A Decentralized Multi-Agent Coordination Method for Dynamic and Constrained Production Planning

Extended Abstract

Marin Lujak
IMT Lille Douai, Univ. Lille
Lille, France
marin.lujak@imt-lille-douai.fr

Alberto Fernández
Universidad Rey Juan Carlos
Madrid, Spain
alberto.fernandez@urjc.es

Eva Onaindia
Valencian Research Institute for AI
Valencia, Spain
onaindia@dsic.upv.es

ABSTRACT

In the capacitated production planning problem, quantities of products need to be determined at consecutive periods within a given time horizon when product demands, costs, and production capacities vary through time. We focus on a general formulation of this problem where each product is produced in one step and setup cost is paid at each period of production. Additionally, products can be anticipated or backordered in respect to the demand period. We propose a computationally efficient decentralized approach based on the *spillover effect* relating to the accumulation of production costs of each product demand through time. The performance of the spillover algorithm is compared against the state-of-the-art mixed integer programming branch-and-bound solver CPLEX 12.8 considering optimality gap and computational time.

KEYWORDS

Coordination and control models for multi-agent systems; multi-agent planning and scheduling; distributed problem solving; self-organisation

ACM Reference Format:
Marin Lujak, Alberto Fernández, and Eva Onaindia. 2020. A Decentralized Multi-Agent Coordination Method for Dynamic and Constrained Production Planning. In *Proc. of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020), Auckland, New Zealand, May 9–13, 2020,* IFAAMAS, 3 pages.

1 INTRODUCTION

Production planning considers the best use of resources to satisfy the production demand over a time horizon while guaranteeing service quality and minimizing production and inventory costs. In contexts of a production demand that varies through time and scarcity of resources to satisfy the demand, the dynamic capacitated lot-sizing problem (CLSP) addresses questions like when and how much to produce of each product such that overall production costs are minimised [2, 11]. In this paper, we focus on deterministic, single-level, multi-item CLSP with backorders and independent setup costs (MCLSP-BO). That is, we address the problem of producing multiple items from raw material with no intermediate subassemblies using *backordering* [16] (possibility of satisfying the demand of the current period in future periods).

Proc. of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020), B. An, N. Yorke-Smith, A. El Fallah Seghrouchni, G. Sukthankar (eds.), May 9–13, 2020, Auckland, New Zealand.

Anticipating or delaying production planning is crucial in contexts of a very high demand [15, 21]. The main question in MCLSP-BO is which items to backorder, which ones to produce at the time of demand release, and which ones beforehand [6, 10]. This is an NP-hard problem [1] which has been mostly addressed with heuristic approximations like the improvement heuristics, which start from an initial (often infeasible) solution for the complete planning horizon and then try to enforce feasibility conditions by shifting lots from period to period at minimal extra cost [2, 13]. There also exist meta-heuristic approaches based on the linear programming relaxation of a shortest path formulation of the same problem, tabu search algorithms [12], heuristic algorithms based on local optima [4] that examine the Lagrangean relaxation and design heuristics to generate upper bounds within a subgradient optimization procedure [19], or genetic algorithms that use fix-and-optimize heuristic and other mathematical programming techniques [20].

Existing approaches to MCLSP-BO are centralized solutions that apply to only one decision maker. As such, they are not adaptable to address intrinsically decentralized scenarios like those in supply chain management, surgical scheduling of patients to a network of private hospitals, aircraft arrival planning, or telecommunications packet scheduling. We propose a decentralized multi-agent algorithm based on the *spillover effect*, which is defined as a situation that starts in one place and expands or has an effect elsewhere [18]. Particularly, the spillover effect attempts to find the best allocation of resources to each item demand and time period based on the ordering of accumulated costs, and spreads the non-allocated demand over the time horizon. We leverage the spillover effect to design, to the best of our knowledge, the first decentralized algorithmic solution approach to the MCLSP-BO problem.

