2015 IEEE Symposium Series on Computational Intelligence

A Collaborative Lot-Sizing Problem with Production Limitations

Mario Ziebuhr
Chair of Logistics
University of Bremen
28334 Bremen, Germany
ziebuhr@uni-bremen.de

Tobias Buer
Computational Logistics – Cooperative Junior Research Group of University of Bremen and ISL – Institute of Shipping Economics and Logistics
28334 Bremen, Germany
tobias.buer@uni-bremen.de

Herbert Kopfer
Chair of Logistics
University of Bremen
28334 Bremen, Germany
kopfer@uni-bremen.de

***Abstract*—Manufacturing companies are using collaborative planning for the coordination of lot-sizing decisions in inter-organisational supply chains. By using collaborative planning, the members of a supply chain try to identify a production plan which results in lower costs compared to individual plans by simultaneously preserving their autonomy. A distributed lot-sizing problem with rivaling agents (DULR) is studied where each item can be produced by more than one member (agent) of the coalition. Thereby, it occurs that agents compete for the production quotas of items. However, the goal of this contribution is to extend the DULR by considering two types of items. One type can be produced by more than one agent, while the other one can only be produced by an appointed agent due to contractual obligations. We denote the former type of items as concurrent items and the latter one as compulsory items. To solve the DULR with different types of items, an existing negotiation mechanism based on simulated annealing is applied and modified. A benchmark study shows that the modified solution approach outperforms the best-known approach for the DULR. Based on this finding, a second study is applied where the impact of compulsory items is investigated for the DULR.**

I. Introduction

Material resource planning of an inter-organisational supply chain with multiple decision makers is crucial due to small margins in many industries. An important subproblem of material resource planning is lot-sizing. Over a planning horizon of multiple periods, a decision maker (referred to as *agent*) has to determine the production quantity of each item in each period. The goal is to minimize the total production costs over the planning period [1]–[3]. Hereby, the decision maker has to balance setup and inventory holding costs for every item. Main challenges for lot-sizing in a supply-chain with multiple decision makers are asymmetric and private information of the decision makers and the selfish goal of each decision maker to minimize the own costs (referred to as *local costs*). To deal with these issues a collaborative planning approach based on a negotiation mechanism is applied. Related approaches are also discussed by [4]–[9]. Collaborative planning enables the decision makers to preserve their autonomy and coordinating their local plans in order to achieve a joint (global) production plan that is superior compared to non-coordinated planning [10], [11].

Point of origin for this paper is the distributed multi-level uncapacitated lot-sizing problem with rivaling agents (DULR) which was introduced by Buer et al. [12] and Eslikizi et al. [13]. The DULR considers many features of real-world supply chains like a multi-level product structure, setup and inventory holding decisions, and multiple decision makers. Therefore, it can be seen as a representative planning problem for lot-sizing in supply chains. The DULR generalizes the distributed multi-level uncapacitated lot-sizing problem (DMLUSLP, [14]–[16]). In the DULR and in the DMLULSP each decision maker is able to produce a subset of items. In the DMLULSP these subsets are *disjoint*, i.e., every item is produced by only one agent. The agents compete for the effective production date of their items but not on the production quantity of each item over the planning period. However, in the DULR the assignment of items to agents is no longer disjoint. Some items may be produced by more than one agent, these items are denoted as *concurrent items*. With respect to concurrent items, the agents compete on the production quantity over the planning horizon. Clearly, this increases competition and stresses the agents to preserve private information during the coordination of planning. While the DULR introduced by Buer et al. [12] considers only a subset of items as concurrent, Eslikizi et al. [13] consider all items as concurrent.

We introduce the *DULR with Production Limitations* (DULR-PL). Like the DULR, the DULR-PL considers concurrent items but also compulsory items. *Compulsory items* have to be produced by an appointed agent. The reasons for having compulsory items are for example safety concerns or premium goods for which a manufacturing company has to be able to distinguish between the type of service performed on the item. In this paper, two service types are considered: a standard service and a premium service. If a standard service is requested for an item, this item can be produced by any member of the coalition, while an item with a premium service has to be produced by an appointed agent. Depending on the service, items differ with respect to their impact on costs and production plans. The corresponding items of these services are denoted as concurrent and compulsory items.

Collaborative planning is also relevant in other logistic areas, e.g., transportation planning [17]–[21]. However, different types of items like compulsory items are hardly discussed. These types of items are prohibited to be fulfilled by a different company because of contractual obligations [22]. In transportation planning items are usually denoted as requests. The impact of different types of requests has already been addressed in transportation planning. Schönberger [23] studies compulsory requests for a pickup and delivery problem with time windows (PDPTW) and two different modes of transportation (own fleet and common carriers). Ziebuhr and Kopfer [24] analyze the impact for a PDPTW with three different transportation modes (own fleet, long-term carriers, and common carriers). In terms of distributed lot-sizing in supply chains, there is a lack of approaches which consider different types of items.

Therefore, the goal of this paper is to study the effects of compulsory items on the total costs of a DULR. The DULR is extended by considering concurrent items and compulsory items at the same time. To solve the DULR-PL an existing negotiation mechanism is modified and applied. The idea of the negotiation approach is that in each negotiation round a mediator proposes a binary encoded production plan to the decision makers. If all agents approve the proposal then the proposal is accepted. However, slightly worse proposals may also be approved, because a simulated annealing acceptance criterion is applied. We extend the negotiation mechanism by two features. The first modification reduces the variation range of the solution quality by modifying the binary coded production plan. The second modification ensures that only feasible production plans for the DULR-PL are generated.

The remaining paper is structured as follows. Section II describes the DULR-PL. Section III extends a negotiation mechanism from the literature to solve the DULR-PL. Section IV presents results of computational studies. Finally, Section V concludes the paper and gives some ideas for future research.

