Operations Research Forum (2024) 5:68
https://doi.org/10.1007/s43069-024-00350-8

RESEARCH

A Modeling and Hybridized Decomposition Approach for the Multi-level Capacitated Lot-Sizing Problem with Setup Carryover, Backlogging, and Emission Control

Nusrat T. Chowdhury[1,3] · Mohammed F. Baki[2] · Ahmed Azab[3]

Received: 6 October 2023 / Accepted: 12 July 2024

Abstract

The goal of the production planning problem is to determine the optimum quantity to produce in order to satisfy demand over a predetermined planning horizon with the least amount of money spent. Making the appropriate choices in production planning will impact a manufacturing company's performance and productivity, which is crucial to remain competitive in the market. Therefore, developing and enhancing techniques for solving production planning problems is very significant. This paper proposes a mixed-integer linear programming model for this extension of the dynamic multi-level capacitated lot-sizing under study, where setup carryover, backlogging, and emission control are considered. An item Dantzig-Wolfe decomposition-based heuristic procedure is developed, and a dynamic programming and column generation approach is used to solve the problem. We also propose a multi-step iterative capacity allocation heuristic procedure to handle any infeasibilities that arise when solving the problem. We evaluate the performance of the developed solution approach using a test data set available in the literature. Computational results show that the proposed optimization framework provides competitive solutions within a reasonable timeframe.

Keywords Inventory · Multi-level capacitated lot-sizing · Emission control · Setup carryover · Dantzig-Wolfe decomposition

✉ Nusrat T. Chowdhury
nusrat.chowdhury@bemidjistate.edu

Mohammed F. Baki
fbaki@uwindsor.ca

Ahmed Azab
azab@uwindsor.ca

[1] School of Technology, Art and Design, Bemidji State University, Bemidji, MN 56601, USA

[2] Odette School of Business, University of Windsor, Windsor, ON N9B 3P4, Canada

[3] Production & Operations Management Research Laboratory, Faculty of Engineering, University of Windsor, Windsor, ON N9B 3P4, Canada

Published online: 06 August 2024

1 Introduction

There are numerous models available for inventory control and production planning. Lot-sizing problems involve determining the optimum production plan or inventory replenishment policy while minimizing the total cost of the system. The capacitated dynamic lot-sizing problem (CLSP) deals with the problem of determining time-phased production quantities that meet given external demands and the capacity limits of the production system. The multi-level extension of the CLSP, known as multi-level capacitated lot-sizing problem (MLCLSP), deals with the production of multiple items when an interdependence among them at the different production levels is imposed by the product structure. The classical MLCLSP is introduced by Billington et al. [1], which is considered the theoretical basis for material requirements planning [2].

Setup operations are significant in some manufacturing industries and may strongly influence lot-sizing decisions. Setup operations prepare the processing units to manufacture production lots, consume production capacity (setup time), and incur setup costs. The classical CLSP assumes that the setup of the resources for each item produced in each period is necessary. However, some researchers assume that the setup state of a machine can be fully maintained over consecutive periods. In the literature (e.g., [3]), this is denoted as setup carryover. Simply put, setup carryover permits a setup state to be conserved between two consecutive periods. Haase [4] points out that solutions change considerably when setup carry-over is considered.

Many industries are responsible for the emission of greenhouse gases (GHG), such as carbon dioxide (CO_2), nitrous oxide (N_2O), and methane (CH_4), often throughout the entire production process (Fig. 1). Carbon is emitted directly from energy generation and the consumption of energy in setup, production, and inventory-holding activities [5]. Recently, there has been growing concern about the effect of these gases on climate change. Many countries implement various carbon regulatory measures and legislation such as carbon cap, carbon cap-and-trade,