2 PROBLEM FORMULATION

The MCLSP-BO belongs to the class of deterministic dynamic lot-sizing problems well known in the inventory management literature [2, 10, 11, 14, 16, 17]. The objective is to find a production schedule for a set I of items minimizing the total backorder, holding inventory, production, and independent setup costs over a finite time horizon $T = \{1, \ldots, |T|\}$ subject to demand and capacity constraints. Incoming orders are demands d_{it} for item $i \in I$ at each time period $t \in T$. The number of resources available at the beginning of period t is given by R_t. We assume a time-varying setup cost s_{it} and a linear production cost c_{it} for item i produced at time t. Demand d_{it} can be anticipated or delayed in regards to the requested time t. If anticipated, it is at the expense of a linear holding cost $h_{it'}, t' < t$, accrued per each anticipated time unit; if delayed, a linear backorder cost $b_{it'}, t' > t$, is accrued for every delayed time unit. There is a buffer associated to each combination item-time such that its content is increased by production quantity u_{it} and reduced by demand d_{it}. Hence, if $u_{it} = d_{it}$, there is no stock nor backorder of i at t; if $u_{it} > d_{it}$ there is stock of i at t; and if $u_{it} < d_{it}$ there is a backorder of demands of item i at time t. We represent the stock level as x^+_{it} and the backorder level as x^-_{it} and note that $x^+_{it} \cdot x^-_{it} = 0$ since no stock and backorder can coexist at the same time.

The single CLSP is NP-hard for many special cases [1, 7] and the MCLSP is proved to be strongly NP-hard [3]. Unlike the MCLSP model presented in [1], which does not feature integrality constraints for variables $u_{it}, x^+_{it}, x^-_{it}$, we assume non-negative integer values for these variables, thus leading to a generally NP-complete integer programming problem.

3 THE SPILLOVER ALGORITHM

The spillover algorithm builds upon an iterative auction model. The demand of item i at time t is represented by an autonomous *liquid* agent $a = (i, t)$ that solves a single-item CLSP and bids for the allocation of demand d_a. Unlike models that use a single resource allocating agent [8, 9], we propose a decentralized approach with multiple time *container* agents, one per each period $k \in T$ constrained by available resources (capacity) R_k. Hence, each one of $|A| = |I| \cdot |T|$ liquid agents bids for allocation of its own demand d_a in periods $k \in T$ with locally minimum cost. Each container agent $k \in T$ allocates its available resources to bidders ensuring that $\sum_a u_{ak} \le R_k$; i.e., the sum of allocated demands remains bounded by its capacity R_k. A liquid agent (i, t) is thus regarded as having a chamber chamber^i_k at each container k whose volume $u_{ak} \ge 0$ is dependent of the volumes of other liquid agents allocated to the container and upper-bounded by the container's capacity R_k.

Liquid agent a iteratively requests resource allocation in container agents $k \in T$ until its demand d_a doesn't get completely allocated, i.e., $\sum_{k \in T} u_{ak} = d_a$ or it hasn't bid for all $k \in T$. Initially, this demand is released at container $k = t$, which is modelled as a rational collaborative agent that controls the a's demand through valve c^a_k (production u_{ak}). Depending on how much liquid of d_a is produced at t, agent a negotiates advancing or postponing the demand with the containers before or after t through two types of valves, a valve to control the flow of demand of a to the posterior time periods $k \in \{t + 1, \dots, |T|\}$ and a valve to control the flow of liquid agent (i, t) to time periods prior t; i.e., $k \in \{t - 1, \dots, 1\}$. Figure 1 shows the flow of demand of a particular item $i = 1$ at container t, across subsequent containers $t + 1, t + 2, \dots$, and prior containers $t - 1, t - 2$, etc.