II. Distributed Lot-Sizing with Compulsory Items

The DULR-PL is jointly solved by multiple independent decision makers who have to coordinate their lot-sizing decisions over multiple planning periods in order to meet the given customer demand for each product in each period. In the DULR-PL a set A of agents is given who jointly produce a set I of m items. Agent $a \in A$ produces the set of items I_a with $I = \bigcup_{a \in A} I_a$. In the DULR-PL and in contrast to earlier approaches like [14] or [12], the allocation of items to agents is usually non-disjoint, i.e., $\bigcap_{a \in A} I_a \neq \emptyset$. An item is denoted as *concurrent item*, if more than one agent is able to produce it. Let I^c denote the set of *concurrent* items and let $I^c_a \subset I^c$ denote the set of concurrent items of agent $a \in A$. The existence of concurrent items complicates the coordination problem significantly. Furthermore, there are items denoted as *compulsory items* which have to be produced by an appointed agent. Let I^d denote the set of *compulsory* items and let $I^d_a \subset I^d$ denote the set of compulsory items of agent $a \in A$. The following relationship between concurrent items and compulsory items holds for each instance of the DULR-PL: $I = I^c \cup I^d$ and $I^c \cap I^d = \emptyset$.

The DULR-PL covers also aspects of typical dynamic multi-level lot-sizing problems. The considered lot-sizing problem is dynamic because the set of items I has to be produced during a set T of n periods. Among the items there are interdependencies, which are defined by a multi-level product structure (i.e. a gozintograph) which defines the composition of end products by intermediate products and raw materials. For each item $i \in I$ the set of all direct successors is given by the set $\Gamma^+(i)$ while the set of all direct predecessors is given by the set $\Gamma^-(i)$. Final products have an empty set of successors, raw materials have an empty set of predecessors. The production coefficient r_{ij} indicates the required units of item i to produce one unit of item j. Without loss of generality, $r_{ij} = 1$ is assumed. For all final products an exogenous demand $d_{ita} \geq 0$ is given ($i \in I, t \in T, a \in A$). Furthermore, for each $i \in I$ and each $a \in A$ the setup costs s_{ia}, the inventory holding costs h_{ia}, and a function to calculate the unit costs $u_{ai}(x_{ait})$ are given.

There are five types of decision variables. The binary variable y_{ita} indicates whether a machine setup for item $i \in I$ occurs in period $t \in T$ by agent $a \in A$ ($y_{ita} = 1$) or not ($y_{ita} = 0$). Variable $x_{ita} \geq 0$ indicates the lot-size of item $i \in I$ in period $t \in T$ produced by agent $a \in A$. Furthermore, we have to decide about the internal demand $d_{ita} \geq 0$ ($i \in I$ with $\Gamma^+(i) \neq \emptyset, t \in T, a \in A$) and the inventory $l_{ita} \geq 0$ ($i \in I, t \in T, a \in A$). Finally, the last family of decision variables al_{ai} allocates the relative production quantity of each item $i \in I$ to an agent $a \in A$ (shared production). The relative production quantity of an item, i.e., the *production quota*, is effective during the entire planning horizon. For example, a production quota of $al_{ia} = 0.35$ means that agent a is awarded 35 percent of the production quantity of item i over all periods.

In Buer et al. [12] the complete production quantity of an item was always assigned to only one agent, splitting the production quantity to several agents has not been supported. The reason lies in the used objective function for the multi-agent case, which is closely adapted from the single-agent case of the multi-level uncapacitated lot-sizing problem. This traditional objective function does not have to consider unit costs because there is only one (central) agent and the required quantity of every item is exogenously given; furthermore, due to a lack of production capacities (unlike, e.g. [5], [6], [25]), overtime costs of machines are not considered in the objective function. For such an objective function, it is not reasonable to allocate the production quantity of an item to more than one agent. To solve this issue in Eslikizi et al. [13] it is proposed to extend the objective function by integrating unit costs and reducing the existing setup costs. In the DULR-PL we use f_a as the local cost function of agent $a \in A$, which is the same as presented in Eslikizi et al. [13].

$$\min f_a = \sum_{i \in I_a} \sum_{t \in T} (s_{ia} \cdot y_{ita} + h_{ia} \cdot l_{ita} + u_{ia}(x_{ita})) \qquad (1)$$

The local costs f_a of agent a consist of all setup costs, all inventory holding costs, and the variable production costs. The variable production costs are determined by the function $u_{ia}(x_{ita})$ which implicitly takes machine capacities into account, because it assumes that the unit costs u_{ia} increase by the factor α when the production quantity x_{it} exceeds a given threshold $\overline{d}_i$ ($i \in I$). For the sake of simplicity, we simply assume $\alpha = 2$ in the remainder. For an actual instance of the DULR-PL the threshold value $\overline{d}_i$ is calculated as the average demand per item and per period. The variable cost function is defined as:

$$u_{ia}(x_{ita}) = \begin{cases} u_{ia} x_{ita} & \text{if } x_{ita} \leq \overline{d}_i \\ u_{ia}\overline{d}_i + \alpha u_{ia}(x_{ita} - \overline{d}_i) & \text{if } x_{ita} > \overline{d}_i \end{cases} \qquad (2)$$