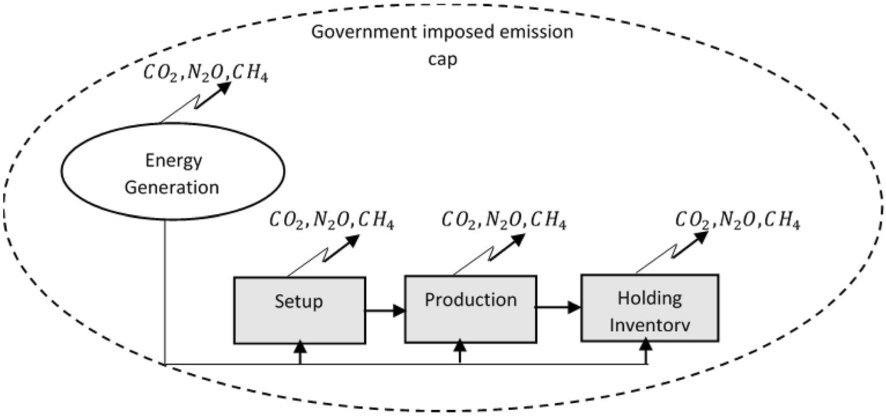


Fig. 1 Greenhouse gas emission from the different activities of the production process

carbon tax, and carbon cap-and-offset. These laws and various regulatory mechanisms in regard to emissions oblige the manufacturing industry to implement alternative, more environment-friendly production systems. They need to invest in more energy-efficient technology as well as renewable energy sources, which are costly capital investments and practices. This has motivated many researchers to consider the environmental impact of emissions by incorporating emission-related measures in their models for optimizing production lot size.

In this paper, we present a Mixed Integer Linear Programming (MILP) model for the extension of the classical MLCLSP by allowing setup carryover, backlogging, and emission control. We refer to the problem as MLCLSP with SCBE (multi-level capacitated lot-sizing problem with setup carryover, backlogging, and emission control). We apply an item Dantzig-Wolfe (DW) decomposition approach to solve the proposed MILP formulation with an embedded column generation (CG) procedure. We propose a dynamic programming approach to solve each of the subproblems and develop a multistep iterative capacity allocation (CA) heuristic to generate feasible solutions. An integer linear programming (ILP) model is proposed to determine the setup carryover plan for a given production schedule. This approach is hybridized with an LP-based improvement procedure to refine the solution, thereby improving the solution quality given by the DW decomposition method.

The remainder of this paper is organized as follows: In Sect. 2, the related literature is discussed. In Sect. 3, we present the problem statement along with the formulated mathematical model. The proposed DW decomposition heuristic method is described in Sect. 4. Numerical results are discussed in Sect. 5. Finally, in Sect. 6, we conclude and provide a few recommendations for future work.

2 Literature Review

Production planning and enterprise resource planning (ERP) systems are built around lot-sizing decisions. Therefore, these decisions have been the focus of in-depth inquiry by academics and professionals for decades. Since the seminal paper of Wagner and Whitin [6] addressing the simplest version of the problem—the uncapacitated single-item lot-sizing problem—various types of lot-sizing problems have been investigated. Only some special cases of these problems can be solved in polynomial time (e.g., Federgruen and Tzur [7]). Florian et al. [8] have proven that the capacitated version of the single-item lot-sizing problem is NP-hard. Later, Bitran and Yanasse [9] show that even special cases, which are solvable in polynomial time, become NP-hard when introducing a second item. Therefore, because of the intractable nature of the lot-sizing problems, different solution algorithms and strategies have been employed. This includes Lagrangean relaxation (LR), Dantzig-Wolfe (DW) decomposition, and fix and optimize (FO) approach.

Tempelmeier and Derstroff [10] apply Lagrangean relaxation (LR) to decompose the MLCLSP into several single-item uncapacitated lot-sizing problems (SIULSP) to obtain the lower bounds and propose a heuristic finite scheduling approach to find the upper bounds. Sox and Gao [11] incorporate setup carryover into a multi-item CLSP and provide a Lagrangean decomposition heuristic that quickly generates near-optimal solutions and proposes a dynamic programming approach to solve N independent single-item subproblems. Later, Briskorn [3] revisits the problem addressed by Sox and Gao [11]. He identifies a flaw in the dynamic programming approach by Sox and Gao [11] and provides the necessary correction to solve the subproblems optimally.

