound determined so far, then a second Lagrangian relaxation is run with the smoothing procedure to determine an upper bound. In the latter case, since it is rather time-consuming, the smoothing procedure

is only run every five iterations. After some analysis, we observed that obtaining a good upper bound with the smoothing procedure was not related to the quality of the lower bound obtained by the Lagrangian relaxation. Thus, the previous strategy led to large losses in resolution time and solution quality. In addition, performing the smoothing procedure every five iterations does not guarantee that the best possible feasible production plan will be found.

Therefore, we first decided to always apply the Lagrangian relaxation combined with the smoothing procedure. The following most important improvement consists in defining when applying the smoothing procedure. From our analysis, we observed that the lower bound without Lagrangian costs can help in choosing when applying the smoothing procedure to make a production plan feasible. Hence, to reduce computational times, we decided to perform the smoothing procedure only when the lower bound without Lagrangian costs was lower than the best upper bound. This can be explained by the fact that, to make a production plan feasible, the solution will usually be degraded, thus increasing the total cost. Moreover, to avoid applying the smoothing procedure too often, it is not run if the lower bound without Lagrangian costs is equal to one of the lower bounds found in the last five iterations. Large computational times are saved using these modifications, helping the tabu search to explore more changes in the sequence of operations.

3.2   Tabu search

The original procedure was taking the arc with the largest sum of Lagrangian multipliers. As we will show in the workshop, this methodology was not always successful, because the fact that a path is violated does not guarantee that all its arcs are interesting to be swapped. It seems that the impact of swapping an arc belonging to a large number of paths is greater than the impact of swapping an arc that belongs to a smaller number of paths (i.e. with a small sum of Lagrangian multipliers). However, swapping an arc can also have some very negative effect. Therefore, we decided to study the use of various neighborhood sizes. The idea is to consider a subset of critical arcs and not only one, to apply the Lagrangian heuristic after swapping independently each arc, and to keep the one which generates the best upper bound.

4   Conclusions and perspectives

A novel approach was previously proposed to solve a general lot-sizing and scheduling problem. Multiple modifications realized at different levels of the approach allowed significant improvements on the speed and the quality of the results. We are currently working on extending the approach to consider multi-level lot sizing and additional constraints.

References

[1] Aggoune R., Dauzère-Pérès, S., and Wolosewicz, C., A solving procedure for a general integrated lot-sizing and scheduling problem, International Workshop on Lot Sizing, Gardanne, France (2010)

[2] Dauzère-Pérès, S. and Lasserre, J.B., Integration of lotsizing and scheduling decisions in a job-shop, European Journal of Operational Research, 75 (2), 413-426 (1994)

[3] Li, Z. and Ierapetritou, M. G., Rolling horizon based planning and scheduling integration with production capacity considerations, Chemical Engineering Science, 65, 5887-5900 (2010)

[4] Ouenniche, J., Boctor, F. F. and Martel, A., The impact of sequencing decisions on multi-item lot sizing and scheduling in flow shops, International Journal Production Research, 37 (10), 2253-2270 (1990)

[5] Stadtler, H., Multi-level single-machine lot-sizing and scheduling with zero lead times, European Journal of Operational Research, 209, 241-252 (2011)

[6] Wagelmans, A., Van Hoesel, S., and Kolen, A., Economic Lot Sizing: an $O(nlogn)$ algorithm that runs in linear time in the Wagner-Within case, Operations Research, 40 (1), (1992)

[7] Wolosewicz, C., Dauzère-Pérès, S. and Aggoune, R., A new approach for solving an integrated planning and scheduling problem. INCOM'06 (12th IFAC Symposium on Information control Problems in Manufacturing), Vol 3, pp. 191-196, May 2006, Saint Etienne, France.

Hybrid Approaches for the Capacitated Lot Sizing Problem with Setup Carryover and Backordering

Hacer Guner Goren Department of Industrial Engineering, Dokuz Eylul University, Izmir, Turkey
hacer.guner@deu.edu.tr

Semra Tunali Department of Business Administration, Izmir University of Economics, Izmir, Turkey
semra.tunali@izmirekonomi.edu.tr

Raf Jans Department of Logistics and Operations Management, Montreal, Canada
raf.jans@hec

1   Problem Statement

The Capacitated Lot Sizing Problem (CLSP) consists of determining the production quantity and timing for several items on a single facility over a finite number of periods so that the demand and capacity constraints can be satisfied at a minimum cost. Besides lot sizing decisions, in recent years this problem has been extended to include semi-sequencing (i.e. the first and last product produced in a period) of the products produced in a period on a single machine. This problem which is known as Capacitated Lot Sizing Problem with Setup Carryover (CLSPC) involves producing more than one product per period but with at most one setup carryover activity per period. Another extension to this problem is to permit backordering by allowing the unsatisfied part of the demand to be satisfied in the following periods in the planning horizon. The resulting problem is called the Capacitated Lot Sizing Problem with setup carryover and backordering (CLSPCB). Unlike the CLSP which is extensively studied in the relevant research, the number of studies addressing the solution of the CLSPCB is very limited. During the survey of current relevant literature, we noted only a few studies focusing on both setup carryover and backordering, abbreviated to CLSPCB ([2], [3], [5]). Karimi and Ghomi [2] propose a greedy heuristic consisting of four stages to deal with this problem. In another study Karimi et al. ([3]), the authors propose a Tabu Search (TS) based approach and use this greedy heuristic

to find a feasible initial solution. The problem is extended to the parallel machine environment by a real life example in Quadt and Kuhn [5].

Considering the perceived research gap in this area, in this study a number of Genetic Algorithm (GA)-based heuristic approaches is proposed for solving the CLSPCB. Although GAs have been applied to many different lot sizing problems ([1]), to the best of our knowledge there is no study employing GA-based approaches to solve this problem. It is well known that GAs are good at finding the promising regions for global optima, however they are not good at locating the minimum of these optima in a large solution space. To overcome this difficulty, we propose to hybridize the GA with the Fix-and-Optimize heuristic and improve its performance in solving the CLSPCB.

2   Overview of the proposed hybrid approaches

In addressing the solution of the CLSPCB, we proposed two classes of hybrid approaches. The first class is named as sequential hybrid approaches whereas the second is named as embedded hybrid approaches. Both classes combine GAs and a MIP based heuristic, namely the Fix-and-Optimize heuristic with different hybridization schemes.

The main idea in the sequential hybrid approaches is to run GAs for a predetermined number of generations and use the overall best solution obtained as the initial solution for the Fix-and-Optimize heuristic. The best solution coming from GAs is improved throughout the iterations in the Fix-and-Optimize heuristic. In the embedded hybrid approaches, the Fix-and-Optimize heuristic is embedded into the loop of GAs. After a new population is formed, a solution is chosen randomly from the new population and it is set as the initial solution in the Fix-and-Optimize heuristic. The motivation behind this hybridization is to refine the solution quality of GAs in each generation. We hope that the Fix-and-Optimize heuristic will help GAs to direct the search towards the regions in the search space where good solutions exist.

To form the problems used in the Fix-and-Optimize heuristic, time and product decompositions are used. These two decomposition schemes are employed in four different ways to form eight hybrid approaches presented in Table 1.

3   Computational Experiments

To test the performance of the proposed hybrid approaches to solve the CLSPCB, the problem instances in Trigeiro et al. [6] were modified to introduce backordering and backordering cost was defined as a linear function of the holding cost (b=fh), where f=2 as in Millar and Yang [3]. Six problem classes which have five instances in each were used to evaluate the performance of the proposed hybrid approaches.

Table 1: Summary of the proposed hybrid approaches

	Definition of the hybrid approach
Sequential Hybrid Approach 1 (H1)	GAs first and then the Fix-and-Optimize heuristic with time decomposition only.
Sequential Hybrid Approach 2 (H2)	GAs first and then the Fix-and-Optimize heuristic with product decomposition only.
Sequential Hybrid Approach 3 (H3)	GAs first and then the Fix-and-Optimize heuristic with time decomposition first, then product decomposition.
Sequential Hybrid Approach 4 (H4)	GAs first and then the Fix-and-Optimize heuristic with product decomposition first, then time decomposition.
Embedded Hybrid Approach 5 (H5)	The Fix-and-Optimize heuristic with time decomposition in the loop of GAs.
Embedded Hybrid Approach 6 (H6)	The Fix-and-Optimize heuristic with product decomposition in the loop of GAs.
Embedded Hybrid Approach 7 (H7)	The Fix-and-Optimize heuristic with time decomposition in one generation and product decomposition in another generation in the loop of GAs.
Embedded Hybrid Approach 8 (H8)	The Fix-and-Optimize heuristic with two decomposition schemes in the sequence of product and time decomposition in the loop of GAs.

Since there is no benchmark study to which we can compare our results, therefore the lower bounds obtained from the Simple Plant Location formulation was used to measure the solution quality.

The parameter settings of proposed approaches were done through some pilot experiments on a medium size problem with 10 products and 20 periods. Based on these experiments, population size of 100, crossover rate of 0.95 and mutation rate of 0.005 were used for 100 generations.

The results obtained by using the proposed approaches show that all proposed hybrid approaches work very well to improve the solution quality across all problem classes. This can be attributed to the success of using the Fix-and-Optimize heuristic in the proposed hybrid approaches. Based on the results, sequential proposed hybrid approaches have better performances than the embedded hybrid approaches in most of the problem classes. Embedded hybrid approaches are better in small size problems where the performance of the sequential hybrid approaches is remarkable in medium and large size problems.

Another important observation is that the type of decomposition has an important effect on the performance of the Fix-and-Optimize heuristic. The hybrid approaches with time decomposition (i.e. H1, H3, H4) show better performances than the approaches with product decomposition (i.e. H2 and H6). Furthermore, using two decomposition schemes sequentially (i.e. H3, H4, H8) improves the solution quality more than using one decomposition scheme.

4   Conclusion

Lot sizing is one of the most well-known optimization problem in production planning. Most of the previous studies in this field focus on solving the CLSP which is known to be NP-Hard. Therefore, it is unlikely to find optimal solutions for realistically large problem instances of the CLSPC in a reasonable computational time. The trend in recent years is to employ computationally efficient solution techniques such as the mathematical programming based heuristics, meta-heuristics and greedy heuristics to deal with the combinatorial nature of the problem.

Unlike all earlier studies employing GAs to solve the CLSP, this study is the first one proposing novel hybrid GA approaches to solve the CLSPCB. These proposed hybrid approaches combine GAs with a MIP based heuristic, namely, Fix-and-Optimize heuristic. In terms of solution quality, promising results were obtained by the proposed hybrid approaches for the problem.

References

[1] Goren Guner H. Tunali S., Jans R., A Review of applications of genetic algorithms in lot sizing, Journal of Intelligent Manufacturing, 21 (4), 575-590 (2010).

[2] Karimi B., Ghomi Fatemi., A new heuristic for the CLSP problem with backlogging and set-up carryover. International Journal of Advanced Manufacturing Systems 5, 2, 66-77 (2002).

[3] Karimi B.,Ghomi Fatemi SMT, Wilson JM., A tabu search heuristic for the CLSP with backlogging and set-up carryover, Journal of the Operational Research Society 57, 140-157 (2006).

[4] Millar H., Yang M., Lagrangian Heuristics for the capacitated multi-item lot-sizing problem with back-ordering. International Journal of Production Economics, 34, 1-15 (1994).

[5] Quadt D., Kuhn H., Capacitated lot-sizing and scheduling with parallel machines, back-orders, and setup carry-over, Naval Research Logistics 56 (4), 366-384 (2009).

[6] Trigeiro W., Thomas L.J., McClain J.O., Capacitated Lot Sizing with setup times, Management Science, 35, 353-366 (1989).

The block planning approach for continuous time based lot sizing and scheduling: A change of paradigm?

Hans-Otto Guenther TU Berlin, Production Management H95, 10623 Berlin, Germany, Strasse des 17.Juni 135,
email: hans-otto.guenther@tu-berlin.de

1   Introduction

Observation: Rapidly changing business conditions impose new challenges on the design and industrial application of lot sizing models.

There are several observations which give reasons to reconsider the current principles of lot size modelling and call for a change of paradigm.

- First, customers in many industries are seeking faster replenishment and shortened cycle times in order to reduce their inventories and their investment in storage facilities. As a consequence, manufacturers are often forced to shift part of their production system from make-to-stock to make-to-order.

- Second, shorter product lifecycles and mass customization lead to a steadily in-creasing complexity of production systems. As a result, the size of the lot sizing model is considerably increased.

- Third, especially in process-related industries, there is often a natural sequence in which the various products are to be produced in order to minimize total changeover time and to maintain product quality standards. Hence, families of products can be identified which are produced in a given sequence under the same basic equipment setup.

- Fourth, since the development of the first dynamic lot sizing models production speed in almost all industries has considerably increased due to rapidly progressing technical advancements. This in combination with the increased number of product variants makes it necessary to base discrete lot sizing models on an accordingly shorter period length which in turn causes a significant increase in the number of variables and constraints.

- Finally, many companies shifted their production control philosophy from push, e.g. the classic material requirements (MRP) planning, to pull systems. Consequently, forecast-driven advanced planning of production runs has been replaced by short-term creation of production schedules which are most often driven by call orders from contract customers.

Since in our view conventional lot sizing and scheduling models do not sufficiently reflect the conditions given in industrial production systems we propose an alternate approach, called block planning, for scheduling production orders on a continuous time scale with demand elements being assigned to distinct delivery dates.

Assessment: Current lot sizing models do not adequately reflect the change in business conditions and technology.

For lot size modelling, three categories of model formulations can be identified.

- The first category of lot sizing models subdivides the entire planning horizon into discrete periods, usually of equal length, and determines setup decisions, lot sizes and inventory levels for each product and period. Two variants of this modelling approach exist. Big-bucket models assume a basic period length which is sufficient to schedule several production lots per period. The main difficulty associated with this approach is that the sequencing and timing of the production runs within a period and the possible carryover of the setup state between periods are not explicitly modelled. In contrast, small-bucket models attempt to integrate lot sizing and scheduling by allowing one or at most two products to be scheduled per period and to carry over the setup state from period to period. Irrespective of the granularity of the underlying time grid, in discrete time based lot sizing models the start and end of production runs as well as the updates of the inventory status are restricted by the period boundaries. Clearly, the accuracy with which the time representation is modelled depends on the relative length of the time periods.

- The second category of hybrid lot sizing models combines a discrete time scale for modelling the production runs of product families and a continuous time scale for scheduling the individual product variants within a period. For this purpose, macro-periods are defined which are divided into a fixed number of non-overlapping micro-periods with variable length. This modelling approach can be regarded as more realistic compared to purely discrete lot sizing models. But still the computational burden associated with solving real-life problem instances can be prohibitive.

- The third and last category of lot sizing models uses a continuous time representation for modelling the production activities. In this regard it also combines is-sues of lot sizing and scheduling in a realistic way.

32

Objective: Minimization of setup and holding costs has to be reconsidered.