Each agent $a \in A$ wants to minimize the local costs f_a, while the constraints (3) to (11) have to be satisfied:

$$l_{ita} = l_{i,t-1,a} + x_{ita} - d_{ita}\,, \quad \forall i \in I_a, \forall t \in T, \tag{3}$$

$$l_{ioa} = 0\,, \quad \forall i \in I_a, \tag{4}$$

$$l_{ita} \geq 0\,, \quad \forall i \in I_a, \forall t \in T \setminus \{o\}, \tag{5}$$

$$\begin{gathered} d_{ita} = (\sum_{j \in \Gamma^+(i)} r_{ij} \cdot x_{j,t+t_i,a}) \cdot al_{ia}\,, \\ \forall i \in \{j \in I_a \mid \Gamma^+(j) \neq \emptyset\},\, t \in T, \end{gathered} \tag{6}$$

$$al_{ia} = 1\,, \quad \forall i \in I_a^d, \tag{7}$$

$$x_{ita} - M \cdot y_{ita} \leq 0\,, \quad \forall i \in I_a, \forall t \in T, \tag{8}$$

$$x_{ita} \geq 0\,, \quad \forall i \in I_a, \forall t \in T, \tag{9}$$

$$al_{ia} \geq 0\,, \quad \forall i \in I_a^c, \tag{10}$$

$$y_{ita} \in \{0, 1\}\,, \quad \forall i \in I_a, \forall t \in T. \tag{11}$$

The inventory balance constraint (3) ensures that the inventory l_{ita} of item i at the end of the current period t is determined by the inventory of the previous period $t-1$ and the amount x_{ita} produced in the current period minus the demand for item i in the current period. For all items, the inventory of the first period $t = 0$ is zero (4) and for remaining periods non-negative (5). The endogenous demand for the components and raw materials are determined by constraint (6) where the shared production is taken into account by the allocation parameter al_{ia}. The latter constraints ensure that the shared production of item j in period $t + t_i$ triggers a corresponding demand d_{ita} for all $i \in \Gamma^-(j)$, that means that there is a demand for each item i preceding item j in the multi-level item structure. The lot-size x_{ita} is defined by constraint (8) and (9). By constraint (8) it is ensured that if an agent a produces an item i in period t then the binary variable y_{ita} is one and otherwise zero. Obviously, lot-sizes cannot be negative which is defined by constraint (9). In terms of the shared production, constraint (7) ensures that all units of a compulsory item $i \in I_a^d$ have to be produced by agent a while constraint (10) ensures that the allocation parameter al_{ai} cannot be negative for concurrent items.

The model (1) to (11) takes the point of view of a single decision maker. When we consider the group decision variant of the DULR-PL the following constraint family has to be met additionally. It ensures that the production quota for each item sum up to 100 percent:

$$\sum_{a \in A} al_{ia} = 1\,, \forall i \in I^c. \tag{12}$$

Furthermore, the goal is to find and agree on a joint *production plan* $p := ((y_{ita}), (al_{ia}))$ which minimizes the global supply chain costs. The global costs are defined as the sum of the local costs of each agent:

$$\min f(p) = \sum_{a \in A} f_a(p) \tag{13}$$

Generally, minimizing the local costs of each agent and minimizing the global costs of the supply chain are conflicting goals. Therefore, we propose a collaborative planning approach based on negotiations.

III. Negotiation of Lot-Sizing Contracts via Simulated Annealing

To solve the DULR-PL a simulated annealing negotiation mechanism is applied. A simulated annealing is a metaheuristic based on local search [26] which can be used to escape from local optima by allowing moves that deteriorate the objective function value. The applied negotiation mechanism evaluates lot-sizing contracts with a shared production via a simulated annealing and was introduced by Eslikizi et al. [13]. The mechanism is known as SA. In this paper, the SA is extended by a new procedure for identifying a promising shared production among the members of the coalition as well as in case of a procedure for handling compulsory items. The extended SA is denoted as SAA.

The SAA is based on a negotiation mechanism which was introduced by Homberger [14] for solving the DMLULSP and was extended by Buer et al. [12] and Eslikizi et al. [13] for solving the DULR. The difference between these approaches for the DULR is that in Buer et al. [12] agents compete for the production of an item where the favorable agent gets the whole production volume of an item. In Eslikizi et al. [13] multiple agents produce the same item. By applying the SA for the DULR, we identified that the quality of the solutions as well as the fluctuation range of the approach can be improved by slightly changing the shared production procedures of the SA.

The SAA is controlled by a neutral mediator and is outlined in Algorithm 1. At the beginning of the algorithm, an initial production plan p has to be determined by the mediator. In our scenario, a production plan is defined by an allocation parameter al and a contract c. The allocation parameter defines the fraction of items that are produced by rivaling agents while the contract represents an encoded solution of the setup decision for each item of the coalition. In our scenario, a contract c has three dimensions, it specifies if an agent $a \in A$ is allowed to produce a specific item $i \in I$ in a given period $t \in T$ or not. The first contract is generated randomly while the initial allocation is generated by splitting the fraction of every item equally among the rivaling agents which means in a scenario with two rivaling agents that both agents produce 50% of the demanded units of an item. In Eslikizi et al. [13] the initial allocation parameter is generated by identifying the best allocation for each item $i \in I$ based on the initial contract c. We realized that this procedure cannot be recommended because the initial contract is generated randomly and therefore might lead into a local optima. Based on the initial production plan p, each agent $a \in A$ evaluates p by the local cost function $f_a(p)$ and determines the cooling schedule τ_a. For more details of determining cooling schedules, it is referred to Homberger [14].

As soon as the cooling schedules are determined, the negotiation phase takes place where in each round r^n out of r^n_{max} negotiation rounds a new proposal for a production plan p' is generated by the mediator. A new allocation parameter al' is generated by slightly updating the allocation parameter al (see Algorithm 2). A new contract c' is generated by simultaneously flipping one element of the contract c for each rivaling agent. Based on the updated allocation parameter al' and contract c', each agent $a \in A$ evaluates the production plan p' by the local cost function $f_a(p')$. An agent accepts a production plan p'

if the plan reduces the local costs or by a specific probability $P_a(p, p', r)$ if the production plan increases the local costs. In [13], [27] it is proposed to consider additional side payments based for the evaluation of a production plan, however, they are difficult to compute under asymmetric information and are not directly related to compulsory items, therefore we do not use side payments in this approach. If all agents accept the production plan p', the contract c' and the allocation parameter al' will be accepted as c and al and can be used to generate a new contract and allocation parameter. After a specific number of rounds, the accepted contract is used for an allocation improvement procedure where the best allocation for each item $i \in I$ is determined (see Algorithm 3). In each negotiation round, the individual temperatures T_a are updated corresponding to the schedule τ_a. The negotiation phase is terminated as soon as each negotiation round is investigated. At the end, the best mutually accepted contract and allocation parameters are returned.