Tempelmeier and Buschkühl [12] consider setup carryover in an MLCLSP and develop a Lagrangean decomposition heuristic. Sahling et al. [13] extend the work of Tempelmeier and Buschkühl [12] by incorporating multi-period setup carryovers and propose an iterative FO approach to solve a series of MILPs. Wu et al. [14] propose an MIP formulation for modeling the MLCLSP with both backlogging and setup carryover. They present a progressive time-oriented decomposition framework that uses relax and fix (RF) heuristic. Setup carryover is considered by Carvalho and Nascimento [15] for a multi-plant CLSP and solved by kernel search (KS) metaheuristic method. Ghirardi and Amerio [16] study a CLSP with back-ordering and setup carryover and develop a feasibility pump algorithm to solve the problem. One of the recent studies that includes setup carryover is conducted by Fiorotto et al. [17] and a relax-and-fix and fix-and-optimize (RF-FO) heuristic is applied to solve the problem.

Lot-sizing with emission constraints was introduced by Benjaafar et al. [18]. Retel Helmrich et al. [19] show that lot-sizing with emission constraints is NP-hard. The most recent research works on production lot-sizing and inventory planning problems focus on the reduction of carbon emission and conservation of resources Mashud et al. [20], Mashud et al. [21], Sepehri [22]. Konur and Schaefer [23] analyze an integrated inventory control and transportation planning problem under carbon cap, cap and trade, cap and offset, and taxing policies. Ghosh [24] identified the two most adopted carbon policies are (i) carbon tax/cost policy and (ii) carbon cap-and-trade policy and solved the MINLP problem for both policies. Ahmadini et al. [25] solve a multi-item multi-objective inventory model with back-ordered quantity incorporating green investment to save the environment.

Dantzig-Wolfe (DW) decomposition is applied for CLSP for the first time by Manne [26], in which lot-sizing problems are decomposed by item. Degraeve and Jans [27] later claim that the decomposition method proposed by Manne [26] has an important structural deficiency. They explain that imposing integrality constraints on the variables in the master problem does not necessarily give an optimal integer solution, as only the production plans, which satisfy the zero inventory property (if production takes place in a period t, the beginning inventory for that period must be zero), can be selected. Degraeve and Jans [27] therefore proposed a new DW reformulation and a Branch-and-Price (B&P) algorithm. Pimentel et al. [28] present a comparison between the item DW decomposition and the period DW decomposition of a multi-item CLSP and apply the B&P algorithm to solve the decomposition models. Caserta & Voß [29] propose the DW decomposition approach in a meta-heuristic framework for the multi-item, multi-period CLSP with setup times. Fiorotto et al. [30] develop an efficient hybrid algorithm by combining LR and DW decomposition for the CLSP with multiple items, setup time, and unrelated parallel machines. Ioannis et al. [31] solve CLSP by implementing DW decomposition in a novel way that regulates the size of the master problem and the subproblem independently. Wu et al. [32] proposed an item DW decomposition for the facility location and production planning problem. They demonstrate that the pricing subproblems of the item decomposition are related to uncapacitated lot-sizing problems with the Wagner-Whitin (WW) property. This property is employed to enhance the performance of column generation for the item decomposition.

CLSP is a mature field of research, and therefore, we can benchmark the performance of our approach against other alternative techniques. Most of the literature considers single-level CLSP with multi-item, multi-period, and setup time. However, there has been insufficient evidence of the implementation of DW decomposition for the MLCLSP. Table 1 chronologically presents some of the studies conducted based on the approach proposed, type, and properties of the lot-sizing problem.

3 Problem Formulation and Decomposition Method for MLCLSP with Setup Carryover, Backlogging, and Emission Control (MLCLSP with SCBE)

3.1 MLCLSP with Setup Carryover, Backlogging, and Emission control (MLCLSP with SCBE)

The model MLCLSP with setup carryover, backlogging and emission control (MLCLSP with SCBE) is based on the following assumptions:

Table 1 Proposed heuristic approaches for solving the capacitated lot-sizing problems