Irrespective of the specific representation of time, lot sizing models in literature are based on the same paradigm of balancing the trade-off between setup costs which are incurred whenever a production run for a product is started and inventory holding costs charged for production in advance of demand. In contrast, scheduling models usually aim at achieving time targets and avoiding delays in the completion of the production schedule. For several reasons we found it difficult to employ conventional lot sizing approaches for scheduling production activities in a number industries.

- First, the usual assignment of setup costs and times to products does not realistically reflect the changeover processes prevalent in advanced manufacturing technology. In a great number of industrial settings, we observed that setup conditions are related to the processing mode of the production equipment rather than to individual product types. Hence, the common assignment of setup costs and times to individual products appears to be questionable since setup costs are often caused by changing the basic processing mode and not for switching be-tween different product types.

- Second, in many industrial applications setup costs are defined as opportunity costs to compensate for the unproductive times during the change of the setup state. Opportunity costs, however, depend on the utilization rate of the equipment and the profitability of the production facility. Clearly, these costs are only essential in bottleneck situations and even then impossible to measure. Despite this obvious interrelation, lot sizing models known from the literature typically assume given values of setup costs.

- Third, in supply chain management attention has shifted towards improved logistical performance. Thus finished product inventories are merely regarded as buffers between the manufacturing and the distribution stage of the supply chain and costs for the deployment of the finished goods to the warehouses in the sup-ply chain often dominate capital-oriented inventory holding costs.

From this discussion the conclusion can be drawn that minimizing lot size dependent setup and holding costs is only appropriate if these costs can be determined as out-of-pocket costs directly assignable to individual product types. In the absence of "true" cost figures, minimizing the makespan, i.e. the time span needed to complete a given portfolio of demand elements and thus minimizing setup times, seems to be more appropriate. Another major advantage of the makespan objective is that production re-sources are freed as soon as possible so that additional not yet known customer demand can be integrated into the production schedule.

Resolution: An alternate approach, called block planning, is proposed for scheduling production orders on a continuous time scale with demand elements being assigned to distinct delivery dates.

In many industries, e.g. in the consumer goods industry, production systems usually consist of a single bottleneck stage after which final products are packed and shipped to distribution centres or individual customers. For this type of application environment the block planning principle is proposed which models lot sizing and scheduling on a continuous time scale. Major characteristics of the block planning concept can be summarized as follows.

- Given the assignment of products to setup families, fixed setup sequences of products within a family are defined based on human expertise and technological requirements. Each block corresponds to a single setup family.

- The assignment of setup families to blocks is modelled by use of binary decision variables and determined by the optimization model based on the size and timing of demand and capacity considerations.

- The composition of blocks is not necessarily the same. Binary decision variables indicate whether a product is set up or not and continuous decision variables reflect the lot size of each product in the block.

- The start-off and completion times of a block are not directly linked to the period boundaries but can be scheduled flexibly on the continuous time scale. For each block a time window is imposed which defines the earliest possible start and the latest feasible completion time.

- A major setup operation is performed before starting or after completing a block while only a minor setup operation is required when changing between products within the same block.

Proposal: An MILP model for lot sizing and scheduling in a single-stage production system based on the block planning principle is developed. In order to provide increased flexibility for scheduling the production activities in face of the large product variety and to avoid that the start and the end of production runs are confined to the period boundaries an MILP model formulation for block planning based on a continuous time representation of time is proposed. This approach makes it unnecessary to use binary variables for the product-period assignments and the changeovers as in capacitated discrete time based lot sizing models. Decision variables refer to lot sizes, the major and minor setup activities, the assignment of product families to blocks, and the timing of the production runs. The objective function aims to minimize the makespan, i.e. to complete the entire production schedule as early as possible.

To examine the computational efficiency of the proposed block planning approach for production systems with a single bottleneck stage numerical experiments are conducted based on a case study from the beverage industry. In particular, it is shown that optimal solutions to problems of realistic size can be obtained within a few seconds of CPU time.

Capacitated Lotsizing and Scheduling: tackling sequence dependent setups using column generation

Luis Guimarães Faculdade de Engenharia - Universidade do Porto, Porto, Portugal
guimaraes.luis@fe.up.pt

Diego Klabjan Department of Industrial Engineering and Management Sciences,
Northwestern University, Evanston, IL, USA
d-klabjan@northwestern.edu

Bernardo Almada-Lobo Faculdade de Engenharia - Universidade do Porto, Porto,
Portugal
almada.lobo@fe.up.pt

1   Introduction

We present a solution procedure combining column generation and mathematical programming-based metaheuristics to solve the capacitated lotsizing and scheduling problem (CLSD). In many manufacturing environments, production planning problems involve the determination of production levels and sequences of different items on a single capacitated machine. Production levels are decided to satisfy deterministic demand over a planning horizon and production sequences must account for the sequence-dependent setup times and costs. The production plans are created with the objective of minimizing the overall costs consisting of holding and setup costs, while satisfying the available capacity and demand requirements. Examples of industries where lotsizing and scheduling must be simultaneously tackled are chemicals, drugs and pharmaceuticals, pulp and paper, textiles, foundries, glass container, food and beverage, and many others.

CLSD is considered a big-bucket model, since several setups are allowed to be performed per period, therefore sequencing decisions within each time period are difficult to model. Modeling approaches to capture sequencing decisions can be divided into product-related formulations ([3]) and sequence-related formulations ([2, 4]). In

product-related formulations sequences are defined explicitly by a mixed integer programming (MIP) model, while in sequence-related formulations the MIP model prescribes for each period a sequence from a set of pre-determined sequences. Sequence-related formulations for CLSD are easier to model and solve, but yield a major drawback coming from the fact that the number of binary decision variables grows exponentially with the number of products.

The computational intractability requires the use of efficient heuristics and meta-heuristics to provide good quality solutions to the CLSD, especially when considering large real-world instances. State-of-the-art optimization engines either fail to generate feasible solutions to this problem or take a prohibitively large amount of computation time, even for the single-machine setting. In [1] two neighborhood search algorithms, tabu and variable neighborhood search were developed, based on a product-related formulation. Mathematical programming-based heuristics to the parallel machine extension of CSLD (CLSD-PM), based on a product-related formulation, are also exhibited by the same authors in [3]. MIP-based construction and improvement heuristics, and also a MIP-based metaheuristic, are described there.

Relying on the sequence-related formulation in [2], the authors propose a fast-enumeration scheme of a branch-and-bound type that defines a solution to the CLSD considering efficient sequences. An efficient sequence is a sequence that for a given first and last product, and a set of produced products minimizes the setup cost. It follows from the assumption that setup costs are proportional to setup times. Although this reduction has a strong impact on the number of decision variables, the applicability of this approach is still limited to instances with relatively small number of items and/or short planning horizon. A different sequence-related MIP model and solution procedure to CLSD-PM is presented in [4]. It consists of dividing the entire production schedule into smaller production sequences, which the authors called split-sequences. For each period t the production sequence is composed of L_t split-sequences. To address the large number of split-sequences arising they propose a column generation based heuristic, where in each iteration the new split-sequences are obtained by an enumeration algorithm with an additional parameter $maxBR$, representing the maximum number of products in the split-sequence. Two different heuristics are proposed, one truncates the branch-and-bound search based on the number of fractional variables while the other one iteratively executes local search to improve an incumbent solution. A major disadvantage of this methodology is that to solve a given problem multiple runs are need with different values of L_t and $maxBR$.

2 Combining column generation and MIP-based heuristics

In this work we present a new solution approach to achieve high quality solutions for CLSD that combines column generation with MIP-based metaheuristics. It follows the excellent results obtained in [3]. We start by modeling a sequence-related formulation to CLSD, in which only product setup variables are binary. Nevertheless, the number of continuous decisions variables representing the sequence selected for each period and machine grows exponentially with the number of products. To tackle this issue column generation is incorporated in our approach.

We start by defining a restricted master problem which has only a small subset of the feasible production schedules in each period. In each iteration of the column generation method the linear relaxation of the master is solved to obtain a primal and dual solution. The dual solution is used to define the subproblems objective function (one for each time period). The subproblems are solved afterwards aiming to find columns that price out favorably to include in each time period. When it fails to identify negative reduced cost columns for any of the time periods, the optimal solution for the relaxed master is found and the column generation algorithm is stopped.

Since our goal is to obtain feasible integer solutions we introduce MIP-based metaheuristics. First we use a constructive MIP-based heuristic to generate a feasible integer solution to CLSD, followed by an improvement MIP-based heuristic to improve the current feasible solution, guided by an iterative local search procedure.

Next we describe the construction heuristic used to generate initial integer feasible solutions. These solutions are based on progressive interval heuristics, namely relax-and-fix (RF). RF heuristic obtains a solution to the original MIP problem by sequentially solving a series of partially relaxed MIP subproblems, therefore reducing the overall complexity. We decompose the original MIP through a time-stage partition, defining a rolling planning horizon. At each iteration of RF binary product setup decision variables can be divided into three different subsets (see Figure 1). In the first subset are setup variables from periods, whose value has been fixed in the previous iterations. The second subset regards setup variables belonging to periods where integrality is required in the next iteration, and the remaining is composed by setup variables of the following periods for which integrality is relaxed. This describes the MIP subproblem to be solved at each iteration. Furthermore, our construction heuristic uses the time decomposition to combine the different models to the problem. In the time periods colored in gray (see Figure 1) we apply the sequence-related formulation, while in the remaining time horizon a product-related formulation. Doing so, we first call our column generation algorithm generating new production sequences to the light gray time horizon. Then we solve the MIP over the current set of sequences to obtain a integer solution to those time periods, leaving the remaining time

horizon relaxed and with a product-related formulation. This enable us to solve a limited number of subproblems at each RF iteration while still having an estimation of future costs. At the end of the final iteration an integer solution to the problem is obtained prescribing a production sequence to each time period.
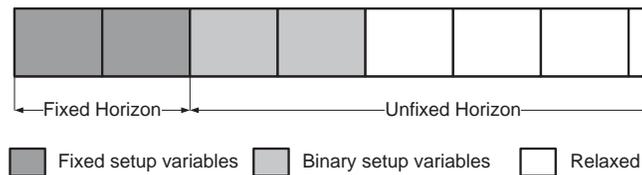
Figure 1: Construction heuristic for the CLSD

Following the creation of an initial solution to the problem we perform a neighborhood search to improve the solution quality. The new construction heuristic developed by the combination of RF principles with column generation represents an effort to build very good quality integer solution, yet the MIP-based construction heuristic is deterministic and can lead to local optima. To avoid entrapment, our MIP-based improvement heuristic works similarly to the improvement heuristic described in [3], but once again combined with column generation. Two different neighborhoods are defined.

The first neighborhood consists in selecting a joint subset of pairs of periods, for which setup decision variables are re-optimized using a sub-MIP, while the remaining setup decision variables are fixed. Note that the decisions on production quantities can be re-optimized for the entire planning horizon and for all machines. Column generation is used to discover new production sequences for the pairs of periods selected before solving the sub-MIP. Pairs of periods are selected according to their potential decrease in the objective function. We evaluate this potential by performing a limited number of iterations of our column generation algorithm to each pair. Based on this

evaluation we filter the pairs which can lead to greater reductions and randomly pick one as our next neighbor. A pair is not selected if it has already been used and the current objective value was obtained. If all pairs combinations have been tried and no improvement in the solution has been achieved, then we are in a local minima.

The second neighborhood is based on a product local search. Once again, we re-optimize setup decision variables, but this time for a subset of products over the entire planning horizon, calling first the column generation algorithm and solving the sub-MIP afterwards. Subsets of products are selected according to their correspondent probability. Initially the probability associated with each product is the same. However as products are selected, the probability is reduced. Probabilities are updated considering two factors: frequency and recency. The more frequently a product has been used, and the more recently it has been used, the lower the probability of selection.

The improvement heuristic starts with the period local search until local minima is achieved. We then call the product local search to escape from entrapment. When a new incumbent solution is found by this second neighborhood we restart the product local search. The improvement heuristic iterates until the user defined stooping criteria is met (maximum running time, number of neighbors explored in the product local search without improvement, or both).

3   Final notes

Preliminary computational tests performed on the single machine instances with capacity variation of [3] confirm the potential of the approach. Our heuristic yields an overall average deviation from lower bound of 1.06% on an average running time of 850.6s compared to 1.10% and 1870.1s reported.

Our contributions are as follows. We present a new approach to solve CLSD based on the combination of column generation and MIP-based heuristics and metaheuristics. A new sequence-related formulation and column generation procedure were developed for the problem. New construction and improvement heuristics embedding elements from column generation, MIP-based heuristics and combine product-related and sequence-related formulations in a single model. Related previous work on MIP-based heuristics is available in [3], but by considering a product related formulation. Column generation has also been applied to CLSD in [4], although without considering sequence-dependent setup times. Differences are also present in the column generation scheme, due to alternative formulation and solution to the arising sub-problems.

References

[1] B. Almada-Lobo and R. James. Neighbourhood search metaheuristics for capacitated lotsizing with sequence-dependent setups. *International Journal of Production Research*, 48:861–878, 2010.

[2] Knut Haase and Alf Kimms. Lot sizing and scheduling with sequence-dependent setup costs and times and efficient rescheduling opportunities. *International Journal of Production Economics*, 66(2):159 – 169, 2000.

[3] Ross J.W. James and Bernardo Almada-Lobo. Single and parallel machine capacitated lotsizing and scheduling: New iterative MIP-based neighborhood search heuristics. *Computers & Operations Research*, In Press, Corrected Proof:–, 2011.

[4] Sungmin Kang, Kavindra Malik, and L. Joseph Thomas. Lotsizing and scheduling on parallel machines with sequence-dependent setup costs. *Management Science*, 45(2):273–289, 1999.

Dynamic capacitated lot sizing with random demand subject to a backlog/waiting-time oriented δ-service level measure

Stefan Helber   Leibniz Universität Hannover
helber@prod.uni-hannover.de

Florian Sahling   Leibniz Universität Hannover
sahling@prod.uni-hannover.de

Katja Schimmelpfeng   Brandenburg University of Technology
katja.schimmelpfeng@tu-cottbus.de

1   Problem statement

We assume that a single production system or machine is required to produce K different products. This machine has a (regular) capacity b_t in each of the T discrete periods of the planning horizon. It can be extended by overtime at a cost oc per time unit. We do not consider the sequence of the products within any period. If a product k is produced during period t, i.e., with production quantity $q_{kt} > 0$, a setup time $ts_k \geq 0$ is required and a setup cost $sc_k \geq 0$ occurs. The processing time for a unit of product k is tb_k. The cost of holding one unit of physical inventory for one period is denoted as hc_k.