Algorithm 1 SAA (cf. Eslikizi et al. [13])

Data: problem data, allocation parameter (si^{al}, oq^{al})
mediator: generate initial allocation al
mediator: generate initial contract c
each $a \in A$: evaluate p by local objective function $f_a(p)$
each $a \in A$: compute cooling schedule τ_a
while $r^n < r^n_{max}$ **do**
  mediator: generate a new allocation $al' \leftarrow N(al)$
  mediator: generate a new contract $c' \leftarrow N(c)$
  each $a \in A$: evaluate p' with $f_a(p')$
  each $a \in A$: accept p' with probability $P_a(p, p', r)$
  if all agents accept the production plan p' **then**
    mediator: update contract $c \leftarrow c'$
    mediator: update allocation $al \leftarrow al'$
    if allocation improvement is activated **then**
      mediator: identify the best allocation al
    end if
  end if
  each $a \in A$: update temperature T_a
  mediator: $r^n \leftarrow r^n + 1$
end while
return mutually accepted contract and allocation

As mentioned, the shared production procedures of Eslikizi et al. [13] are modified with the goal to improve the quality of the SAA solution. In Eslikizi et al. [13] it is proposed to update the allocation parameter after a specific number of negotiation rounds. We identified that the allocation parameters should be slightly updated like the contract in each negotiation round. Furthermore, we propose to use the allocation improvement procedure during the negotiation instead of using it for identifying a promising initial allocation parameter like in the SA.

In each negotiation round of the SAA, the allocation parameter al is updated by slightly changing some of the item allocations. The pseudocode is represented by Algorithm 2. Thereby, the mediator chooses a subset $I^{al} \subset$ I of items randomly. The size of I^{al} is defined by the given parameter si^{al} and the change of order quantity oq^{al}. Each item i in I^{al} is investigated for the modification of the allocation. Corresponding to the selected item the rivaling agents a_1 and a_2 are identified with their current best allocation al_i and $1 - al_i$. In a next step, the allocation is increased or decreased randomly which is defined by the operator o^{al}. The allocation updating procedure terminates when each item has been looked at. The procedure is repeated in each negotiation round.

Algorithm 2 Updating the allocation parameter

Data: problem data, si^{al}, oq^{al}, al_i
mediator: choose items and store them in I^{al}
for $i \in I^{al}$ **do**
  mediator: identify rivaling agents A_i^{al}
  mediator: choose operation $o^{al} \in \{0, 1\}$ randomly
  if $(o^{al} = 1 \vee al_i + oq^{al} \leq 100) \wedge (o^{al} = 0 \vee al_i - oq^{al} \leq 0)$ **then**
    mediator: update $al_i \leftarrow al_i + oq^{al}$
  end if
  if $(o^{al} = 1 \vee al_i^b + oq^{al} \geq 100) \wedge (o^{al} = 0 \vee al_i - oq^{al} \geq 0)$ **then**
    mediator: update $al_i \leftarrow al_i - oq^{al}$
  end if
  return al_i the best allocation of item i
end for

Instead of identifying a promising allocation parameter for the initial contract like in the SA, it is proposed to use this procedure during the negotiation when the approach is close to be trapped in a local optima. The allocation improvement procedure is executed as soon as fi^{al} negotiation rounds are executed and is repeated as soon as a new mutually accepted production plan is identified and at least 1000 negotiation rounds are performed. As soon as the procedure is activated, the allocation parameter is rebuilt from the scratch by seeking for each item $i \in I$ the best allocation parameter al_i which is defined by Algorithm 3. For the initial allocation one agent a_1 gets zero percent ($al'_i = 0$) of the production volume of item i and his rivaling agent a_2 receives hundred percent. In the first round al'_i is stored as the best allocation parameter al_i and the set of rivaling agents A_i^{al} is identified. Corresponding to the current allocation parameter al'_i for item i the demand, the lot-size, and the inventory of the rivaling agents have to be updated. Each agent $a \in A_i^{al}$ evaluates the updated contract with al'_i for a_1 and $1 - al'_i$ for a_2, respectively. Every time when the allocation parameter al'_i leads to less costs than the best allocation parameter al_i the parameter has to be updated. As long as the allocation parameter al'_i does not include the whole production volume, the process will be repeated by increasing the allocation parameter al'_i of agent a_1 by 0.5%. If the stop criterion is reached, the procedure is repeated for the remaining items in I.

To be suitable for compulsory items it is proposed to modify the shared production procedures defined by Algorithm 2 and Algorithm 3. First, it is necessary that the initial allocation parameter considers compulsory items besides concurrent items. If a compulsory item is selected, the responsible agent will receive the whole production share of the item. Secondly, Algorithm 2 has to be modified in order that compulsory items are skipped for the updating phase. At last, Algorithm 3 is modified in order that the responsible agent of an item receives the whole production share of an item. Based on these extensions the solution approach can be used for solving the DULR-PL.