The anytime spillover algorithm iteratively runs multiple auction-based negotiations between liquid and container agents. Containers announce their available resources and liquid agents bid for available containers with locally lowest cost. Then containers assign the demand of liquid agents that locally maximize their social welfare. It is essentially a decentralized heuristic approach that spills the demand d_{it} of liquid agent (i, t) over neighbouring containers due to limited capacity of container t to satisfy such demand; and this applies to every item i at every time period t. Generally speaking, given a liquid agent $a = (i, t)$, the direction and quantity of

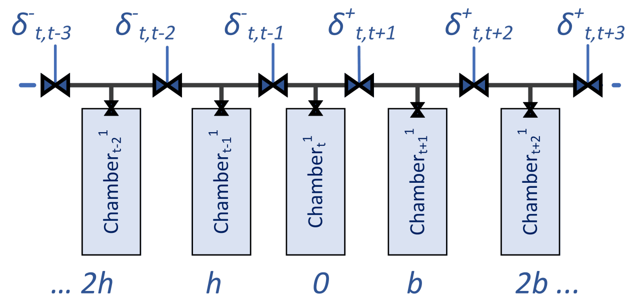

Figure 1: Flow of demand of a liquid agent $(1, t)$

spillage depends on: (1) *accumulated unit production cost (UPC_{ak})* for time period k: Eq. (1) (set up cost and production cost at k plus accumulated holding and backorder costs if any); and (b) *estimated accumulated cost (EAC_a)*: Eq. (2), where $\Gamma_a \subset T$ is the set of available containers that have capacity to produce at least one unit of their product.

$$UPC_{ak} = \begin{cases} c_{ik} + s_{ik} + \sum_{m=k}^{t-1} h_{im}, & \forall k \in T | k < t \\ c_{ik} + s_{ik}, & \text{for } k = t \\ c_{ik} + s_{ik} + \sum_{m=t+1}^{k} b_{im}, & \forall k \in T | k > t \\ M \cdot \sum_{m=t+1}^{k} b_{im}, & \text{for } k = |T| + 1 \end{cases} \quad (1)$$

$$EAC_a = \sum_{k \in \Gamma_a \cup \{|T|+1\}} UPC_{ak} \quad (2)$$

4 EXPERIMENTS AND RESULTS

We compared the performance of our spillover algorithm, the first decentralized approach to the MCLSP-BO – to the best of our knowledge –, with a centralized and optimal CPLEX solution over randomly generated and diversified problem set for large-scale capacitated lot-sizing. The experiment parameters are generated similarly to [5] and [9]. Table 1 shows the optimality gap in relation to the centralized solution found by CPLEX and the computational time of the Spillover algorithm. The average optimality gap is 25% and we can observe it is not generally dependent on the number of items. The computational time is less than 0.1 in all experiments, which contrasts with the exponentially increasing time of CPLEX. Besides the significant results of the algorithm, we also highlight other benefits of decentralizing MLCSP-BO such as keeping the minimum possible exposition of private agent information.

Table 1: Summary of the computational results

# items	50	60	70	80	90	100	110	120	130	140	150
Avg. Gap (%)	28	28	30	16	16	28	27	28	27	27	27
CPU time (ms)	11	12	18	16	18	19	23	24	27	28	30

Acknowledgements. This work is supported by: the Spanish MINECO projects RTI2018-095390-B-C33 (MCIU/AEI/FEDER, UE) and TIN2017-88476-C2-1-R, the French ADEME project E-Logistics, and an STSM Grant funded by the European ICT COST Action IC1406, cHiPSet.

REFERENCES

[1] Gabriel R Bitran and Horacio H Yanasse. 1982. Computational complexity of the capacitated lot size problem. *Management Science* 28, 10 (1982), 1174–1186.

[2] Lisbeth Buschkühl, Florian Sahling, Stefan Helber, and Horst Tempelmeier. 2010. Dynamic capacitated lot-sizing problems: a classification and review of solution approaches. *Or Spectrum* 32, 2 (2010), 231–261.

[3] W-H Chen and J-M Thizy. 1990. Analysis of relaxations for the multi-item capacitated lot-sizing problem. *Annals of operations Research* 26, 1 (1990), 29–72.