The demand of product k in period t is modeled as a random variable D_{kt} with a given probability distribution, given expected value $\mathrm{E}[D_{kt}]$ and variance $\mathrm{VAR}[D_{kt}]$. The demand for the product-period combination (k, t) is assumed to be independent from those for any other combination $(\hat{k}, \hat{t})$ with $k \neq \hat{k}$ and/or $t \neq \hat{t}$. Estimators of $\mathrm{E}[D_{kt}]$ and variance $\mathrm{VAR}[D_{kt}]$ are assumed to be provided by a forecasting system. For each product k, the total production $\sum_{t=1}^{T} q_{kt}$ over the entire planning horizon must at least be sufficient to meet the expected total demand $\sum_{t=1}^{T} \mathrm{E}[D_{kt}]$.

If in any period t the cumulated (random) demand $\sum_{\tau=1}^{t} D_{k\tau}$ of product k exceeds the cumulated (deterministic) production $\sum_{\tau=1}^{t} q_{k\tau}$, the unmet demand is backordered and a positive value of the (random) backlog

$$BL_{kt} = \max\left(0, \sum_{\tau=1}^{t}(D_{k\tau} - q_{k\tau})\right) \qquad (1)$$

occurs. The opposite case results in a positive value of the (random) physical inventory:

$$YP_{kt} = \max\left(0, \sum_{\tau=1}^{t}(q_{k\tau} - D_{k\tau})\right) \qquad (2)$$

Both the expected physical inventory $E[YP_{kt}]$ and the expected backlog $E[BL_{kt}]$ are non-linear functions of the cumulated production in periods 1 to t.

The new δ-service level measure for product k with random demand D_{kt}, deterministic production quantity q_{kt}, and random backlog $BL_{kt} = \max(0; \sum_{\tau=1}^{t}(D_{k\tau} - q_{k\tau}))$ is computed as follows based on expected values of the respective random variables:

$$\delta_k = 1 - \frac{\sum_{t\in\mathcal{T}} E[BL_{kt}]}{\sum_{t\in\mathcal{T}}(T - t + 1)E[D_{kt}]}. \qquad (3)$$

The research presented in our paper is most closely related to those presented in [2] and [3]. Instead of limiting backorders via a β-service-level constraint, we aim at limiting backlog and hence take the customer waiting time into account using the δ-service level.

2 The non-linear stochastic capacitated lot-sizing problem (SCLSP) with a δ-service-level constraint

Based on the assumptions in section 1, the δ-service level introduced in section **??**, and the notation in Table 1, we now state the SCLSP as follows:

SCLSP Model

$$\min Z = \sum_{k\in\mathcal{K}}\sum_{t\in\mathcal{T}} hc_k \cdot E[YP_{kt}] + \sum_{k\in\mathcal{K}}\sum_{t\in\mathcal{T}} sc_k \cdot x_{kt} + \sum_{t\in\mathcal{T}} oc \cdot o_t \qquad (4)$$

42

Table 1: Notation used for the SCLSP model

Indices and index sets:

$\mathcal{K}$ set of products ($k \in \{1, \ldots, K\}$)

$\mathcal{T}$ set of periods ($t \in \{1, \ldots, T\}$)

Deterministic parameters:

b_t available capacity in period t

δ_k minimum δ-service level of product k

hc_k holding cost of product k per unit and period

M big number

oc overtime cost per unit of overtime

sc_k setup cost of product k

tb_k production time per unit of product k

ts_k setup time of product k

Random variables:

BL_{kt} backlog of product k in period t

D_{kt} demand of product k in period t

Y_{kt} net inventory position of product k at the end of period t

YP_{kt} physical inventory of product k at the end of period t

Decision variables:

o_t overtime in period t

q_{kt} production quantity (lot size) of product k in period t

x_{kt} binary setup variable of product k in period t

subject to:

$$\sum_{k \in \mathcal{K}} (ts_k \cdot x_{kt} + tb_k \cdot q_{kt}) \leq b_t + o_t, \qquad \forall t \qquad (5)$$

$$q_{kt} - M \cdot x_{kt} \leq 0, \qquad \forall k, t \qquad (6)$$

$$Y_{k,t-1} + q_{kt} - Y_{kt} = D_{kt}, \qquad \forall k, t \qquad (7)$$

$$YP_{kt} = \max(0, Y_{kt}), \qquad \forall k, t \qquad (8)$$

$$BL_{kt} = \max(0, -Y_{kt}), \qquad \forall k, t \qquad (9)$$

$$\sum_{t \in \mathcal{T}} q_{kt} \geq \sum_{t \in \mathcal{T}} \mathrm{E}[D_{kt}], \qquad \forall k \qquad (10)$$

$$\sum_{t \in \mathcal{T}} \mathrm{E}[BL_{kt}] \leq (1 - \delta_k) \sum_{t \in \mathcal{T}} (T - t + 1)\mathrm{E}[D_{kt}], \qquad \forall k \qquad (11)$$

$$q_{kt} \geq 0, \qquad \forall k, t \qquad (12)$$

$$o_t \geq 0, \qquad \forall t \qquad (13)$$

$$x_{kt} \in \{0, 1\}, \qquad \forall k, t \qquad (14)$$

43

The objective (4) is to minimize the total expected costs of physical inventory, setups, and overtime. Constraints (5) guarantee that the capacity needed for setups and production does not exceed the sum of regular and overtime capacity. Constraints (6) enforce setups in production periods. The inventory balance equations (7) relate the (random) end-of-period inventory position Y_{kt} to the random demand, the production quantity and the inventory position from the previous period. In the following equations, (8) and (9), the physical inventory as well as the backlog are determined. Constraints (10) make sure that at least the expected cumulated demand of product k is produced until the last period T. Furthermore, constraints (11) ensure that the backlog does not exceed the target δ-service level.

Unfortunately, we are not aware of a method available to solve the SCLSP in the non-linear form presented above. For this reason we developed a numerically tractable model variant that approximates the non-linear functions of expected backlog and inventory by piecewise linear functions. See [1] for model details and algorithmic aspects.

References

[1] Helber, S., F. Sahling, K. Schimmelpfeng. 2011. Dynamic capacitated lot sizing with random demand and dynamic safety stocks. *Leibniz Universität Hannover, Wirtschaftswissenschaftliche Fakultät. Working paper. http://www.wiwi.uni-hannover.de/Forschung/Diskussionspapiere/dp-465.pdf* .

[2] Tempelmeier, H., S. Herpers. 2010. ABC$_\beta$ - a heuristic for dynamic capacitated lot sizing with random demand under a fillrate constraint. *International Journal of Production Research* **48**(17) 5181–5193.

[3] Tempelmeier, Horst. 2011. A column generation heuristic for dynamic capacitated lot sizing with random demand under a fill rate constraint. *Omega* doi: 10.1016/j.omega.2011.01.003.

The economic lot-sizing problem with an emission constraint

Mathijn Retel Helmrich Wilco van den Heuvel Albert P.M. Wagelmans Erasmus School of Economics, Erasmus University Rotterdam
retelhelmrich@ese.eur.nl

Raf Jans HEC Montréal

1 Introduction

In recent years, there has been a growing tendency to not only focus on costs in a production process, but also on its environmental implications. Particular interest is paid to emissions of pollutants, such as carbon dioxide. This shift towards a more environmentally friendly production process can be caused by legal restrictions or by a company's desire to pursue a 'greener' image by reducing its carbon footprint.

For these reasons, the classic economic lot-sizing model has been extended. In addition to the usual financial costs, there are emission 'costs' associated with production, keeping inventory and setting up the production process. Other interesting works that integrate carbon emission constraints in lot-sizing problems are [2] and [2]. The lot-sizing model that we consider minimises the (financial) costs under an emission constraint. This constraint can be seen as one global restriction over all periods. The model can be formally defined as follows:

$$\min \quad \sum_{t=1}^{T} \left(p_t(x_t) + h_t(I_t) \right) \tag{1}$$

$$
\begin{aligned}
\text{s.t.} \quad I_t &= I_{t-1} + x_t - d_t & t = 1, \ldots, T \tag{2} \\
I_0 &= 0 \tag{3} \\
x_t, I_t &\geq 0 & t = 1, \ldots, T \tag{4} \\
\sum_{t=1}^{T} \left(\hat{p}_t(x_t) + \hat{h}_t(I_t) \right) &\leq \hat{C} & , \tag{5}
\end{aligned}
$$

where p_t and h_t are production and holding costs, and $\hat{p}_t$ and $\hat{h}_t$ are production and holding emissions, respectively. We assume that all functions are concave, nondecreasing and nonnegative. This includes the well-known case with fixed set-up costs and linear production and holding costs. Of course, $\hat{p}_t$ and $\hat{h}_t$ don't necessarily refer to emissions. They can be any kind of costs other than those in the objective function.

2   Complexity results

With a reduction from KNAPSACK, we show that lot-sizing with an emission constraint is $\mathcal{NP}$-hard, even if only production emits pollutants and these production emissions are linear. As a consequence of the proof, we also find that lot-sizing *with two production modes* in each period is $\mathcal{NP}$-hard, even if only production emits pollutants (linearly) and either all (financial) costs or all emissions are time-invariant.

3   Structural properties

We show that, for (generalised) Wagner-Whitin (nonspeculative) costs and emissions, the single-sourcing property holds in all periods. In general, we show that the single-sourcing property holds in all but (at most) one period.

4   Algorithms

4.1   Lagrangean heuristic

We present a Lagrangean heuristic that dualises the emission capacity constraint (5). We solve this dual problem in $\mathcal{O}(T^4)$ time with Megiddo's method [3] applied to the Wagner-Whitin algorithm [4]. After solving, we obtain both a lower bound and a feasible solution.

4.2   Pseudo-polynomial algorithm for Wagner-Whitin costs and emissions

Assuming that all parameters are integer and all costs and emissions are Wagner-Whitin, we can exactly solve the problem in $\mathcal{O}(T^2 opt)$ time. Our algorithm minimises emissions given a (financial) budget $\text{\euro}$ in $\mathcal{O}(T^2\text{\euro})$. We try budget $\text{\euro} = 1, 2, 3, \ldots$ until the minimum emissions are within the emission capacity.

4.3 FPTAS for Wagner-Whitin costs and emissions

We turn the pseudo-polynomial algorithm into a fully polynomial-time approximation scheme (FPTAS) by letting our algorithm only consider budgets $\left(1 + \frac{\varepsilon}{(e-1)(T+1)}\right)^k$ for integer values of k. This leads to a running time of $\mathcal{O}(T^3 \ln(opt)/\varepsilon)$.

4.4 FPTAS for general costs and emissions

In the general case, there may be a period with two sources. We extend our algorithm by iterating over all possible 'double-sourcing' periods, and over all possible budgets \$ in the 'double-sourcing block', where \$ can only assume values of the type $(1+\varepsilon)^k$. This leads to a running time of $\mathcal{O}(T^4 \ln^2(opt)/\varepsilon^2)$.

4.5 Using a lower bound to speed up the FPTAS

The FPTASes can be sped up by using a lower bound, for instance the one obtained by the Lagrangean heuristic. The running time becomes $\mathcal{O}\left(T^3 \ln\left(\frac{opt}{LB}\right)/\varepsilon\right)$ for the FPTAS for Wagner-Whitin costs and emissions, and $\mathcal{O}\left(T^4 \ln^2\left(\frac{opt}{LB}\right)/\varepsilon^2\right)$ for the general FPTAS.

5 Computational tests

In order to gain insight into the performances of the different algorithms, we used them to solve 1800 randomly generated problem instances. A part of these instances were constructed such that their costs and emissions were nonspeculative, so that they could also be solved by the dedicated algorithms. Other instances had a certain degree of speculativity. Some of these instances represented a problem with two production modes, 'cheap & dirty' and 'expensive & clean'.

Because all test data sets had fixed plus linear costs and emissions, we were able to also solve them with CPLEX and compare the results. We used two different MIP formulations in CPLEX, both a 'natural' and a shortest path formulation.

References

[1] Saif Benjaafar, Yanzhi Li and Mark Daskin, Carbon Footprint and the Management of Supply Chains: Insights from Simple Models. Working paper, University of Minnesota, 2010.

[2] Nabil Absi, Stéphane Dauzère-Pérèz, Safia Kedad-Sidhoum, Bernard Penz and Christophe Rapine, Lot-sizing with carbon emission constraints. International Workshop on Lot Sizing 2010, Gardanne, France, August 25th – 27th, 2010.

[3] N. Megiddo, Combinatorial optimization with rational objective functions. Mathematics of Operations Research, 4, 414–424, 1979.

[4] H.M. Wagner and T.M. Whitin, Dynamic version of the economic lot size model. Management science, 5, 89–96, 1958.

A Polynomial Algorithm for the Multi-Stage Production-Capacitated Lot-Sizing Problem

Hark-Chin Hwang Department of Industrial Engineering, Chosun University 375 Seosuk-Dong, Dong-Gu, Gwangju 501-759, Korea
hchwang@chosun.ac.kr

Hyun-soo Ahn Department of Operations and Management Science, Ross School of Business, University of Michigan, Ann Arbor MI 48109
hsahn@bus.umich.edu

Philip Kaminsky Department of Industrial Engineering and Operations Research, University of California, Berkeley CA 94720
kaminsky@ieor.berkeley.edu

The multi-stage lot-sizing problem with production capacities (MLSP-PC) deals with a supply chain that consists of a manufacturer with stationary production capacity and intermediaries (distribution centers or wholesalers) and a retailer to face deterministic demand. An optimal supply chain plan for the MLSP-PC specifies when and how many units each organization of the supply chain has to produce or transport to ultimately fulfill the demand at the retailer with the objective of minimizing total supply chain cost. All the production, transportation and inventory holding costs in each organization are assumed to be concave.

The single-stage uncapacitated lot-sizing problem for a manufacturer was introduced by Wagner and Whitin [6] and the multi-stage version of the uncapacitated problem was solved by Zangwill [5]. To address the manufacturer's production capacitated situation, Florian and Klein [1] solved the single-stage capacitated lot-sizing problem. Optimal algorithms for the multi-stage problem with production capacity were first presented by Kaminsky and Simchi-Levi for the two-stage case (2LSP-PC) [2]. Van Hoesel et al. [3] generalized the 2LSP-PC to the multi-stage lot-sizing problem MLSP-PC.

For the multi-stage dynamic lot-sizing problem with production capacities, Van Hoesel et al. [3] provide an $O(LT^4 + T^7)$ algorithm when no speculative motive exits in transportation (we call it the MLSP-PC with *non-speculative cost structure*) where L is the number of stages in the supply chain and T is the length of the planning horizon. For the most general MLSP-PC problem with concave cost structure,

however, no polynomial time algorithm has been presented until now. Although the non-speculative cost structure explains quite well the value-adding phenomena from upstream to downstream operations in a supply chain, it has limitation in modeling the economies of scale in general, for instance, the quantity discount for large shipment in transportation industry. To address the economies of scale in a more realistic supply chain, we need to attack the MLSP-PC with concave costs in all the stages. The primary purpose of this paper is to provide a polynomial time algorithm for the MLSP-PC with concave costs both in the number of stages L and the length of horizon T.