Algorithm 3 Identifying a new allocation parameter

Data: problem data, c
for $i \in I$ **do**
  mediator: identify rivaling agents A_i^{al}
  mediator: compute allocation by $al'_i \leftarrow 0$
  mediator: update best allocation $al_i \leftarrow al'_i$
  while $al'_i \leq 200$ **do**
    each $a \in A_i^{al}$: evaluate c with $f_a(c)$ by al'_i
    if $f(al'_i) < f(al_i)$ **then**
      mediator: update allocation $al_i \leftarrow al'_i$
    end if
    mediator: update allocation $al'_i \leftarrow al'_i + 0.5$
  end while
  return al_i best allocation of item i
end for

IV. COMPUTATIONAL RESULTS

The performance of the algorithm SAA and the effect of compulsory items in the DULR-PL are evaluated. Section IV-A describes the setup of the study and the generated test instances. In Section IV-B our solution approach SAA is compared to the SA from Eslikizi et al. [13] by means of a benchmark study. In Section IV-C the impact of compulsory items is studied by two additional experiments.

A. Setup of computational studies

In the literature there do not exist test instances for the DULR-PL. That is the reason why it is proposed to modify the DULR instances presented by Eslikizi et al. [13] such that some items can only be produced by one agent. We use the DULR instances of Eslikizi et al. [13] instead of Buer et al. [12] because the former approach ensures a shared production by using reduced setup and integrating unit costs. The DULR instances trace back to instances for a distributed lot-sizing problem without rivaling agents [14] which on their parts are based on instances of the multi-level uncapacitated lot-sizing problem [28]. The DULR instance set includes four groups of instances denoted as $s3$, $s5$, $m3$, and $m5$ with a total of 178 instances. In our research, we focus on the instance groups $m3$ and $m5$ where either 40 or 50 items have to be produced by three agents ($m3$) or five agents ($m5$) over several production periods. The remaining instance groups are excluded from our investigation because their number of items is five only which this is not suitable for determining the impact of compulsory items.

In our computational experiments, instances with different ratios of compulsiveness are generated. Ratios of 10%, 20%, 30%, and 40% are considered, 10% means for example, that 10% of all items are compulsory items and 90% are concurrent items. Thereby, it is necessary to determine which item is selected as a compulsory item and which rivaling agent produces all units of this compulsory item. Both selections are executed randomly. Fifteen samples are generated for each ratio and instance. In total 60 samples are generated per instance, which are solved once.

In a preliminary study, appropriate parameter values for the SAA are identified. The study uses ten random instances from the instance groups $m3$ and $m5$. The focus of the study is on the percentage of changeable items si^{al}, the percentage of changeable order quantity oq^{al}, the number of negotiation rounds r_{max}^n, the end temperature T^E, and the first activation of the allocation improvement phase fi^{al}. Table I shows the identified values. The SAA is implemented in JAVA (JDK 1.7) and the computational experiments are executed on a Windows 7 personal computer with Intel Core i7-2600 processor (3.4 GHz and 16 GB main memory).

TABLE I. PARAMETER OF SAA PER INSTANCE GROUP

Group	si^{al}	oq^{al}	r_{max}^n	T_e	fi^{al}
m3	2.5%	0.1%	400000	0.01	160000
m5	2.5%	0.1%	400000	10	120000

B. The SAA in comparison with the SA

In this section, the SAA and the SA from Eslikizi et al. [13] are compared regarding the quality of the solutions and the variation range of the solutions. Each instance is solved three times per solution approach. The best solutions from both approaches are presented in Table II. Thereby, the best solution of the SA is determined out of twelve solutions because the SA uses four different solution strategies and each solution strategy is applied three times per instance. Table II shows that SAA

TABLE II. GLOBAL COSTS ACHIEVED BY SA AND SAA

No.	$\lvert A\rvert = 3$ agents		$\lvert A\rvert = 5$ agents	
	SA	SAA	SA	SAA
m01	759985.02	685141.81	778936.79	712127.65
m02	687545.60	614205.20	727333.09	641543.70
m03	1086089.11	1010118.84	1106915.64	1007190.56
m04	860086.20	805668.98	866018.42	800859.28
m05	1580327.01	1445365.10	1560647.75	1415650.00
m06	119290.82	105682.38	132523.86	111466.31
m07	622895.81	573645.31	630651.38	592323.62
m08	341277.72	322437.10	354097.10	328454.34
m09	701352.42	669115.25	707833.82	668569.27
m10	695321.03	620964.45	692166.15	624480.07
m11	2878993.28	2751008.15	2904481.25	2714465.48
m12	2592415.63	2314365.52	2617284.94	2349151.61
m13	915806.01	860621.23	981255.49	878482.00
m14	1264060.93	1077118.95	1255948.30	1108384.33
m15	2144132.99	1971335.68	2100621.57	1952109.18
m16	1210373.02	1075056.03	1276869.30	1107464.10
m17	2885857.03	2709147.90	2937496.08	2737008.52
m18	2213306.91	1946159.95	2284017.16	1953077.38
m19	778879.63	697896.15	819053.25	736060.31
m20	659816.25	571054.81	682764.42	595560.01
m21	1571572.91	1445561.03	1585372.43	1457849.85
m22	733602.68	675464.79	721462.56	659424.94
m23	2029045.86	1878867.59	2027576.94	1881469.41
m24	1342887.04	1245906.77	1335475.59	1234960.46
m25	247199.80	233676.94	258842.50	234070.02
m26	246334.21	227652.17	253456.11	226948.91
m27	1390227.37	1291415.78	1415592.65	1302254.72
m28	683908.57	641018.66	687528.08	642794.74
m29	2348688.68	2221812.47	2337850.64	2216032.48
m30	1265675.40	1200274.46	1223338.26	1174446.83
m31	5277262.81	5040619.25	5478064.86	5121352.26
m32	314004.11	289300.36	322084.86	298130.71
m33	1734672.10	1611625.96	1796772.88	1620648.81
m34	1368306.39	1228153.37	1319663.31	1208041.70
m35	3758096.16	3460015.42	3820414.65	3433879.13
m36	2181687.99	2042050.46	2223391.42	2056270.27
m37	6517142.22	6234355.45	6618031.87	6211692.30
m38	3353490.36	3176007.16	3408114.91	3182427.85
m39	1515636.55	1380737.27	1541220.44	1392516.36
m40	1166154.56	1077637.27	1202244.64	1093404.97
mean	**1601085.20**	**1485706.54**	**1624835.38**	**1492076.11**