Reference	Type of lot-sizing problem	Properties of the Problem	Solution approach
Manne [26]	SL	OT	DW
Billington et al. [1]	ML	ST	LR and B&B
Tempelmeier and Derstroff [10]	ML	ST	LDH
Sox and Gao [11]	SL	ST, SC	LDH
Degraeve and Jans [27]	SL	MI, ST	DW, CG and B&P
Tempelmeier and Buschkühl [12]	ML	MI, ST, SC	LDH
Wu et al. [14]	ML	ST, SC, BL	RF
Gören and Tunalı [33]	SL	SC, ST, BL	FO heuristic
Fiorotto et al. [17]	SL	MI, ST, SC	RF and FO
Ahmadini et al. [25]	SL	MI, BL, EC	FGP
Mashud et al. [20]	SL	SC, EC, BL	Solved optimally for small problems
*Chowdhury, Baki, Azab	ML	MI, ST, SC, EC	DW, CA, and CG

Problem solved: *SL* single-level, *ML* multi-level, *MI* multi-item, *ST* setup time, *SC* setup carryover, *SDST* sequence dependent setup time, *SM* single machine, *OT* over time, *BL* backlogging, *EC* emission control

Solution approach: *B&B* branch and bound, *LR* Lagrangean relaxation, *LDH* Lagrangean decomposition heuristic, *DW* Dantzig-Wolfe, *CG* column generation, *B&P* branch and price, *GA* genetic algorithm, *FO* fix and optimize, *RF* relax and fix, *FGP* fuzzy goal programming, *CA* capacity allocation heuristic

*This paper

1. The planning horizon is divided into T periods (usually shifts or days), and there are m resources with period-specific capacities.
2. n items (including end items and subassemblies) are arranged in an assembly product structure with a unique assignment of each item to a particular machine.
3. External dynamic demand is assumed to occur only at stages that have no successors and are known in advance. Full demand occurs at the beginning of each period.
4. Production cost is time varying, and setup cost is fixed over time.
5. A setup incurred may cause setup cost as well as a setup time. Setup costs, as well as setup times, are sequence independent.
6. At the beginning of the planning horizon, machines are not setup for any item.
7. At most one setup state can be carried over on each resource from one period to the next, such that no setup activity is necessary in the second period.
8. The conservation of one setup state for the same product over two consecutive periods is possible if only one item is produced in a particular period.
9. A setup state is not lost if there is no production on the machine within a planning period.
10. Backlogging is allowed for the items with external demand.

Moreover, we account for carbon emissions generated by different activities in the plant and warehouse. This includes production (e.g., greenhouse gas emissions due to burning fossil fuels for energy, as well as certain chemical reactions necessary to produce goods from raw materials), holding (emissions due to energy spent on storage), and setup (emissions due to machine setup). A carbon emission cap regulatory mechanism is considered in which the total emissions due to all production-related activities over the planning horizon cannot exceed a carbon cap imposed by a regulator over the planning horizon.

The problem is to find a production schedule, machine setup and carryover plans, inventory levels, and backlogging quantities in each time period that meet the demand requirements, limited capacity resources along with the emission cap, taking into consideration the bill of material (BOM) structure while simultaneously minimizing the total cost over the planning horizon of T periods.

Indices:

T Number of periods
n Number of items
m Number of machines

The decision variables are as follows:

I_{jt} Inventory level of item $j \in [1, n]$ at the end of period $t \in [1, T]$

$$Y_{jt} = \begin{cases} 1 & \textit{if there is a setup for item } j \in [1,n] \textit{on machine } i \in [1,m] \textit{in period } t \in [1,T] \\ 0 & \textit{otherwise} \end{cases}$$

$\alpha_{jt} =$

$$\begin{cases} b_{jt} \text{ Quantity back ordered for item } j \in [1,n] \text{ in period } t \in [1,T] \\ 1 \quad \textit{if the setup state of machine } i | j \in \varphi(i) \textit{at the end of period } t \in [1,T] \textit{and at the beginning of period} (t+1) \textit{isitem} j \in [1,n] \\ 0 \qquad \textit{Otherwise} \end{cases}$$

X_{jt} Production quantity of item $j \in [1,n]$ in period $t \in [1,T]$
E_t Emission due to production, inventory, and setup in period $t \in [1,T]$