Most dynamic lot-sizing problems are modeled as discrete-time dynamic programming, and are solved by iteratively enumerating over time periods. For instance, when solving the single-stage capacitated problem defined in Florian and Klein (1971), one needs to solve the optimality equation for each state (that is, cumulative production quantity) in a given period. Then, the same computations are repeated for each subsequent period to determine the optimal policy and the resultant production schedule. This is reflected by the fact that we often use time as a subscript of the value function. Indeed, van Hoesel et al. (2005) show that a traditional time-based enumeration solves the MLSP-PC with non-speculative transportation costs in polynomial time, using the fact that this multi-stage lot sizing problem with fixed-charge transportation and linear inventory costs is fully specified by characterizing manufacturing decisions. However, under a general concave cost structure, manufacturing decisions no longer characterize the entire plan. In order to solve the DP, we need to keep track of production and transportation decisions at all stages. Consequently, there is no polynomial algorithm that will solve this problem by performing recursive calculations over the time periods only.

On the other hand, in this paper, we propose a different way to conduct iterative computations to solve the MLSP-PC. Instead of iterating over time, we iterate along path in the two dimensional space of time and stage in the supply chain, which we call a *basis path*. Consequently, in contrast to every other lot-sizing DP that we are aware of, our algorithm requires us to in general iterate both forward and backward in time. We first present a DP algorithm for the case where a basis path given. The path contains *basis nodes* of pairs of stage and period over which the DP iterates. By exploiting the structures derived from consecutive basis nodes, we establish optimality equations with immediate costs to evaluate the value function for each basis node. This algorithm does not directly yield a polynomial algorithm since there are a large (indeed, exponential) number of basis paths. However, because the evaluation of the immediate cost at each basis node depends on its neighbor basis nodes not on the entire basis path, this allows for focusing on a sufficiently small set of possible basis paths, leading to a polynomial time algorithm that solves the MLSP-PC with general concave costs in $O(LT^{10})$ time. In addition to this effective solution methodology, we improve the algorithm to run in $O(LT^8)$ time by efficiently evaluating costs associated

with basis paths.

References

[1] Florian, M. and Klein, M. Deterministic Production Planning with Concave Costs and Capacity Constraints, *Management Science*, 18, 12−20 (1971).

[2] Kaminsky, P. and Simchi-Levi, D. Production and distribution lot sizing in a two stage supply chain. *IIE Transactions*, 35, 1065-1075 (2003).

[3] Van Hoesel, S., Romeijn, H.E., Romero Morales, D. and Wagelmans, A.P.M. Integrated lot-sizing in serial supply chains with production capacities. *Management Science*, 51, 1706-1719 (2005).

[4] Wagner, H.M. and Whitin, T.M. Dynamic version of the economic lot-size model. *Management Science*, 5, 89−96 (1958).

[5] Zangwill, W.I. A backlogging model and a multi-echelon model of a dynamic economic lot size transportation system−a network approach. *Management Science*, 15, 506−527 (1969).

The Economic Lot-Sizing Problem with Lost Sales and Bounded Inventory

Hark-Chin Hwang Department of Industrial Engineering, Chosun University, South Korea
email: hchwang@chosun.ac.kr

Wilco van den Heuvel Econometric Institute, Erasmus University Rotterdam, The Netherlands
e-mail: wvandenheuvel@ese.eur.nl

Albert P.M. Wagelmans Econometric Institute, Erasmus University Rotterdam, The Netherlands
e-mail: wagelmans@ese.eur.nl

1   Introduction

The single-item deterministic economic lot-sizing problem with lost sales and bounded inventory (ELS-LB) extends the classical uncapacitated lot-sizing problem of Wagner and Whitin ([6]), by allowing lost sales and introducing inventory bounds. Both production and allocation decisions must be made in each period to minimize the sum of production, inventory, and lost sales costs when the maximum inventory in each period (that is, the sum of initial inventory and production quantity) is bounded. In this talk we discuss a variant of this problem that is more general than the cases that have already been studied in the literature.

We assume that production cost functions are concave, reflecting economies of scale in production. The production cost function is said to have a fixed-charge cost structure if it is composed of a fixed setup cost and a variable (per-unit) cost of production. If the fixed-charge cost structure is such that the per-unit production cost of a period is not larger than the per-unit production cost plus the per-unit holding cost of the previous period, then we say that there are no speculative motives to hold inventory, and we call the cost structure non-speculative.

We assume a per-unit lost sales cost that may vary over time. These costs may model lost revenue (if the lost sales cost equals the selling price), or they may reflect loss of goodwill. Furthermore, we assume that customers are not willing to wait for

orders that arrive late, which means that backlogging is not allowed. Finally, note that lost sales and outsourcing are conceptually equivalent ([2]). This means that the lost sales models in this talk can also be interpreted as models in which there is an outsourcing option to fulfill demand.

The challenge of solving our variant of the ELS-LB problem lies in the fact that classical classes of policies do not necessarily produce optimal solutions. For example, in the zero-inventory-ordering (ZIO) policy production only occurs when the inventory level drops to zero. If the production cost is non-speculative or if there is no storage capacity, an optimal solution can be found by considering only this policy ([1], [2]), but this does not hold for the general ELS-LB problem. Another classical policy is first come first served (FCFS), in which demand is always satisfied if there is inventory on hand. When lost sales costs are non-increasing over time ([4], [5], [3]), it suffices to consider this policy. Again, however, the FCFS rule does not apply anymore for general lost sales costs. To the best of our knowledge, we are the first to study the general ELS-LB problem when both ZIO and FCFS policies do not apply.

2   Results

In this talk we discuss properties of optimal solutions for the general ELS-LB problem and we show that these properties can be used to develop a dynamic programming algorithm to solve the problem in $O(T^4)$ time, where T is the length of the planning horizon. When lost sales costs are non-increasing over time, our algorithm can be simplified, resulting in an $O(T^2)$ running time. Moreover, with the additional assumption that there are no speculative motives to hold inventory, we can derive a linear time algorithm, which improves upon a previous result by [3].

References

[1] D. Aksen, K. Altinkemer and S. Chand. The single-item lot-sizing problem with immediate lost sales. *European Journal of Operational Research*, 147:558–566, 2003.

[2] F. Chu and C. Chu. Single-item dynamic lot-sizing models with bounded inventory and outsourcing. *IEEE Tansactions on systems, man, and cybernetics - part A: systems and humans*, 38:70–77, 2008.

[3] X. Liu and Y. Tu. Production planning with limited inventory capacity and allowed stockout. *International Journal of Production Economics*, 111:180–191, 2008.

[4] R.A. Sandbothe and G.L. Thompson. A forward algorithm for the capacitated lot size model with stockouts. *Operations Research*, 38:474–486, 1990.

[5] R.A. Sandbothe and G.L. Thompson. Decision horizons for the capacitated lot size model with inventory bounds and stockouts. *Computers & Operations Research*, 20:455–465, 1993.

[6] H.M. Wagner and T.M. Whitin. Dynamic version of the economic lot-size model. *Management Science*, 5:89–96, 1958.

Integrating Lot Sizing and Scheduling with the Vehicle Routing Problem: A special look into perishable products

Anderson Meneses DEIG, Faculty of Engineering, University of Porto, Rua Dr. Roberto Frias, s/n, 4600-001 Porto, Portugal
anderson_meneses@hotmail.com

Pedro Amorim DEIG, Faculty of Engineering, University of Porto, Rua Dr. Roberto Frias, s/n, 4600-001 Porto, Portugal
amorim.pedro@fe.up.pt

Bernardo Almada-Lobo DEIG, Faculty of Engineering, University of Porto, Rua Dr. Roberto Frias, s/n, 4600-001 Porto, Portugal
almada.lobo@fe.up.pt

Christian Almeder
Chair for Supply Chain Management, European University Viadrina Frankfurt (Oder), Große Scharrntstr. 59, 15230 Frankfurt (Oder), Germany
almeder@europa-uni.de

1   Introduction

An integrated perspective of industrial functions such as production and distribution provides performance improvements reflected in visible cost reduction. Besides empirical knowledge and practices of combined production and distribution, theoretical and methodological approaches for the optimization of integrated versions of those problems have been studied over the years, indicating that holistic perceptions and integrated analyses are advantageous in strategic, tactical and operational levels.This integration entails substantial impact on both industrial activities and savings. These advantages are believed to be leveraged when the goods under study are subject to physical deterioration.

Our work will focus on developing a mathematical model to integrate the general lot-sizing and scheduling problem (GLSP) [1] and the vehicle routing problem with time windows (VRPTW) in order to understand the complexities and advantages of such integration both from a methodological and practical perspective.

2 Problem Statement and Mathematical Formulation

The operational production and distribution planning problem considered in this work consists of parallel lines with a limited capacity, which produce multiple items to be delivered to a set of final customers. This kind of problems is recurrent in catering companies. Here, orders arrive with a very short notice and they need to be produced and delivered complying with very tight customer requirements. Furthermore, the utilization of equipments such as ovens make the setup between different products highly dependent on the sequence. Hence, products are to be scheduled on the parallel production lines over a finite planning horizon that coincides with the last scheduled delivered.

There exists initial stock that may be used to fulfil current demand. This stock is particularly important for smoothing the pressure of delivering products to the customers requiring early deliveries. However since we are dealing with perishable products its use is of limited utility and durability.

The planning horizon is divided into a fixed number of non-overlapping micro-periods with variable length. Since the production lines can be scheduled independently, this is done for each line separately. The length of a micro-period is a decision variable, expressed by the production quantity of a certain product on a line and by the time to set up the machine in case it is necessary. A sequence of consecutive micro-periods, where the same product is produced on the same line, defines the size of the lot of a product. Therefore, a lot may continue over several micro-periods. The number of micro-periods of each day defines the upper bound on the number of setups possible to be done during the planning horizon.

The delivery function is assured by a logistic provider, which follows the company instructions in terms of routing. There exists a fixed price for each vehicle used throughout the planning horizon as well as a variable cost dependent on the total distance travelled. It is assumed that the logistic provider is able to cope with whatever distribution planning was decided beforehand and, hence, there exists no overall capacity restriction for transportation. This assumption is realistic since usually reference contracts are established assuring that there exists always a fleet with sufficient size available. Our first analysis considers that the distances between the production plant and customers are small enough so that the decrease of freshness during the transportation process is considered to be negligible.

A customer order may aggregate several products that have to be delivered within strict time windows. Moreover, it is assumed that demand is dynamic and deterministic.

The problem is to model production and distribution so as to minimize total cost of the supply chain covering these processes over the planning horizon.

56

The challenges related with the mathematical model in order to assure a correct synchronization between stages of this integrated problem go in two directions. GLSP has an hybrid time structure encompassing discrete macro-periods and continuous micro-periods. Whereas, in the VRPTW, time is modelled in a continuous way. One of the reasons driving the existence of a discrete time structure for the GLSP is the existence of important external factors influencing production planning such as demand split per day. When integrating with VRPTW the demand elements are controlled by the distribution process, which is closer to the customer and so the delivery time is fixed by the time windows. For this reason we just used the micro-period structure in the integration of these models going for a continuous time representation. The second challenge relates with the restriction of only letting one vehicle depart if all the production for all the customers serviced by this vehicle is completed. We modelled this constraint with the help of a continuous decision variable, f_c, controlling the possible time of departure for each customer separately. Ensuring, afterwards, that the departure takes place only after the latest customer order arrives.

3   Illustrative example

The aim of our illustrative example is to demonstrate that the integrated formulation is advantageous in comparison to the solution of solving both problems separately.

Let us consider an instance with five products to be produced in only one production line. Changing the production between any pair of products entails a setup dependent on the sequence. There exist 25 customers having demand for all products. The data for the customers regarding total demand, service time, travel time and time windows is taken from instance C101 of Solomon [2].

In Table 1, two solutions for this instance are presented, one coming from the solution of the integrated problem and another coming from its decoupled form.

Both solutions were found by running the models in Cplex. In this small case the integrated approach is 7.5% more economic than the decoupled one. If we were to solve the decoupled approach merely by either, first addressing the production problem and then feeding f_c to the distribution one, or vice-versa, it would not be possible to find a feasible solution. The method was then to consider the minimization

	Production	Distribution	Total
Integrated	240	900.03	1140.03
Decoupled	100	1132.2	1232.2

Table 1: Solutions for the illustrative example.

(maximization) of f_c in the production (distribution) objective function and then feed forward (backward) its value.

4   Conclusion

This work aims at shedding some light in the integrated operational production and distribution of perishable goods. A mathematical model to integrate these complex problems was developed and its main subtleties unveiled. Future work will point in two directions. First, a robust solving method to such a complex problem needs to be developed. Second, it is of most importance to incorporate the perishability phenomenon explicitly in the formulation so that we can understand the dynamics of its interaction in the integrated planning.

References

[1] Fleischmann, B. and Meyr, H., The general lotsizing and scheduling problem, OR Spectrum, 19, 11-21 (1997)

[2] Solomon, M. M., Algorithms for the vehicle routing and scheduling problems with time window constraints, Operations Research, 35, 254-265 (1987)

The Economic Lot Sizing Problem with Perishable Items and Production Capacities

Mehmet Onal Isik University, Istanbul, Turkey
onal@isikun.edu.tr

H. Edwin Romeijn University of Michigan, Michigan, U.S.A.
romeijn@umich.edu

Wilco van den Heuvel Erasmus University Rotterdam, Rotterdam, Netherlands
wvandenheuvel@ese.eur.nl

1   Introduction

In the basic economic lot sizing problem (ELS), it is assumed that the items remain intact and therefore can be kept in inventories indefinitely to meet future demands. Since the items do not deteriorate, the order the items are consumed does not matter. Any item can be sold any time during the planning horizon. The economic lot sizing problem with perishable items (ELS-PI) is an extension of the ELS where it is assumed that items perish after an expiration date. Unlike the ELS, in the ELS-PI, the order of consumption matters; if the items with earlier expiration dates are not consumed early, they may deteriorate before they are consumed.

In this paper, we consider the ELS-PI with production capacities. We assume that there is a discrete and finite planning horizon of T periods and any item procured in period t, $1 \leq t \leq T$, expires in period v_t. As in [1], we analyze the problem assuming four different item consumption orders: *First-In-First-Out* (FIFO), *Last-In-First-Out* (LIFO), *First-Expiration-First-Out* (FEFO), and *Last-Expiration-First-Out* (LEFO). In FIFO (LIFO), the items produced in period t are always consumed earlier than the items produced in period t' if $t < t'$ ($t > t'$). In FEFO (LEFO), the items produced in period t are always consumed earlier than the items produced in period t' if $v_t < v_{t'}$ ($v_t > v_{t'}$). We assume that there is a finite production capacity C_t for $t = 1, \ldots, T$ such that amount of items produced in period t can not exceed C_t. It is well known that the ELS with procurement capacities is NP-hard even under various special cost structures (see Florian et al. [2]). Since the ELS-PI generalizes

the ELS by incorporating item deterioration, it immediately follows that the ELS-PI with production capacities is also NP-hard under all those special cases. On the other hand, the economic lot sizing problem with constant production capacities can be solved in polynomial time under general concave cost functions (see Florian and Klein [3]). We, therefore, analyze the problem assuming that $C_t = C$ for $t = 1, \ldots, T$.