outperforms SA on 80 out of 80 instances. The achieved cost reduction is about 7.94% per instance of the instance group $m3$ and 8.77% per instance of the instance group $m5$. Furthermore, the fluctuation rate of the solutions can be reduced from 5.6% to 1.2% per instance for instance group $m5$ and from 3.2% to 0.9% per instance for instance group $m3$. A fluctuation range of 1.2% means that the worst solution has 1.2% higher costs than the best solution of a particular instance. Obviously, the SAA is the favorable approach that even outperforms the SA on all instances in case the approach is only executed once. Corresponding to these figures it is identified that the new allocation procedure is suitable for solving the DULR. Especially the reduced fluctuation of the solution quality is important for the investigation of the impact of compulsory items because a high fluctuation rate might falsifies the result of our experiments. A disadvantage of the SAA is the forced acceptance of the allocation parameter after the allocation improvement procedure where a globally improving solution is preferred. However, the allocation improvement procedure is activated in less than 0.01% of all negotiation rounds and the procedure generates a cost reduction of about 4.5% per instance (examined instances are $m20 - m29$).

C. The impact of compulsory items

In the following experiments, the impact of compulsory items is examined by considering different ratios of compulsory items. As in the previous section, we solve each instance of the instance groups $m3$ and $m5$, but this time for the DULR-PL. Thereby, we consider fifteen samples per ratio where each instance is solved once due to the computational effort. As mentioned, we investigate ratios between 10% to 40%. In Table III the percentage of cost increases are shown per compulsory item. The figures of the instance $m01$ can be interpreted as follows: a coalition with three agents has to compensate an average cost increase of about 0.57% of the original production costs per compulsory item. Table III shows that the consideration of compulsory items always leads to higher production costs than without them. Furthermore, it can be derived that the increase of costs is almost independent from the size of the coalition. On average, the coalition has additional costs of 0.46% ($m3$) or 0.47% ($m5$) per compulsory item. By considering the results for the different ratios, it is also observed that the increase of costs decreases by higher ratios of compulsory items. Usually, it is expected that higher ratios lead to higher additional costs like in the approach presented by Ziebuhr and Kopfer [24]. However, the DULR-PL has the characteristics of an uncapacitated production volume and different cost levels corresponding to the product structure. It is assumed that if an item of a higher level of the production structure is selected as a compulsory item it will lead to higher additional costs compared to an item on a lower level. Thereby, it is obvious that we have higher costs for low ratios because the instances are more sensitive in these scenarios.

A second study is applied with the goal to confirm our assumption concerning the dependance of the additional costs and the product structure. In contrast to the first study, we do not use a random selection of compulsory items. It is proposed that each item of the same level of the product structure is selected as a compulsory item. Furthermore, we do not consider the instances $m21 - m40$ in our investigation because their product structures have several predecessors

TABLE III. INCREASE OF COSTS BY CONSIDERING COMPULSORY ITEMS (IN %)

No.	$\|A\| = 3$ agents					$\|A\| = 5$ agents				
	10%	20%	30%	40%	mean	10%	20%	30%	40%	mean
m01	0.59	0.56	0.58	0.56	**0.57**	0.51	0.48	0.48	0.50	**0.49**
m02	0.35	0.39	0.45	0.44	**0.41**	0.62	0.37	0.45	0.43	**0.47**
m03	0.32	0.43	0.53	0.41	**0.42**	0.79	0.61	0.56	0.52	**0.62**
m04	0.44	0.43	0.36	0.45	**0.42**	0.42	0.36	0.35	0.39	**0.38**
m05	0.97	0.70	0.71	0.72	**0.78**	1.29	0.94	0.77	0.79	**0.95**
m06	2.11	1.35	1.28	0.81	**1.39**	0.60	0.60	0.36	0.60	**0.54**
m07	0.43	0.47	0.36	0.42	**0.42**	0.34	0.32	0.32	0.44	**0.36**
m08	0.37	0.29	0.38	0.28	**0.33**	0.24	0.19	0.23	0.29	**0.24**
m09	0.30	0.39	0.31	0.31	**0.33**	0.44	0.42	0.39	0.35	**0.40**
m10	0.52	0.52	0.49	0.57	**0.53**	0.54	0.47	0.47	0.48	**0.49**
m11	0.56	0.54	0.48	0.52	**0.53**	0.62	0.60	0.61	0.48	**0.58**
m12	0.58	0.63	0.55	0.56	**0.58**	0.62	0.53	0.54	0.49	**0.54**
m13	0.48	0.32	0.38	0.29	**0.37**	0.40	0.40	0.30	0.34	**0.36**
m14	0.71	0.50	0.53	0.50	**0.56**	0.73	0.58	0.51	0.39	**0.55**
m15	0.49	0.56	0.43	0.50	**0.49**	0.45	0.42	0.42	0.51	**0.45**
m16	0.57	0.58	0.53	0.40	**0.52**	0.64	0.55	0.43	0.55	**0.54**
m17	0.48	0.46	0.52	0.46	**0.48**	0.49	0.56	0.55	0.54	**0.53**
m18	0.54	0.52	0.53	0.55	**0.54**	0.76	0.61	0.59	0.56	**0.63**
m19	0.46	0.45	0.43	0.43	**0.44**	0.43	0.36	0.30	0.28	**0.34**
m20	0.46	0.34	0.45	0.35	**0.40**	0.44	0.37	0.39	0.25	**0.36**
m21	0.43	0.42	0.48	0.41	**0.44**	0.47	0.46	0.43	0.44	**0.45**
m22	0.49	0.53	0.50	0.37	**0.47**	0.57	0.51	0.44	0.45	**0.49**
m23	0.36	0.45	0.40	0.46	**0.42**	0.46	0.54	0.49	0.43	**0.48**
m24	0.52	0.54	0.53	0.49	**0.52**	0.72	0.46	0.43	0.46	**0.52**
m25	0.41	0.07	0.10	0.22	**0.20**	0.66	0.50	0.63	0.52	**0.58**
m26	0.33	0.28	0.24	0.26	**0.28**	0.76	0.44	0.38	0.28	**0.46**
m27	0.45	0.44	0.46	0.46	**0.45**	0.48	0.43	0.39	0.39	**0.42**
m28	0.30	0.32	0.33	0.33	**0.32**	0.26	0.30	0.35	0.31	**0.31**
m29	0.47	0.56	0.38	0.43	**0.46**	0.39	0.50	0.52	0.38	**0.45**
m30	0.44	0.44	0.36	0.39	**0.41**	0.40	0.29	0.39	0.38	**0.37**
m31	0.36	0.41	0.43	0.44	**0.41**	0.45	0.43	0.36	0.40	**0.41**
m32	0.68	0.41	0.23	0.23	**0.39**	0.05	0.08	0.06	0.08	**0.07**
m33	0.32	0.36	0.41	0.34	**0.36**	0.39	0.48	0.55	0.49	**0.48**
m34	0.50	0.49	0.46	0.45	**0.47**	0.51	0.47	0.47	0.44	**0.47**
m35	0.55	0.51	0.48	0.47	**0.51**	0.57	0.58	0.43	0.49	**0.51**
m36	0.54	0.51	0.47	0.44	**0.49**	0.53	0.51	0.53	0.43	**0.50**
m37	0.34	0.45	0.43	0.43	**0.41**	0.49	0.50	0.51	0.49	**0.50**
m38	0.36	0.39	0.47	0.39	**0.40**	0.52	0.44	0.41	0.41	**0.45**
m39	0.38	0.40	0.38	0.40	**0.39**	0.58	0.43	0.44	0.45	**0.47**
m40	0.28	0.28	0.40	0.29	**0.31**	0.61	0.46	0.32	0.36	**0.44**
mean	**0.51**	**0.47**	**0.45**	**0.43**	**0.46**	**0.53**	**0.46**	**0.44**	**0.43**	**0.47**