The parameters used are as follows:

g_{jk} Quantity of item $j \in [1,n]$ required to produce one unit of item $k \in [1,n]$
D_{jt} External demand of item $j \in [1,n]$ in period $t \in [1,T]$
h_j Holding cost of item $j \in [1,n]$
c_j Setup cost for item $j \in [1,n]$
s_j Setup time for item $j \in [1,n]$
M A large number
I_{j0} Initial inventory level of item $j \in [1,n]$
R_{it} Available capacity of machine $i \in [1,m]$ in period $t \in [1,T]$ (in time units).
$\Gamma(j)$ Set of immediate successors of item $j \in [1,n]$ based on BOM.
P_{jt} Production cost per unit of finished item $j \in [1,n]$ at period $t \in [1,T]$
$\varphi(i)$ Set of items that can be assigned to machine $i \in [1,m]$
ω Set of end items (items with external demand only)
$\mu(j)$ Set of immediate predecessors of item $j \in [1,n]$
$\rho(j)$ Set of machines eligible to process item $j \in [1,n]$
p_j Processing time required to produce one unit of item $j \in [1,n]$
β_j Backlogging cost for one unit of item $j \in [1,n]$ per period.
$\widehat{s}_j$ Carbon emission related to the setup of item $j \in [1,n]$
$\widehat{p}_j$ Carbon emission related to per unit production of item $j \in [1,n]$
$\widehat{h}_j$ Carbon emission related to per unit holding inventory of item $j \in [1,n]$
C_{cap} Total allowable carbon emission cap

Model MLCLSP_SCBE:

$$Min \sum\nolimits_{j=1}^{n} \sum\nolimits_{t=1}^{T} (P_{jt}X_{jt} + h_j I_{jt} + c_j Y_{jt} + \beta_j b_{jt}) \tag{1}$$

Subject to:

$$I_{j0} = b_{j0} = \alpha_{j0} = 0 \forall j \in [1,n] \tag{2}$$

$$I_{jt} = I_{j(t-1)} + X_{jt} + b_{jt} - b_{j(t-1)} - D_{jt} \forall j \in [1,n], t \in [1,T] | j \in \omega \tag{3}$$

$$I_{jt} = I_{j(t-1)} + X_{jt} - \sum\nolimits_{k \in \Gamma(j)} g_{jk} X_{kt} \cdots \forall j \in [1,n], t \in [1,T] | j \notin \omega \tag{4}$$

$$X_{jt} \leq \mathrm{M}\left(Y_{jt} + \alpha_{j(t-1)}\right)\forall i \in [1,m], j \in \varphi(i), t \in [1,T] \tag{5}$$

$$\sum\nolimits_{j\in\varphi(i)} p_j X_{jt} + \sum\nolimits_{j\in\varphi(i)} s_j Y_{jt} \leq R_{it} \forall i \in [1,m], t \in [1,T] \tag{6}$$

$$\alpha_{jt} \leq Y_{jt} + \alpha_{j(t-1)} \forall i \in [1,m], j \in \varphi(i), t \in [1,T] \tag{7}$$

$$\sum\nolimits_{j\in\varphi(i)} \alpha_{jt} \leq 1 \forall i \in [1,m], t \in [1,T] \tag{8}$$

$$E_t = \sum\nolimits_{j=1}^{n} \left(\hat{p}_j X_{jt} + \hat{h}_j I_{jt} + \hat{s}_j Y_{jt}\right) \forall t \in [1,T] \tag{9}$$

$$\sum_{t=1}^{T} E_t \leq C_{cap} \tag{10}$$

$$I_{jt} \geq 0, X_{jt} \geq 0, b_{jt} \geq 0, \forall j \in [1,n], t \in [1,T] \tag{11}$$

$$b_{jT} = 0 \forall j \in [1,n] \tag{12}$$

$$Y_{jt,} \alpha_{jt} \in [0,1] \forall i \in [1,m], j \in \varphi(i), t \in [1,T] \tag{13}$$