We prove that, although the ELS with constant production capacities is polynomially solvable, the ELS-PI with constant production capacities is NP-hard when the consumption order is FEFO or LIFO. This holds true even when the inventory holding costs are zero and production cost functions have a fixed charge structure where the variable part is zero. Then, we propose an $O(T^4)$ algorithm for the problem with LEFO consumption order and another $O(T^4)$ algorithm for the problem with FIFO consumption order assuming general concave production and inventory holding cost functions.

We also note that for any two periods $i < j$, if the expiration dates have the property that $v_i \leq v_j$, then the problem with FEFO consumption order is equivalent to the problem with FIFO consumption order; and the problem with LIFO consumption order is equivalent to the problem with LEFO consumption order. Hence, in that case, the problems with FEFO and the LIFO consumption orders can both be solved in polynomial time assuming general concave production and inventory holding costs.

References

[1] M. Onal, Extensions of the Economic Lot Sizing Problem, PhD thesis, University of Florida (2009)

[2] M. Florian, J.K. Lenstra, A.H.G. Rinnooy Kan, Deterministic production planning: algorithms and complexity, Management Science, 26, 669-670 (1980)

[3] M. Florian, M. Klein, Deterministic production planning with concave costs and capacity constraints, Management Science, 18, 12-20 (1971)

Combining the principles of Variable Neighborhood Decomposition Search and Fix & Optimize heuristic to solve multi-level lot-sizing and scheduling problems

Florian Seeanner   Darmstadt University of Technology
seeanner@pscm.tu-darmstadt.de

Bernardo Almada-Lobo   University of Porto
almada.lobo@fe.up.pt

Herbert Meyr   University of Hohenheim
h.meyr@uni-hohenheim.de

1   Introduction

The consumer goods industry is typically characterized by a highly automated flow shop production system which often consists of two or three production stages (e. g. make-and-pack). On each stage potentially several production lines can be used alternatively as they offer – at least partially – the same services. Generally, many final items of different types are produced which can be assigned to a few setup families. Usually setup times for changeovers between products of the same family can be neglected. In contrast, setup times between different families are significantly sequence-dependent. So there is a need for simultaneously determining lotsizes and sequences. Moreover, this industry typically has to face time-varying demands due to seasonality, promotion activities and other factors. As a consequence, the demand mixture of items may change over time. According to the bill-of-material different demand mixtures may utilize production stages differently. Therefore, this can cause so-called "shifting bottlenecks" (on different lines and periods) which enforce a simultaneous consideration of several production levels, too. Unfortunately, only a few models and solution procedures meeting these requirements do actually exist [3]. One reason for this might be that even single level models are hard to solve in terms of complexity (cf. [4, 6, 13]). But also the scalability of solution methods might be a critical issue.

Motivated by these facts we developed a new heuristic which is able to solve multi-level lot-sizing and scheduling problems in a reasonable amount of time. It represents a combination of a meta-heuristic and a heuristic based on an exact mathematical programming approach which is similar to the approach by James and Almada-Lobo [12]. Strictly speaking, we bring together the Variable Neighborhood Decomposition Search (VNDS) and the Fix&Optimize (F&O) heuristic, also known as "Exchange".

Hansen and Mladenović [7] developed the principle of Variable Neighborhood Search (VNS) which is based on the concept of neighborhood search. In order to avoid being entrapped in a local optimum which is not the global one, VNS systematically changes the neighborhood structure in a so-called shaking phase. In the past years many variants of this meta-heuristic have been successfully applied to a wide range of optimization problems, like problems of graph theory (e.g. traveling salesman problem [5, 7], minimum spanning trees [19, 22]), supply chain planning problems (e. g. car sequencing [21]), continuous optimization [15], or lot-sizing and scheduling problems [1, 2]. One variant of VNS is called Variable Neighborhood Decomposition Search (VNDS) [9]. In contrast to standard VNS it does not search the whole solution space but only a subset which is the result of some kind of decomposition.

On the other hand, Fix&Optimize is a Mixed-Integer-Programming (MIP) based improvement heuristic which solves iteratively a series of sub-MIPs. It starts with a given solution and decomposes the integer variables into two subsets in every step. One part of the variables are fixed to the values found so far. The other variables, however, are "released" which means that they are to be optimized again [20]. Accordingly, a feasible solution can always be found by a standard MIP-solver and thus, a new solution is at least as good as the old one [20]. Note that it is really crucial to decide which and how many variables should be released. Typically, several iterations of this procedure with different subsets are executed. In the last couple of years this heuristic has become quite popular. Referenced as "Fix&Optimize" this heuristic was applied to lot-sizing (and scheduling) problems quite successfully [11, 23, 24, 25].

The basic idea of our approach is to apply the concept of VNDS in order to adapt methodically the variable sets for the Fix&Optimize heuristic.

2   Presentation content

As an example of a lotsizing and scheduling model the General Lotsizing and Scheduling Problem for Multiple production Stages (GLSPMS) is briefly described first. Then, after a short introduction to the basics of VNDS and Fix&Optimize, we want to explain the new heuristic. For that purpose we illustrate the used decompositions with the help of the GLSPMS. In order to demonstrate the strength of our approach computational results of artificial test instances and a real-world-instance are shown.

References

[1] Almada-Lobo, B. and James, R.J.W., Neighbourhood search meta-heuristics for capacitated lot-sizing with sequence-dependent setups., International Journal of Production Research, 48, 861 - 878 (2010)

[2] Almada-Lobo, B. and Oliveira, J. F. and Carravilla, M. A., Production planning and scheduling in the glass container industry: A VNS approach, International Journal of Production Economics, 114, 363 - 375 (2008)

[3] Clark, A. and Almada-Lobo, B. and Almeder, C., Editorial on Lot Sizing and Scheduling: Industrial extensions and research opportunities, European Journal of Production Research, 49, 1 - 4 (2011)

[4] Drexl, A. and Haase, K., Lot sizing and scheduling survey and extensions, European Journal of Operational Research, 99, 337 - 348 (1997)

[5] Felipe, A. and Ortuño, M. T. and Tirado, G., The double traveling salesman problem with multiple stacks: A variable neighborhood search approach, Computers & Operations Research, 36, 2983 - 2993 (2009)

[6] Fleischmann, B., The discrete lotsizing and scheduling problem with sequencedependent setup costs, European Journal of Operational Research, 75, 395 - 404 (1994)

[7] Hansen, P. and Mladenović, N., Variable neighborhood search: Principles and applications, European Journal of Operational Research, 130, 449 - 467 (2001)

[8] Hansen, P. and Mladenović, N., Variable neighborhood search for the P-median, Location Science, 5, 207 - 226 (1997)

[9] Hansen, P. and Mladenović, N. and Perez-Britos, D., Variable Neighborhood Decomposition Search, Journal of Heuristics, 7, 335 - 350 (2001)

[10] Hansen, P. and Oğuz, C. and Mladenović, N., Variable neighborhood search for minimum cost berth allocation, European Journal of Operational Research, 191, 636 - 649 (2008)

[11] Helber, S. and Sahling, F., A fix-and-optimize approach for the multi-level capacitated lot sizing problem, International Journal of Production Economics, 123, 247 - 256 (2010)

[12] James, R. J. W. and Almada-Lobo, B., Single and parallel machine capacitated lotsizing and scheduling: New iterative MIP-based neighborhood search heuristics, Computers & Operations Research, 38, 1816 - 1825 (2011)

[13] Meyr, H., Simultaneous lotsizing and scheduling on parallel machines, European Journal of Operational Research, 139, 277 - 292 (2002)

[14] Meyr, H., Simultaneous lotsizing and scheduling by combining local search with dual reoptimization, European Journal of Operational Research, 120, 311 - 326 (2000)

[15] Mladenović, N. and Dražić, M. and Kovačevic-Vujčić, V. and Čangalović, M., General variable neighborhood search for the continuous optimization, European Journal of Operational Research, 191, 753 - 770 (2008)

[16] Mladenović, N. and Hansen, P., Variable Neighborhood Search, Computers and Operations Research, 24, 1097 - 1100 (1997)

[17] Mladenović, N. and Petrović, J. and Kovacević-Vujcić, V. and Cangalović, M., Solving speed spectrum radar polyphase code design problem by tabu search and variable neighbourhood search, European Journal of Operational Research, 151, 389 - 399 (2003)

[18] Mladenović, N. and Urošević, D. and Peérez-Brito, D. and García-González, C., Variable neighbourhood search for bandwith reduction, European Journal of Operational Research, 200, 14 - 27 (2010)

[19] Naji-Azimi, Z. and Salari, M. and Golden, B. and Raghavan, S. and Toth, P., Variable neighborhood search for the cost constrained minimum label spanning tree and label constrained minimum spanning tree problems, omputers & Operations Research, 37, 1952 - 1964 (2010)

[20] Pochet, Y. and Wolsey, L., Production Planning by Mixed Integer Programming. Springer, New York (2006)

[21] Prandtstetter, M. and Raidl, G. R., An integer linear programming approach and a hybrid variable neighborhood search for the car sequencing problem, European Journal of Operational Research, 191, 1004 - 1022 (2008)

[22] Ribeiro, C. C. and Souza, M. C., Variable neighborhood search for the degree-constrained minimum spanning tree problem, Discrete Applied Mathematics, 118, 43-54 (2002)

[23] Sahling, F., Mehrstufige Losgrenplanung bei Kapazittsrestriktionen. Gabler, Hannover (2010)

[24] Sahling, F. and Buschkhl, L. and Tempelmeier, H. and Helber, S., Solving a multi-level capacitated lot sizing problem with multi-period setup carry-over via

a fix-and-optimize heuristic, Computers & Operations Research, 36, 2546-2553 (2009)

[25] Stadtler, H., Multi-Level Single Machine Lot-Sizing and Scheduling with Zero Lead Times, European Journal of Operational Research, 209, 241-252 (2011)

New Algorithms and New Questions for the Multiple Stage Lot Sizing Problem

Natalie Simpson University at Buffalo (SUNY)
nsimpson@buffalo.edu

1   Introduction

In the un-capacitated multiple stage lot sizing problem, each stage of a product structure network is assumed to have a fixed ordering cost and a linear holding cost associated with it. One or more stages within this network will experience deterministic, time-variant external demand, creating dependent demand across the balance of the network. Lead times between the stages are assumed to be constant, and thus can be translated into an equivalent problem in which lead times are zero. These assumptions model situations as diverse as finished product assembly from a variety of components, the cracking of a single commodity into a range of finished products, and the distribution of goods to several locations through multiple intermediaries. The Non-sequential Incremental Part Period Algorithm with Right-shifting, or NIPPAR, is an improved heuristic for identifying low cost replenishment schedules across all these network types. It is a direct descendent of the Non-sequential Incremental Part Period Algorithm (NIPPA), first introduced in [5]. NIPPA is a neighborhood search procedure that begins with a feasible solution, usually lot-for-lot, and left-shifts to create an improved solution, consolidating the order for an item, as well as the orders for any necessary predecessors of that item, into their next earliest respective ordering periods. To select the next move in the search, the ratios of cost incurred to savings gained by each and every opportunity to left-shift in the current solution are calculated. The lowest ratio is then selected and left-shifting is implemented if that ratio value is less than one; otherwise, the algorithm terminates with its recommended solution. NIPPAR introduces the option of right-shifting as a potential move from the current solution. Right-shifting as a move in neighborhood search is defined as establishing an order in a non-ordering period, combined with establishing orders at any succeeding stages that are likewise currently non-ordering periods, maintaining nestedness as a quality of the solution. The size of the proposed new orders are calculated to satisfy Wagner-Whitin conditions, and the next earliest order periods are reduced to reflect the relocation of this replenishment farther to the right on the time line. As the option of right-shifting can be introduced into a search in a variety

66

of ways, one objective of this study is to determine which implementation is most desirable. Five different implementations are tested:

NIPPAR-1, Right-Shift Only, begins from a Stockpiled Starting Solution.

NIPPAR-2. Left and Right-Shifts, begins from a Lot-for-Lot Starting Solution.

NIPPAR-3. Left and Right-Shifts, begins from a Stockpiled Starting Solution.

NIPPAR-4. The best of four heuristics, NIPPA and NIPPAR-1,2,3.

NIPPAR-5. The best of NIPPAR-2 and NIPPAR-3.

Stockpiled starting solutions are initial solutions in which all demand is met in the first period of the time horizon.

2   Test Bed Description

Product Structures and Demand Patterns. The test bed currently consists of 15 product structures, divided into a 5-stage and 32-stage group. The nine 5-stage structures include one serial structure, two general structures, and three sets of paired assembly and distribution networks, where set members are vertical reflections or reverse interpretations of one another. Most of these structures have a long history of test bed use in the published literature, with references as early as [2] [3]. The 32-stage group currently consists of six structures, or three pairs of assembly/distribution network siblings. Each end item external pattern consists of 24 periods, distributed discrete uniform (0,200). Ten random replications are employed per structure.

Cost Parameters. In pure assembly or distribution structures, the singular end item or the singular source is assigned a per period echelon holding cost of 1.0. The echelon holding costs at all other stages in the product structure are then randomly selected from this set of values: 0.1, 0.5, 1.0, 2.0. In general structures, the echelon holding cost of all stages are selected randomly from this same set. Five holding cost patterns are generated for each structure. For pure assembly patterns, the fixed cost at each stage is selected randomly from this set: 150, 300, 600, 1500. For distribution and general structures, a value X is chosen at random from this set of values: 75, 150, 300, 600, 1500, 3000 for each stage. The fixed cost assigned to that stage is nX, where n is the number of parents of that stage.

Benchmark Solutions.   To evaluate cost performance, a benchmark is identified for each problem instance. In the case of assembly structures, the optimal solution to each problem instance is found by solving a modified form of an all-binary formulation first introduced by [4]. For generalized and distribution structures, a lower bound on each problem instance is identified with Lagrangean relaxation, initially discussed in [1].

Noise Levels. To simulate the uncertainty that may exist concerning the true costs associated with any stage of a multiple-stage system, each problem instance is simulated with one of three noise levels concerning holding and ordering cost param-

eters. If the actual ordering or per period echelon holding cost of an item is X, the corresponding parameter used in the application of any heuristic is rX, where r is a random factor distributed continuous uniform(1-u, 1+ u). The three noise levels represent three values of u: u= 0, u = 0.5, and u = 1. A factor level of u = 0 represents no noise or mistaken information passed to the heuristic procedure, the traditional test of the heuristic against a problem instance.

Heuristics. NIPPA and each variant of NIPPAR is applied to each problem and the total cost of the solution is recorded. Graves Algorithm, the best performing decomposition-based procedure from a [3] is applied for comparison, as well as two relatively simple procedures, the Sequential Wagner-Whitin (SWW) and the Cumulative Wagner-Whitin (CWW). SWW provides an important benchmark of local optimization, in that each stage is solved for its optimal solution with respect to its local costs and the demands passed down from above it. CWW is a similar top-down, single-pass procedure that differs only in its use of aggregated ordering costs (the combination of a stages ordering cost with those of all its necessary predecessors) when finding replenishment schedules at each stage.