which belong to different agents on a different production level. That is why we apply our study on the remaining instances which have two different production structures. One product structure is denoted as $t1$ ($m01$, $m03$, $m05$, $m07$, $m09$, $m11$, $m13$, $m15$, $m17$, $m19$) which has got a five level product structure. The other one is denoted as $t2$ ($m02$, $m04$, $m06$, $m08$, $m10$, $m12$, $m14$, $m16$, $m18$, $m20$) and has got an eight level product structure. In Table IV the results are presented, which can be interpreted as in the previous study. Table IV shows that the increase of costs per compulsory item

TABLE IV. IMPACT OF PRODUCT STRUCTURE (IN %)

Type	$\|A\|$	Level of product structure								
		1	2	3	4	5	6	7	8	9
t1	3	7.86	1.52	0.49	0.23	0.16	-	-	-	-
t1	5	6.04	1.33	0.57	0.25	0.21	-	-	-	-
t2	3	7.50	1.38	0.83	0.87	0.50	0.45	0.39	0.32	0.31
t2	5	4.97	1.21	0.61	0.52	0.43	0.39	0.32	0.30	0.18

decreases from the first production level to the last one in both scenarios ($t1$, $t2$). It can be concluded that the product level of an item has a significant impact on the additional costs. For example, when an item of the first level concerning the product structure is selected as a compulsory item then it is recommended to be aware of higher costs compared to the case when an item of the last level concerning the product structure is selected. By considering the figures of Table IV, it can be observed that especially the first level of the product structure has a significant impact on the solution.

V. Conclusion

This paper studies the DULR-PL. The DULR-PL is an inter-organizational lot-sizing problem with rivaling agents where some items have to be produced by an appointed member of the coalition. These items are denoted as compulsory items. To solve this problem a simulated annealing negotiation mechanism denoted as SAA is applied. The SAA extends an existing solution approach by a new procedure for identifying a suitable shared production among the members of a coalition and a procedure for handling compulsory items.

In a benchmark study, we compare the SAA with the only existing approach for the DULR. Thereby, we identify that SAA outperforms the other approach on 80 out of 80 instances where on average over all instances a cost reduction of about 8% per instance is achieved by simultaneously reducing the fluctuation range of the approach. Based on these figures, the SAA is used for solving the DULR-PL where several findings could be derived. The experiments show that compulsory items always lead to higher production costs and that items on a higher level of the product structure have a higher impact on the fulfillment costs. In our study, we identify that each compulsory item causes additional costs of about 0.47% of the total costs.

For future research it appears promising to apply our findings for developing an improved solution approach where the heuristic focuses on the investigation of items on a higher level of the product structure because their impact on the solution quality is more significant. Further on, it might be interesting to develop a combined solution approach for solving simultaneously the lot-sizing problem together with the corresponding transportation planning problem by considering compulsory items.

Acknowledgment

The cooperative junior research group on *Computational Logistics* is funded by the University of Bremen in line with the Excellence Initiative of German federal and state governments.

References

[1] L. Yelle, "Materials requirements lot sizing: A multilevel approach," *International Journal of Production Research*, vol. 17, no. 3, pp. 223–232, 1979.

[2] E. Steinberg and H. Napier, "Optimal multi-level lot sizing for requirements planning systems," *Management Science*, vol. 26, no. 12, pp. 1258–1271, 1980.

[3] L. Buschkühl, F. Sahling, S. Helber, and H. Tempelmeier, "Dynamic capacitated lot-sizing problems: a classification and review of solution approaches," *OR Spectrum*, vol. 32, no. 2, pp. 231–261, 2010.