[4] C H Cheng, M S Madan, Y Gupta, and S So. 2001. Solving the capacitated lot-sizing problem with backorder consideration. *Journal of the Operational Research Society* 52, 8 (2001), 952–959.

[5] Moustapha Diaby, Harish C Bahl, Mark H Karwan, and Stanley Zionts. 1992. A Lagrangean relaxation approach for very-large-scale capacitated lot-sizing. *Management Science* 38, 9 (1992), 1329–1340.

[6] Andreas Drexl and Alf Kimms. 1997. Lot sizing and schedulingâĂŤsurvey and extensions. *European Journal of operational research* 99, 2 (1997), 221–235.

[7] Michael Florian, Jan Karel Lenstra, and AHG Rinnooy Kan. 1980. Deterministic production planning: Algorithms and complexity. *Management science* 26, 7 (1980), 669–679.

[8] Stefano Giordani, Marin Lujak, and Francesco Martinelli. 2009. A decentralized scheduling policy for a dynamically reconfigurable production system. In *International Conference on Industrial Applications of Holonic and Multi-Agent Systems*. Springer, 102–113.

[9] Stefano Giordani, Marin Lujak, and Francesco Martinelli. 2013. A distributed multi-agent production planning and scheduling framework for mobile robots. *Computers and Industrial Engineering* 64, 1 (2013), 19–30.

[10] Raf Jans and Zeger Degraeve. 2008. Modeling industrial lot sizing problems: a review. *International Journal of Production Research* 46, 6 (2008), 1619–1643.

[11] Behrooz Karimi, SMT Fatemi Ghomi, and JM Wilson. 2003. The capacitated lot sizing problem: a review of models and algorithms. *Omega* 31, 5 (2003), 365–378.

[12] Behrooz Karimi, S. M. T. Fatemi Ghomi, and John M. Wilson. 2006. A tabu search heuristic for solving the CLSP with backlogging and set-up carry-over. *JORS* 57, 2 (2006), 140–147.

[13] Johan Maes and Luk Van Wassenhove. 1998. Multi-Item Single-Level Capacitated Dynamic Lot-Sizing Heuristics: A General Review. *The Journal of the Operational Research Society* 39, 11 (1998), 991–1004.

[14] Harvey H Millar and Minzhu Yang. 1994. Lagrangian heuristics for the capacitated multi-item lot-sizing problem with backordering. *International Journal of Production Economics* 34, 1 (1994), 1–15.

[15] Yves Pochet and Laurence A Wolsey. 1988. Lot-size models with backlogging: Strong reformulations and cutting planes. *Mathematical Programming* 40, 1-3 (1988), 317–335.

[16] Yves Pochet and Laurence A. Wolsey. 2006. *Production Planning by Mixed Integer Programming*. Springer Series in Operations Research and Financial Engineering.

[17] Daniel Quadt and Heinrich Kuhn. 2008. Capacitated lot-sizing with extensions: a review. *4OR* 6, 1 (2008), 61–83.

[18] Tatyana A Rand, Jason M Tylianakis, and Teja Tscharntke. 2006. Spillover edge effects: the dispersal of agriculturally subsidized insect natural enemies into adjacent natural habitats. *Ecology letters* 9, 5 (2006), 603–614.

[19] Haldun Süral, Meltem Denizel, and Luk N. Van Wassenhove. 2009. Lagrangean relaxation based heuristics for lot sizing with setup times. *European Journal of Operational Research* 194, 1 (2009), 51–63.

[20] Claudio Fabiano Motta Toledo, Renato Resende Ribeiro de Oliveira, and Paulo Morelato França. 2013. A hybrid multi-population genetic algorithm applied to solve the multi-level capacitated lot sizing problem with backlogging. *Computers & OR* 40, 4 (2013), 910–919.

[21] Willard I Zangwill. 1966. A deterministic multi-period production scheduling model with backlogging. *Management Science* 13, 1 (1966), 105–119.