In summary, each product structure will be solved or bounded for 10 random demand patterns x 5 random fixed ordering cost patterns x 5 random echelon holding cost patterns, for a total of 10x5x5 = 250 test bed instances per product structure. Three levels of noise for each of these instances results in 750 problem instances per structure, and 9+6 = 15 product structures yields a total of 11,250 problem instances to test each heuristic.

3 Early Results

Table 1 provides the overall performance of the nine lot-sizing techniques when applied to each test problem instance. The introduction of noise at a level of u = 0.5 represents ordering and holding costs being mistaken at random by as much as 50% above or below their actual values. Table 2 shows the result of solving the 5-stage problems with this level of mistaken information, calculated as the increase in the true cost of heuristics solution when found with the mistaken parameters versus the true parameters.

4 Discussion

The new algorithm NIPPAR is demonstrated as a robust solution technique across the broad scope of this test bed. NIPPAR-3, (NIPPAR commencing from a stockpiling solution) appears to be the best single heuristic application, although solving a problem twice using a stockpiling and a lot-for-lot starting solution in each case improves overall performance further (NIPPA-5). While distinctly superior overall, the

neighborhood search logic of NIPPAR does not dominate in all cases within the test bed, an issue that appears related to the interaction of a high number of end items within a product structure and a low number of levels within same, which will be explored further. The early results from noise level simulations are quite intriguing, as shown in Table 2. In the currently available cases of five stage structures, a noise level of 0.5 represents feeding each algorithm mistaken values for the cost parameters that can range as high as 50% on either side of the actual values. Using this noisy information, the actual cost performance of the algorithms suffer a fairly uniform 2%-3% loss of true cost performance, suggesting that the assumption of perfect information concerning these values implied by the problem formulation may not be critical to good performance in reality. How this loss behaves in the context of even higher noise levels and the larger product structures is currently being explored.

References

[1] Afentakis P. and Gavis, B. Optimal Lot-Sizing Algorithms for Complex Product Structures, Operations Research, Vol. 34(2), 237-249 (1986).

[2] Blackburn J. and Millen R., Improved Heuristics for Multistage Requirements Planning Systems,Management Science, 28, 44-56 (1982)

[3] Graves S. , Multi-Stage Lot-Sizing: An Iterative Procedure. In Multi-Level Production/Inventory Control Systems: Theory and Practice, ed. L. B. Schwarz. North-Holland, Amsterdam (1981).

[4] McKnew M. Saydam C. and Coleman B., An Efficient Zero-One Formulation of the Multilevel Lot-Sizing Problem, Decision Sciences, 22, 280-295 (1991) .

[5] Simpson N. and Erenguc S., Improved Heuristic Methods for Multiple Stage Production Planning, Computers and Operations Research, 25 (7-8), 611-623 (1998)

Algorithm	5-Stage Assembly (n=1000)	32-Stage Assembly (n=750)	5-Stage Distribution (n=750)	5-Stage Generalized (n=500)	32-Stage Distribution (n=750)
NIPPA	0.91%	1.02%	2.96%	2.66%	7.18%
NIPPAR-1	0.78%	0.74%	2.65%	3.20%	9.09%
NIPPAR-2	0.90%	1.01%	2.94%	2.63%	7.16%
NIPPAR-3	0.74%	0.71%	2.25%	2.71%	6.52%
NIPPAR-4	0.46%	0.60%	1.68%	1.76%	5.35%
NIPPAR-5	0.46%	0.60%	1.68%	1.76%	5.35%
Graves	2.30%	2.50%	1.96%	1.92%	5.61%
SWW	5.65%	8.56%	5.89%	4.90%	12.45%
CWW	1.34%	6.18%	33.00%	28.05%	49.03%

Table 1: Average Optimality Gap (Assembly) or Gap From Lower Bound, u = 0 (No Noise).

Algorithm	5-Stage Assembly (n=1000))	5-Stage Distribution (n=750)	5-Stage Generalized (n=500)
NIPPA	2.29%	2.11%	1.67%
NIPPAR-1	2.33%	1.87%	2.21%
NIPPAR-2	2.28%	2.09%	1.64%
NIPPAR-3	2.38%	1.84%	2.46%
NIPPAR-4	2.29%	2.09%	2.01%
NIPPAR-5	2.29%	2.09%	2.01%
Graves	2.21%	3.82%	2.70%
SWW	1.69%	3.19%	4.89%
CWW	2.48%	3.61%	4.52%

Table 2: Average Loss of Cost Performance from Noise Level u = 0.5.

The Impact of Modeling on Robust Inventory Management under Demand Uncertainty

Oguz Solyali Business Administration Program, Middle East Technical University, Northern Cyprus Campus, Kalkanli, TRNC, Mersin 10, Turkey
solyali@metu.edu.tr

Jean-François Cordeau CIRRELT and Canada Research Chair in Logistics and Transportation, HEC Montréal, Montréal H3T 2A7, Canada
jean-francois.cordeau@hec.ca

Gilbert Laporte CIRRELT and Canada Research Chair in Distribution Management, HEC Montréal, Montréal H3T 2A7, Canada
gilbert.laporte@cirrelt.ca

1   Extended Abstract

Robust optimization (RO) is a powerful approach to optimization problems involving uncertain parameters with ambiguous probability distributions. It incorporates uncertain parameters into the optimization models in a tractable way and finds the best solution considering all possible realizations of uncertain data while ensuring feasibility regardless of their realization.

Several modeling frameworks are available within the RO paradigm because uncertainty can be handled in a number of ways. Here, we consider two of the prominent RO modeling frameworks: the approaches of Bertsimas and Sim [2] and Ben-Tal et al. [3]. Bertsimas and Sim [2] proposed the budget of uncertainty approach in which only some of the uncertain parameters are allowed to simultaneously deviate from their nominal values and the degree of conservativeness of solutions can be controlled. This modeling framework has the desirable property that the robust counterpart of the uncertain problem preserves the complexity of its nominal (i.e., without uncertainty) problem and thus this approach naturally extends to discrete optimization problems [1]. Ben-Tal et al. [3] developed the adjustable robust counterpart (ARC) for uncertain multi-stage LP problems in which some decision variables, called nonadjustable, are determined a priori (i.e. at time 0) whereas others, called adjustable, are set having observed the realization of some uncertain parameters. [3] show that

ARC leads to a smaller objective function value than do pure robust counterpart if uncertainty affecting one constraint also affects others. Since the ARC of an uncertain LP problem is intractable except for some restricted cases, [3] proposed, as a tractable approximation to the ARC, the affinely adjustable robust counterpart (AARC), in which adjustable variables are expressed as an affine function of realized uncertain parameters.

Robust inventory management problems under demand uncertainty have been addressed by many researchers using different robust optimization approaches. Ben-Tal et al. [3, 4] formulate two different inventory management problems as uncertain LP problems and consider their AARCs which give a better objective function value than their pure robust counterparts. Bertsimas and Thiele [5] model a single-installation and a multi-installation inventory management problem with a tree network structure as mixed integer linear programming problems (resp. linear programming problems) in the presence (resp. absence) of fixed order costs using the robustness approach developed by [2]. Bienstock and Ozbay [6] solve the same single-installation inventory management problem without fixed order costs as [5], besides the problem with base-stock policy. For these problems, [6] propose an interesting decomposition algorithm where a nonconvex auxiliary subproblem, called adversarial problem, is first solved to determine a realization of demands which is then used to solve a master problem with all previously generated demand realizations, and this process is iteratively applied until the gap between the upper and lower bounds vanishes. Ben-Tal et al. [7] model a multi-installation inventory management problem with a serial structure as an uncertain LP problem, and consider its globalized robust counterpart, which is an extension of the AARC. Unlike the mentioned studies, See and Sim [8] consider a single-installation inventory management problem with nonzero order lead times and characterize uncertain demand by covariance and directional deviations besides the usual mean and support information. They formulate the problem as a second order cone programming problem. Note that all these papers except [5] consider zero fixed order costs (i.e., there are no integer decision variables).

In this study, we consider the same basic single-installation inventory management problem as [5] and [6], where the aim is to determine when and how much to order under interval demand uncertainty over a planning horizon of T time periods, so that the sum of order and inventory holding/backlogging costs is minimized, while ensuring solution feasibility for any possible demand realization in the given uncertainty set. Our aim is to study the impact of modeling in robust inventory management under demand uncertainty, with and without budget of uncertainty. We present several robust formulations constructed by using the main known robust optimization approaches (i.e. pure robust, budget of uncertainty robust and affinely adjustable robust counterparts), and compare these with newly proposed ones.

While all studies in the robust inventory management literature, to the best of our knowledge, use a variant of standard inventory flow balance equations in their robust

72

formulations, we consider a reformulated approach and propose two new robust formulations called NR1 and NR2, without and with budget of uncertainty, respectively. We show that our new robust formulation without budget of uncertainty represents the problem better than the pure and affinely adjustable robust counterparts, by yielding a smaller objective function value when the initial inventory is zero.

We also show that the new robust formulation with budget of uncertainty can be used to solve the problem with budget of uncertainty in polynomial time by solving $O(T^2)$ LP problems when the initial inventory is zero. We further prove that the same problem is polynomially solvable in $O(T^4)$ time when the initial inventory is zero. Moreover, we show that the new robust formulation yields a smaller objective function value than the robust formulation of [5] when the initial inventory and fixed order costs are zero.

We show that the new formulation with budget of uncertainty dominates the one without budget of uncertainty when the initial inventory is zero.

The computational results on test instances reveal that NR1 and NR2 not only show a superior performance with regard to the optimal objective function value compared to other formulations, but also outperform the others in terms of CPU time needed for optimally solving instances.

References

[1] D. Bertsimas and M. Sim, Robust discrete optimization and network flows, Mathematical Programming Series B, 98, 49–71 (2003)

[2] D. Bertsimas and M. Sim, The price of robustness, Operations Research, 52, 35–53 (2004)

[3] A. Ben-Tal, A. Goryashko, E. Guslitzer and A. Nemirovski, Adjustable robust solutions of uncertain linear programs, Mathematical Programming, 99, 351–376 (2004)

[4] A. Ben-Tal, B. Golany, A. Nemirovski and J.-P. Vial, Retailer-supplier flexible commitments contracts: A robust optimization approach, Manufacturing & Service Operations Management, 7, 248–271 (2005)

[5] D Bertsimas and A. Thiele, A robust optimization approach to inventory theory, Operations Research, 54, 150–168 (2006)

[6] D. Bienstock and N. Ozbay, Computing robust basestock levels, Discrete Optimization, 5, 389–414 (2008)

[7] A. Ben-Tal, B. Golany and S. Shimrit, Robust multi-echelon multi-period inventory control, European Journal of Operational Research, 199, 922–935 (2009)

[8] C.-T. See and M. Sim, Robust approximation to multiperiod inventory management, Operations Research, 58, 583–594 (2010)

Capacitated lot sizing with multiple machines, sequence-dependent setups and common setup operators: A practical case

Horst Tempelmeier, Sascha Herpers, and Karina Copil   University of Cologne, Dep. of Supply Chain Management and Production,
Albertus Magnus-Platz, D-50932 Cologne, Germany
tempelmeier@wiso.uni-koeln.de

1   Introduction

Although a large number of approaches to lot sizing problems have been developed and much progress has been observed in the past, the solution of real-life problem instances is still a big challenge. In the current paper, we describe a practical case and discuss the first steps undertaken to solve the problem. Besides sequence-dependent setup times, the main complication of the problem is, that for the setups a specialized setup resource is required. This prevents the decomposition of the multi-machine problem into multiple single-machine lotsizing problems. Although we focus on a practical case from food production, problems of this type have also been observed in several automotive companies.

We propose a model that extends the Capacitated Lot Sizing Model with Linked Lotsizes and sequence dependent setups (CLSP-L-SD). A similar problem scenario is considered by [1]. These authors proposed a model based on the Proportional Lot Sizing Problem (PLSP). CLSP-L-type models without consideration of the common setup operator are presented by [2] for a single machine and extended to the multi-machine case by [3].

2   Problem description

We consider a lot sizing and scheduling problem that occurs in a German food company. A large number of products with dynamic demands is produced in a single production step on multiple resources. Setup times and production times are specific for each product-resource combination. Each production order is produced in several batches, whereby the processing time for the order, in addition to the setup time required at the beginning of the processing of the order, depends on the number

75

of batches. Setups are sequence-dependent and the setup state of the production resource can be carried over to the next periods (multiple setup carry-overs). The production resources require a specialized setup resource which can handle only one production resource at a time. Thus, all setups have to be coordinated on those production resources that require the same setup resource. In particular, the underlying problem has the following characteristics:

- The planning horizon is divided into T periods.

- K products are produced.

- An external period-specific demand per product d_{kt} must be met.

- Backlogging is not allowed.

- As the production runs over three shifts, overtime is not possible.

- M production resources (machines) with period-specified capacities b_t^m are available.

- A production lot is always associated with a setup operation performed by the setup resource which consumes time and costs money. These times and costs are sequence-dependent.

- A production lot consists of multiple batches. Before a new batch is started, a fix amount of time for cleaning the production resource is required.

- The production of a lot may last more than one period. Setup states are carried over to the next period.

- Maximum and minimum shelf-life: The time between the production and the delivery date must not exceed given limits.

- The objective is to minimize the sum of holding and setup costs.

A typical problem instance that must be solved routinely comprises about 50 to 60 products, and about 6 to 8 production resources handled by a single setup operator. The planning horizon contains about 10 to 14 periods. The number of product-specific lots produced per production resource and period is about 6 to 8.

3   Solution approach

We develop model formulations that can be used to cover selected aspects of the considered problem. We model the problem with a big-bucket CLSP-L-type model considering sequence-dependent setups and setup carry-overs. For selected problem characteristics, we compare the CLSP-L formulation with a modification of the PLSP based model introduced by [1]. In order to account for the setup resource, we keep track of the exact timing of the setup operations on all associated production resources. The different model formulations are implemented with OPL and CPLEX and their performance is compared. The results show that small problem instances can be solved with CPLEX. However, for the complete problem size CPLEX does not even find a feasible solution. Hence, heuristic approaches are required.