[4] A. Fink, "Supply chain coordination by means of automated negotiation between autonomous agents," in *Multiagent-based supply chain management*, B. Chaib-draa and J. & Müller, Eds. Berlin: Springer, 2006, pp. 351–372.

[5] G. Dudek and H. Stadtler, "Negotiation-based collaborative planning between supply chains partners," *European Journal of Operational Research*, vol. 163, no. 3, pp. 668–687, 2005.

[6] ——, "Negotiation-based collaborative planning in divergent two-tier supply chains," *International Journal of Production Research*, vol. 45, no. 2, pp. 465–484, 2007.

[7] A. Taghipour and J.-M. Frayret, "Mutual adjustment search with incentive for supply chain planning coordination," *International Journal of Computer Integrated Manufacturing*, vol. 25, no. 10, pp. 946–962, 2012.

[8] ——, "Dynamic mutual adjustment search for supply chain operations planning co-ordination," *International Journal of Production Research*, vol. 51, no. 9, pp. 2715–2739, 2013.

[9] A. Fink, N. Kliewer, D. Mattfeld, L. Mönch, F. Rothlauf, G. Schryen, L. Suhl, and S. Voß, "Model-based decision support in manufacturing and service networks," *Business & Information Systems Engineering*, vol. 6, no. 1, pp. 17–24, 2014.

[10] H. Stadtler, "A framework for collaborative planning and state-of-the-art," *OR Spectrum*, vol. 31, no. 1, pp. 5–30, 2009.

[11] F. Lang and A. Fink, "Learning from the metaheuristics: Protocols for automated negotiations," *Group Decision and Negotiation*, vol. 24, no. 2, pp. 299–332, 2015.

[12] T. Buer, M. Ziebuhr, and H. Kopfer, "A coordination mechanism for a collaborative lot-sizing problem with rivaling agents," in *Logistics Management*, ser. Lecture Notes in Logistics, H. Kotzab, J. Schönberger, J. Dethloff, H.-D. Haasis, and H. Kopfer, Eds., 2014, pp. 325–337.

[13] S. Eslikizi, M. Ziebuhr, H. Kopfer, and T. Buer, "Shapley-based side payments and simulated annealing for distributed lot-sizing," in *Proceedings of the 15th IFAC Symposium on Information Control Problems in Manufacturing (INCOM 2015)*, 2015, pp. 1637–1642, in press.

[14] J. Homberger, "Decentralized multi-level uncapacitated lot-sizing by automated negotiation," *4OR: A Quarterly Journal of Operations Research*, vol. 8, pp. 155–180, 2010.

[15] T. Buer, J. Homberger, and H. Gehring, "A collaborative ant colony metaheuristic for distributed multi-level uncapacitated lot-sizing," *International Journal of Production Research*, vol. 51, no. 17, pp. 5253–5270, 2013.

[16] M. Ziebuhr, T. Buer, and H. Kopfer, "Agent-negotiation of lot-sizing contracts by simulated annealing with part-way resets," in *Multiagent System Technologies - 11th German Conference, MATES 2013, Koblenz, Germany, Sept 16-20, 2013. Proceedings.*, ser. LNAI, M. Klusch, M. Thimm, and M. Paprzycki, Eds., vol. 8076. Berlin Heidelberg: Springer, 2013, pp. 166–179.

[17] O. Ö. Özener and Ö. Ergun, "Allocating costs in a collaborative transportation procurement network," *Transportation Science*, vol. 42, no. 2, pp. 146–165, 2008.

[18] T. Buer and H. Kopfer, "A Pareto-metaheuristic for a bi-objective winner determination problem in a combinatorial reverse auction," *Computers & Operations Research*, vol. 41, pp. 208–220, 2014.

[19] T. Buer, "An exact and two heuristic strategies for truthful bidding in combinatorial transport auctions," Computational Logistics, University of Bremen, Tech. Rep. arXiv:1406.1928, 2014.

[20] S. Berger and C. Bierwirth, "Solutions to the request reassignment problem in collaborative carrier networks," *Transportation Research Part E: Logistics and Transportation Review*, vol. 46, no. 5, pp. 627–638, 2010.

[21] M. Gansterer and R. Hartl, "Request evaluation strategies for carriers in auction-based collaborations," *OR Spectrum*, in press.

[22] O. Ö. Özener, Ö. Ergun, and M. Savelsbergh, "Lane-exchange mechanisms for truckload carrier collaboration," *Transportation Science*, vol. 45, no. 1, pp. 1–17, 2011.

[23] J. Schönberger, *Operational Freight Carrier Planning – Basic Concepts, Optimization Models and Advanced Memetic Algorithms*. Springer-Verlag, 2005.

[24] M. Ziebuhr and H. Kopfer, "The integrated operational transportation planning problem with compulsory requests." in *Proceedings of the International Conference on Computational Logistics 2014, LNCS 8760*, 2014.

[25] F. Reiß and T. Buer, "A coordination mechanism for capacitated lot-sizing in non-hierarchical n-tier supply chains," in *2014 IEEE Symposium on Computational Intelligence in Production and Logistics Systems (CIPLS 2014)*, 2014, pp. 9–15.

[26] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, pp. 671–680, 1983.

[27] J. Homberger, H. Gehring, and T. Buer, "Integrating side payments into collaborative planning for the distributed multi-level unconstrained lot sizing problem," in *48th Annual Hawaii International Conference on System Sciences*. IEEE Computer Society Press, 2015, pp. 1068–1077.

[28] N. Dellaert and J. Jeunet, "Solving large unconstrained multilevel lot-sizing problems using a hybrid genetic algorithm," *International Journal of Production Research*, vol. 38, no. 5, pp. 1083–1099, 2000.