References

[1] Horst Tempelmeier and Lisbeth Buschkühl, Dynamic multi-machine lotsizing and sequencing with simultaneous scheduling of a common setup resource. International Journal of Production Economics, 113(2008), pp. 401–412

[2] B. Almado-Lobo and D. Klabjan and M. A. Carravilla and J.F. Oliveira, Single machine multi-product capacitated lot sizing with sequence-dependent setups. International Journal of Production Research 20(2006), pp. 4873–4892

[3] R.J.W. James and B. Almado-Lobo, Single and parallel machine capacitated lotsizing and scheduling: New iterative MIP-based neighborhood search heuristics. Computers and Operations Research, 38(2011), pp. 1816–1825

Multilevel capacitated lot sizing and scheduling: A model overview and comparison

Renate Traxler, Christian Almeder
Chair for Supply Chain Management, European University Viadrina Frankfurt
(Oder), Große Scharrntstr. 59, 15230 Frankfurt (Oder), Germany
{traxler,almeder}@europa-uni.de

1   Introduction

It is well known that when doing lot sizing for multi-stage systems with limited capacities a simultaneous consideration of the scheduling aspect is mandatory in order to avoid drawbacks in the solution quality. There are several models available in literature, each one based on slightly different assumptions. In this paper, a comparison of three model classes is conducted. The first class reflects the big-bucket models. Starting from the basic multilevel capacitated lot sizing problem (MLCLSP) Two variants presented in [1] are considered for this comparison. One covers the lot streaming case (referred to as MLCLSP_{LS}) and the other allows production in batches (MLCLSP_{BS}), where the single batches are synchronized. The second class of models are the pure small-bucket models. Here the continuous setup lot sizing problem (CSLP_{ML}) and the proportional lot sizing problem (PLSP_{ML}) for the multilevel case are analyzed. Moreover, the third class consists of variants of the general lot sizing and scheduling problem for multiple stages (GLSPMS) proposed in [2] with their underlying two-stage time structure. The main characteristic is the existence of variable micro periods embedded into macro periods with fixed length. In [2] the model formulation is stepwise refined in order to improve coordination between production stages. Four different variants are distinguished:

- GLSPMS_{base}: Either setup or production is allowed to be performed within a micro period on a machine.

- GLSPMS_{20}: Production prohibition in setup periods is limited to machines, where there exists a predecessor-successor relation.

- GLSPMS_{21}: Production is enabled as long as the micro period covers at least the production time for the particular predecessor, plus the time required to setup from the successor to another successor.

78

- GLSPMS$_{QS}$: The production amount of one period is fragmented into two parts, one is available to subsequent processing in the ongoing period and the other in the following period at the earliest.

2 Model characteristics

Scheduling is integrated in PLSP$_{ML}$ and CSLP$_{ML}$ implicitly, as by definition only one or two items are allowed to be produced within one period. In GLSPMS, scheduling is done simultaneously with lot sizing by determining the borders of the variable micro periods. In MLCLSP$_{BS}$, single batches are synchronized by determining the start time for each lot in each period. The obvious drawback of the standard MLCLSP of either generating possible infeasible (with regards to the scheduling issue) production plans when lead times are neglected, or generating production plans with high work-in-process (WIP) levels when a one-period lead time is considered, could be solved this way. In contrast, the small bucket models as well as some variants of the GLSPMS may take into account a short lead time of at least one micro period. However, the variant GLSPMS$_{QS}$ does not account for an extra lead time.

The formulations also have distinct assumptions regarding setup time restrictions. While the CSLP$_{ML}$ allows setups only to be performed at the beginning and the GLSPMS at the end of a micro period, in PLSP$_{ML}$ and MLCLSP, any time period within a micro or macro period is allowed, as long as the borders of the periods are not crossed. The standard MLCLSP does not allow setup carryover, which leads to multiple setups for the same item produced without interruption in consecutive periods. As the name implies, the MLCLSP with linked lot sizes (MLCLSP-L), which is not considered in this work, lifts this unfavorable restriction. It is not an issue in MLCLSP$_{LS,BS}$, small bucket models and GLSPMS as well.

A third group of differences concerns restrictions regarding the number of products and setups per period. As in models with micro period structure the number of products and setups is only restricted to the number of micro periods within a period, they could be seen advantageous beside big bucket models, where items are produced at most once per macro period. By offering the possibility of a user-defined variation of the number of periods, they are very flexible. Indeed, this characteristic does not exclusively offer advantages. One may assume that an increasing number of micro periods induces a reduction in total costs consisting of inventory and setup costs. However, it is possible, that for a certain number of periods there exists no feasible solution at all or that an increase of the number of periods leads also to an increase of the costs.

Table 1 gives an outline of major characteristics of the different time structures.

79

	$CSLP_{ML}$	$PLSP_{ML}$	$MLCLSP_{base}$	$MLCLSP_{BS}$	$MLCLSP_{LS}$	$GLSPMS_{base}$	$GLSPMS_{20}$	$GLSPMS_{21}$	$GLSPMS_{QS}$
lead times									
batching	-	-	x	x	-	-	-	-	-
lot-streaming	x	x	-	-	x	x	x	x	x
lead times	x	x	-	-	-	x	x	x	-
setup times									
time point of setup operation	begin	any	any			end			
setup carryover	x	x	-	x	x	x	x	x	x
setup time restrictions	1 MP.		-	-	-	1 MP.			
products and setups per macro period									
max. number of products	1/MP.	2/MP.	any			1/MP.			
max. number of setups/product and period	number MP.		1			number MP.			

Table 1: Outline of fundamental model characteristics

3   Computational Results

The models have been tested considering 144 instances of the smallest class proposed by [3] with 10 items, 3 machines and a planning horizon of 4 periods. The instances have been tested on an IBM x3650 with 2 Xeon Quadcore processors (3GHz), 24 GB RAM and Linux Ubuntu 7 using IBM ILOG CPLEX 12.1. A 30 minute run time limit is applied. The periods of the small bucket models and GLSPMS are fragmented into 5 micro periods and a one period lead time is incorporated with exception of the variant $GLSPMS_{QS}$. The main results are listed in table 2. The first three rows indicate the mean absolute percent deviation (MAPD) of the best solution derived in total for all models and separated with regard to inventory and setup costs. The next row indicates the average GAP to the lower bounds of each model.

Bearing in mind the quite different assumptions, results strongly differ. It seems clear that the variants which are based on lot streaming ($MLCLSP_{BS}$ and $GLSPMS_{QS}$) reach very small MAPD. Standard MLCLSP as a result of its disability to consider setup carryover as well as $MLCLSP_{BS}$ and $GLSPMS_{base}$ show relatively high MAPD of setup costs.

The results show clearly the consequences of problem complexity. $MLCLSP_{base}$ and the small bucket models $CSLP_{ML}$ and $PLSP_{ML}$ have a small GAP after 30 minutes, while for variants of the GLSPMS enormous high values remain. In spite of the high GAP values for $MLCLSP_{BS,LS}$, a respectable number of instances (39.6 %

	Small Bucket		MLCLSP			GLSPMS			
	$CSLP_{ML}$	$PLSP_{ML}$	base	BS	LS	base	20	21	QS
MAPD (%)	7.3	3.1	15.0	8.7	0.0	7.5	4.2	4.4	1.9
$MAPD_{inv}$ (%)	7.7	3.4	6.8	9.8	0.1	3.0	3.4	4.0	1.1
$MAPD_{setup}$ (%)	9.6	4.1	43.3	13.4	0.1	18.2	7.6	7.6	3.6
GAP (%)	3.9	8.6	0.0	7.9	6.3	30.0	21.5	21.7	20.3
RT (s)	1456	1797	¡1	1151	850	1800	1800	1800	1800

Table 2: Computational performance regarding MAPD, GAP and runtime (RT)

and 61.1%) is solved within 30 minutes leading to an average runtime of 1151 and 850 seconds. Similarly, $CSLP_{ML}$ is able to solve several instances within the time limit resulting in an average runtime of 1456 seconds, while for $PLSP_{ML}$ and GLSPMS 30 minutes are not sufficient.

4 Conclusions

The analysis of the model structures and the computational experiments make clear that the selection of the right model is of significant importance. If several models are suitable for a specific application, the quality of the generated production plans and the computational effort necessary depends heavily on the selected model class.

References

[1] Almeder, C., Klabjan, D., Almada-Lobo, B., Lead time considerations for the multilevel capacitated lot-sizing problem. Working paper (2010).

[2] Meyr, H., Simultane Losgrößen- und Reihenfolgeplanung bei mehrstufiger kontinuierlicher Fertigung. Zeitschrift für Betriebswirtschaft, 74(6), 585-610 (2004).

[3] Tempelmeier, H., Buschkühl, L., A heuristic for the dynamic multi-level capacitated lotsizing problem with linked lotsizes for general product structures. OR Spectrum, 31, 385-404 (2009).

A holding cost bound and a new heuristic for the economic lot-sizing problem

Wilco van den Heuvel Econometric Institute, Erasmus University Rotterdam
wvandenheuvel@ese.eur.nl

Albert P.M. Wagelmans Econometric Institute, Erasmus University Rotterdam
wagelmans@ese.eur.nl

1   Introduction

We consider the economic lot-sizing (ELS) model with time-invariant cost parameters. The model has a finite and discrete time horizon of T periods and there is a known demand stream d_t for the periods $t = 1, \ldots, T$. The problem is to determine the order periods and order quantities such that total costs are minimized. Costs include a fixed order cost K for every order placed, and a unit holding cost h per period for every item held in stock.

Although the ELS problem can be solved efficiently, many heuristics have been proposed in the literature and are still used in material requirements planning (MRP) software. A number of those heuristics utilize some optimality property of the economic order quantity (EOQ) model. For example the Silver-Meal (SM) heuristic minimizes the average cost per period, and the Least Unit Cost (LUC) heuristic minimizes the average cost per item. The Part Period Balancing (PPB) algorithm is a heuristic for the ELS model that is based on the property that the average fixed order and holding cost are equal in an optimal solution of the EOQ model. The PPB algorithm constructs a solution where fixed order and holding costs are balanced.

Since the demands are time-dependent in the ELS model, it is clear that the fixed order and holding costs are not perfectly balanced in an optimal solution in general. We are interested to what extend this property still holds. In particular, we are interested in an upper bound on the holding cost in an optimal order interval, where an order interval is defined as the number of consecutive periods for which demand is satisfied by a single order.

We will show that the holding cost is bounded by a quantity proportional to the fixed order cost and the logarithm of the number of periods in the interval. Based on this property, we will propose a new heuristic. We will prove that this heuristic has a worst case ratio of 2. Furthermore, by combining the new heuristic with an existing one, we are able to identify a class of heuristics with worst case ratio 2.

2 The holding cost bound

To derive the upper bound on the holding cost in an optimal order interval, we first introduce some notation. Consider an interval of t periods. Then the total holding cost H_t in this interval equals

$$H_t = h \sum_{i=2}^{t}(i-1)d_i.$$

Placing an order in some period $p+1$, $p \in \{1, \ldots, t-1\}$, leads to a saving in holding cost of $ph \sum_{i=p+1}^{t} d_i$. Since we consider an optimal solution, this quantity is at most equal to the fixed order cost which gives

$$ph \sum_{i=p+1}^{t} d_i \leq K.$$

Now assume that there exists a constant $c \geq 0$ and a period $p \in \{1, \ldots, t-1\}$ such that

$$cph \sum_{i=p+1}^{t} d_i \geq h \sum_{i=2}^{t}(i-1)d_i. \tag{1}$$

Then in an optimal solution

$$H_t = h \sum_{i=2}^{t}(i-1)d_i \leq cph \sum_{i=p+1}^{t} d_i \leq cK.$$

It follows that if we can find a c that satisfies (1) for all demand sequences $d_1, \ldots, d_t$, then we have found a bound on the total holding cost in an order interval with t periods. It is shown in Van den Heuvel and Wagelmans [1] that $c = \sum_{i=1}^{t-1} \frac{1}{i}$ satisfies (1) for all demand sequences. This leads to the following property:

Property 1 *Let periods $1, \ldots, t$ be an order interval in an optimal solution of a problem instance with demand sequence $d_1, \ldots, d_T$. Then the total holding cost H_t in the interval satisfies*

$$H_t = h \sum_{i=2}^{t}(i-1)d_i \leq K \sum_{i=1}^{t-1} \frac{1}{i}.$$

Since it is known that $\lim_{n \to \infty} \left(\sum_{i=1}^{n} \frac{1}{i} - \log n \right) = \gamma$ with $\gamma \approx 0.577$ a constant, the upper bound on the holding cost is proportional to the fixed order cost and the logarithm of the number of periods in the interval.

3   The class of heuristics

Property 1 suggests a new heuristic for the ELS problem. This heuristic selects an order interval that covers periods $1, \ldots, t$ with t the largest period that satisfies

$$h \sum_{i=2}^{t} (i-1)d_t \leq K \sum_{i=1}^{t-1} \frac{1}{i}.$$

In this way the solution possesses a property that is satisfied by any optimal solution. We note that this heuristic falls within the class of so-called on-line heuristics. In an on-line heuristic, one decides in period t whether to place a new order (or not), using only the information of periods $1, \ldots, t$. Hence, it is not allowed to change previous made decisions (in periods $1, \ldots, t-1$) or to use future information (beyond period t). Those heuristics often perform well in a rolling horizon environment.

   Unfortunately, the worst case performance of the above heuristic can be arbitrarily bad. However, we can modify the heuristic by considering a constant c in (1) that is dependent on the problem instance (instead of $c = \sum_{i=1}^{t-1} \frac{1}{i}$). It can be shown that the interpretation of this refined heuristic, called heuristic H, becomes as follows. Heuristic H chooses the order intervals as large as possible (except for possibly the last order interval) such that no additional order may improve the solution.

   Formally, let $C_1(t) = K + H_t$, let

$$C_2(t) = \min_{p=1,\ldots,t-1} \left\{ K + \sum_{i=2}^{p-1} (i-1)d_i + K + \sum_{i=p+1}^{t} (i-p)d_i \right\}.$$

and let $C_H(t) = C_1(t) - C_2(t)$. So $C_2(t)$ is the minimal cost of a solution with 2 orders in periods $1, \ldots, t$. If the last order is in period 1, then heuristic H places the next order in the first period $t > 1$ which satisfies $C_H(t) < 0$. It is shown in Van den Heuvel and Wagelmans [1] that heuristic H has the following properties:

Property 2

1. *Heuristic H has a worst case ratio of 2.*

2. *If n (n^*) is the number of orders generated by H (an optimal algorithm), then $\frac{1}{2}n^* \leq n \leq n^*$.*

   Although the first property shows that the worst case performance is good, the second property shows that the total number of orders may be (too) small compared to an optimal solution. Therefore, we combine heuristic H with a heuristic that generates relatively many orders, the Part Period Algorithm (PPA), which also has a worst case ratio of 2. The PPA starts a new order if total holding cost in the current

order interval exceeds the fixed order cost. Formally, let $C_{PPA}(t) = K - H_t$. Then PPA places the next order in the first period $t > 1$ which satisfies $C_{PPA}(t) < 0$.

We now propose a new class of heuristics by taking a 'convex combination' of heuristic H and PPA. Let $\lambda \in [0, 1]$ and let $C_\lambda(t) = \lambda C_{PPA}(t) + (1 - \lambda)C_H(t)$. Then for every $\lambda \in [0, 1]$ we define a heuristic H_λ, which places the next order in the first period $t > 1$ which satisfies $C_\lambda(t) < 0$. We will show that any heuristic H_λ in the class of heuristics has a worst case ratio of 2. Finally, we will test the performance of the new heuristics on some randomly generated problem instances.

References

[1] W. van den Heuvel and A.P.M. Wagelmans, A holding cost bound for the economic lot-sizing problem with time-invariant cost parameters, Operations Research Letters, 37, 102–106 (2009)

Finding Pareto optimal solutions for the Multi-Objective Economic Lot-Sizing Problem

Wilco van den Heuvel Econometric Institute, Erasmus University Rotterdam, The Netherlands
e-mail: `wvandenheuvel@ese.eur.nl`

H. Edwin Romeijn Department of Industrial and Operations Engineering, University of Michigan, USA
email: `romeijn@umich.edu`

Dolores Romero Morales Saïd Business School, University of Oxford, United Kingdom
email: `dolores.romero-morales@sbs.ox.ac.uk`

Albert P.M. Wagelmans Econometric Institute, Erasmus University Rotterdam, The Netherlands
e-mail: `wagelmans@ese.eur.nl`

1 Introduction

Consider a planning horizon of length T. For period t, let f_t be the setup lot-sizing cost, c_t the unit production lot-sizing cost and h_t the unit inventory holding lot-sizing cost. Similarly, for period t, let $\hat{f}_t$ be the setup emission, $\hat{c}_t$ the unit production emission and $\hat{h}_t$ the unit inventory emission. Let d_t be the demand in period t and M be a constant such that $M \geq \sum_{t=1}^{T} d_t$. Let us partition the planning horizon into consecutive blocks of ℓ periods. Without loss of generality, we assume that ℓ divides T, if not, we can define an equivalent problem by adding 'dummy' periods with demand equal to zero.

The *Multi-Objective Economic Lot-Sizing Problem with blocks of length ℓ (MOLS$^{(\ell)}$)* is a generalization of the classical *Economic Lot-Sizing Problem (ELSP)*, where we minimize the total lot-sizing costs across the planning horizon, as well as the emission of pollution in each block. The model reads as follows:

$$\text{minimize } \Big(\sum_{t=1}^{T}[f_t y_t + c_t x_t + h_t I_t], \Big(\sum_{t=(i-1)\ell+1}^{i\ell} [\hat{f}_t y_t + \hat{c}_t x_t + \hat{h}_t I_t] \Big)_{i=1}^{T/\ell} \Big)$$

86

subject to $\hspace{9cm}$ (MOLS$^{(\ell)}$)

$$
\begin{array}{rcll}
x_t + I_{t-1} & = & d_t + I_t & t = 1, \ldots, T \hspace{2cm} (1) \\
x_t & \leq & My_t & t = 1, \ldots, T \hspace{2cm} (2) \\
I_0 & = & 0 & \hspace{2.9cm} (3) \\
y_t & \in & \{0,1\} & t = 1, \ldots, T \\
x_t & \geq & 0 & t = 1, \ldots, T \\
I_t & \geq & 0 & t = 1, \ldots, T,
\end{array}
$$

where y_t indicates whether a setup has been placed in period t, x_t denotes the quantity produced in period t, and I_t denotes the inventory level at the end of period t. The first objective in (MOLS$^{(\ell)}$) models the usual *lot-sizing* costs. The second objective function models the total emission of pollution across the first block of ℓ periods, and similarly for the rest of objective functions. Constraints (1) model the balance between production, storage and demand in period t. Constraints (2) impose that production level is equal to zero if no setup is placed in period t. Constraints (3) impose that the inventory level is equal to zero at the beginning of the planning horizon. The last three constraints define the range in which the variables are defined.

When more than one objective function is optimized, Pareto optimal solutions are sought. For (MOLS$^{(\ell)}$), these can be found by minimizing, for instance, the lot-sizing costs, while constraining the block emissions. The Pareto optimal problem can be therefore written as an ELSP with additional constraints. When the lot-sizing cost function is concave, the classical ELSP is solvable in polynomial time in the length of the planning horizon T, see [8], while more efficient algorithms have been developed for special cases in [1, 3, 7]. In this talk, we show that the Pareto optimal problem for (MOLS$^{(\ell)}$) is $\mathcal{NP}$-complete. We then identify classes of problem instances for which Pareto optimal solutions can be obtained in polynomial time.

2 The Pareto optimal problem

Given $\hat{b} \in R$, the following problem defines a Pareto optimal solution for (MOLS$^{(\ell)}$):

$$
\text{minimize} \ \sum_{t=1}^{T} [f_t y_t + c_t x_t + h_t I_t]
$$

subject to $\hspace{8cm}(\mathcal{P}^{(\ell)}(\hat{b}))$

$$
\begin{array}{rcll}
x_t + I_{t-1} & = & d_t + I_t & t = 1, \ldots, T \\
x_t & \leq & M y_t & t = 1, \ldots, T \\
I_0 & = & 0 & \\
y_t & \in & \{0, 1\} & t = 1, \ldots, T \\
x_t & \geq & 0 & t = 1, \ldots, T \\
I_t & \geq & 0 & t = 1, \ldots, T \\
\sum_{t=(i-1)\ell+1}^{i\ell} [\hat{f}_t y_t + \hat{c}_t x_t + \hat{h}_t I_t] & \leq & \hat{b}^{(\ell)} & i = 1, \ldots, T/\ell. \qquad (4)
\end{array}
$$

$(\mathcal{P}^{(1)}(\hat{b}))$ and $(\mathcal{P}^{(T)}(\hat{b}))$ can be found in [2], where the emissions are constrained in each period of the planning horizon or across the whole planning horizon, respectively.

Clearly, if the emission constraints (4) are not binding, $(\text{MOLS}^{(\ell)})$ reduces to an ELSP and, therefore, is polynomially solvable. However, both $(\mathcal{P}^{(1)}(\hat{b}))$ and $(\mathcal{P}^{(T)}(\hat{b}))$ are $\mathcal{NP}$-complete, and so is $(\mathcal{P}^{(\ell)}(\hat{b}))$. Indeed, a special case of $(\mathcal{P}^{(1)}(\hat{b}))$ is the Capacitated Lot-Sizing Problem with non-stationary capacities, which is known to be $\mathcal{NP}$-complete, [4]. Using a special case of the PARTITION problem [5], we can also show that $(\mathcal{P}^{(T)}(\hat{b}))$ is $\mathcal{NP}$-complete.

3 Polynomial time algorithms for $(\mathcal{P}^{(\ell)}(\hat{b}))$

In the rest of the talk, we discuss classes of problem instances for which $(\mathcal{P}^{(\ell)}(\hat{b}))$ can be solved in polynomial time. These classes are summarized in Table 1. In the first column of the table, we give the structure of the lot-sizing costs, if any. In the next three columns, the assumptions on the lot-sizing emissions are detailed. Finally, the last column reports the running time of each algorithm.

LS costs	$\hat{f}_t$	$\hat{c}_t$	$\hat{h}_t$	running time
stationary f_t and c_t	stationary	stationary	$\hat{h}_t = \alpha h_t$	$\mathcal{O}(T^2 \ell^3)$
general	stationary	0	0	$\mathcal{O}(T^2 \hat{b}^{(\ell)})$
non-speculative motives	0	stationary	0	$\mathcal{O}(T^5)$

Table 1: Polynomially solvable classes of problem instances for $(\mathcal{P}^{(\ell)}(\hat{b}))$

4 Conclusions and future research

In this talk we have presented polynomial time algorithms for three classes of problem instances of $(\mathcal{P}^{(\ell)}(\hat{b}))$. In the future, we plan to describe the Pareto frontier.

References

[1] A. Aggarwal and J. K. Park. Improved algorithms for economic lot-size problems. *Operations Research*, 14:549–571, 1993.

[2] S. Benjaafar, Y. Li, and M. Daskin. Carbon footprint and the management of supply chains: Insights from simple models. Technical report, Industrial and Systems Engineering, University of Minnesota, 2010.

[3] A. Federgruen and M. Tzur. A simple forward algorithm to solve general dynamic lot sizing models with n periods in $\mathcal{O}(n \log n)$ or $\mathcal{O}(n)$ time. *Management Science*, 37:909–925, 1991.

[4] M. Florian and M. Klein. Deterministic procurement planning with concave costs and capacity constraints. *Management Science*, 26:669–679, 1971.

[5] 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.

[6] C. P. M. van Hoesel and A. P. M. Wagelmans. Parametric analysis of setup cost in the economic lot-sizing model without speculative motives. *International Journal of Production Economics*, 66:13–22, 2000.

[7] A. P. M. Wagelmans, C. P. M. van Hoesel, and A. Kolen. Economic lot sizing: An O($n \log n$) algorithm that runs in linear time in the Wagner-Whitin case. *Operations Research*, 40:S145–S156, 1992.

[8] H.M. Wagner. A postscript to dynamic problems of the theory of the firm. *Naval Research Logistics Quarterly*, 7:7–12, 1960.

A fixed-charge transportation view for solving big-bucket multi-item lot-sizing

Mathieu Van Vyve CORE, Université catholique de Louvain, Belgium
mathieu.vanvyve@uclouvain.be

1   Introduction

In the fixed charge transportation problem (FCT), we are given a set of depots $i \in I$, each with a quantity of available items a_i, and a set of clients $jinJ$, each with a maximum demand d_j. For each depot-client pair (i,j), both the unit profit $q_{i,j}$ of transporting one unit from t the depot to the client is known, together with the fixed charge $g_{i,j}$ of transportation along that arc. The goal is to find a profit-maximizing transportation program. Problem FCT can therefore be expressed as the following mixed-integer linear program:

$$
\begin{aligned}
\max \quad & \sum_{i \in I} \sum_{j \in J} (q_{i,j} w_{i,j} - g_{i,j} v_{i,j}), \\
& \sum_{j \in J} w_{i,j} \le a_i, \qquad \forall i \in I \\
& \sum_{i \in I} w_{i,j} \le d_j, \qquad \forall j \in J \\
& 0 \le w_{i,j} \le \min(C_i, D_j) v_{i,j}, \qquad \forall i \in I, j \in J, \\
& v \in \{0,1\}^{I \times J},
\end{aligned}
$$

where $w_{i,j}$ is a variable representing the amount transported from depot i to client j and $v_{i,j}$ is the associated binary setup variable.

In this description, the role of clients and depots are interchangeable. Indeed, this problem can be modelled as a bipartite graph where nodes are either depots or clients and edges between a depot and a client exist if the client can be served from that depot. A standard variant (and indeed, a special case) is when the demand of each client must be satisfied, in which case the unit profit is usually replaced by a unit cost.

As shown in Van Vyve [6] FCT has stong ties to the folllowing multi-item big-bucket lot-sizing problem (BBLS). The most natural way to formulate this problem as a mixed-integer program is to define variables x_t^i representing the number of products

i produced in period t and associated binary setup variables y_t^i. The problem is then formulated as

$$\min \quad \sum_{i \in I} \sum_{t=1}^{T} (p_t^i x_t^i + f_t^i y_t^i), \tag{1}$$

$$\sum_{t=1}^{k} x_t^i \geq \sum_{t=1}^{k} D_t^i \qquad \forall i \in I, k \in [1, T-1], \tag{2}$$

$$\sum_{t=1}^{T} x_t^i = \sum_{t=1}^{T} D_t^i \qquad \forall i \in I, \tag{3}$$

$$\sum_{i \in I} x_t^i \leq C_t \qquad \forall k \in [1, T], \tag{4}$$

$$0 \leq x_t^i \leq \min(\sum_{k=t}^{T} D_k^i, C_t) y_t^i \qquad \forall i \in I, t \in [1, T], \tag{5}$$

$$y_t^i \in \{0, 1\}, \qquad \forall i \in I, t \in [1, T]. \tag{6}$$

BBLS appears as a substructure in many production planning problems described in the literature, and has been tackeld as such by several authors [5, 4, 1]. FCT is simultaneously a special case of BBLS, and a strong relaxation of BBLS modelled using the facility location reformulation [3].

2 Our Contribution

We generalize the path-modular inequalities to inequalities of the following form. Let $T \subseteq A$ be a tree, let $L \subseteq T$ be a subpath and let $\overline{T} = T \setminus L$. Let $L = O_L \cup E_L$ be the partition of L such that no edges in O_L or E_L are adjacent to each other, and let $\mathcal{L} = (j_1, j_2, \ldots, j_{|L|})$ be a permutation of L. Let $O_L^{j_k} = \{j_1, \ldots, j_k\} \cap O_L$ and let $E_L^{j_k} = \{j_1, \ldots, j_{k-1}\} \cap E_L$. We call the the following inequality a tree-path-modular inequality:

$$\sum_{i \in T} x_i \leq \phi(O_L \cup \overline{T}) + \sum_{i \in E_L} \rho_i(O_L \cup E_L^i \cup \overline{T} \setminus O_L^i) y_i - \sum_{i \in O_L} \rho_i(O_L \cup E_L^i \cup \overline{T} \setminus O_L^i)(1 - y_i) \tag{7}$$

We prove the validity of these inequalities.

We use the following separating heuristic in a cut-and-branch framework. Given a fractional solution (x^*, y^*), parttition the edges into three sets: $(i, j) \in E_0$ if $x_{i,j} = 0$, $(i, j) \in E_1$ if $x_{i,j} = \min(a_i, b_j)$ and $(i, j) \in E_f$ otherwise. Grow a maximum spanning tree for $E_f \cup E_1$, using weights $1 - y_{i,j}$. At each iteration, select the subpath L of the tree T that leaves out the minimum number of fractional edges, and find a most violated tree-path-modular inequality for this choice of T and P.

We build a test set of instances already used in the literature, augmented by tight and difficult instances of BBLS and FCT. We report on computational experiments solving these instances by using the proposed heuristic separation procedure and valid inequalities. In particular we analyze the strength of the dual bound obtained at the root node with and without the new valid inequalities. We also compare with earlier work [5, 2].

References

[1] Kerem Akartunali and Andrew J. Miller. A heuristic approach for big bucket multi-level production planning problems. *European Journal of Operational Research*, 193(2):396–411 (2009).

[2] Jans, R., Degraeve, Z.: Improved lower bounds for the capacitated lot sizing problem with setup times. Operations Research Letters 32, 185195 (2004).

[3] J. Krarup and O. Bilde. Plant location, set covering and economic lotsizes: an $O(mn)$ algorithm for structured problems. In L. Collatz, editor, *Optimierung bei Graphentheoretischen und Ganzzahligen Probleme*, pages 155–180. Birkhauser Verlag, Basel (1977).

[4] Andrew J. Miller, George L. Nemhauser, and Martin W. P. Savelsbergh. On the polyhedral structure of a multi-item production planning model with setup times. *Math. Program.*, 94(2-3):375–405 (2003).

[5] W. W. Trigeiro, L. J. Thomas, and J. O. McClain. Capacitated lot sizing with setup times. *Management Science*, 35(3):353–366 (1989).

[6] M. Van Vyve. Fixed-charge transportation on a path: optimization, LP formulations, separation. CORE Discussion Paper 2010/68 (2010).

Author Index