The complete MILP model is presented as the minimization of the objective function (1), subject to constraints (2) through (13). The objective function (1) minimizes the total production, holding, setup, and backlogging cost over the planning horizon of T periods. Constraint (2) sets the initial inventory, initial backlog, and initial setup condition of the machines for each item. Constraints (3) and (4) represent the inventory balance for those products that are needed to satisfy external and internal demands, respectively. Constraint (5) ensures that the production of item j takes place in period t only if there is a setup of the machine i for item j during that period (Y_{jt}= 1), or if the resource is already in the correct setup state at the beginning of that period ($\alpha_{j(t-1)} = 1$). Constraint (6) indicates that production cannot exceed the available capacity. Constraint (7) makes sure that a setup can be carried over from period t to period $(t + 1)$ only if either item j is setup in period t or the setup state already has been carried over from period $(t - 1)$ to period t. Constraint (8) states that a machine can carry at most one setup state into the subsequent period. Constraint (9) computes the carbon emission due to production, inventory, and setup for each period. Constraint (10) controls carbon emissions under carbon cap regulatory mechanism applied over the planning horizon, which states that the total emissions should not exceed the total available emission limit (C_{cap}) in any period. Constraint (11) declares the nonnegativity of the variables, constraint (12) states that the ending backlogging quantity is zero, and constraint (13) is the integrality ones.

3.2 Dantzig-Wolfe Decomposition for the MLCLSP with SCBE

DW decomposition is a special technique used to solve linear programming and integer programming models. DW decomposition redefines a new set of variables by replacing the original variables with a convex combination of the extreme points of a subsystem. This technique has been effectively implemented in different contexts. For more details on such techniques see, Vanderbeck [34], and Vanderbeck and Savelsbergh [35]. Degraeve and Jans [27] have presented a DW approach for the CLSP, addressing an important structural deficiency of the standard DW approach for the CLSP proposed by Manne [26]. In this section, borrowing ideas from Degraeve and Jans [27], we decompose the MLCLSP with SCBE into a master problem and a number of single-item uncapacitated subproblems with backlogging and setup carryover. In this decomposition, the solutions of the subproblems are production plans for a single item. For a given product, each production plan specifies the production periods and the production quantity along with the inventory level, setup plan, backlog quantity, and setup carryover decisions.

Let us introduce a more compact notation for the variables: $X^j = (X_{j1}, X_{j2}, \dots, X_{jT})$, $I^j = (I_{j1}, I_{j2}, \dots, I_{jT})$,$b^j = (b_{j1}, b_{j2}, \dots, b_{jT})$,$Y^j = \left(Y_{j1}, Y_{j2}, \dots, Y_{jT}\right)\alpha^j = (\alpha_{j1}, \alpha_{j2}, \dots, \alpha_{jT})$.

Let U_j be the set of all feasible production schedules. For a production scheduleu $u \in U_j$, letX^u_{jt},I^u_{jt}, Y^u_{jt}, b^u_{jt}, and α^u_{jt} be the production quantity, ending inventory level, setup plan, quantity backlogged, and setup carryover decision variable of item j in period t, respectively. The master problem is given as follows:

Master problem (MP_1):

$$Min \sum_{u \in U_j} \sum\nolimits_{j=1}^{n} \sum\nolimits_{t=1}^{T} (P_{jt} X^u_{jt} + h_j I^u_{jt} + c_j Y^u_{jt} + \beta_j b^u_{jt}) \lambda_{ju} \tag{14}$$

Subject to:

$$\sum_{u \in U_j} \sum\nolimits_{j \in \varphi(i)} (p_j X^u_{jt} + s_j Y^u_{jt}) \lambda_{ju} \le R_{it} \forall i, t \tag{15}$$

$$\sum_{u \in U_j} \sum\nolimits_{t=1}^{T} \sum\nolimits_{j=1}^{n} \left(\widehat{p}_j X^u_{jt} + \widehat{h}_j I^u_{jt} + \widehat{s}_j Y^u_{jt} \right) \lambda_{ju} \le C_{cap} \tag{16}$$

$$\sum_{u \in U_j} \sum_{j \in \varphi(i)} \alpha^u_{jt} \lambda_{ju} = 1 \forall t \ge 0, i \tag{17}$$

$$\sum_{u \in U_j} \lambda_{ju} = 1 \forall j \tag{18}$$

$$\lambda_{ju} \ge 0 \forall j, u \in U_j \tag{19}$$

Let λ_{ju} be the new decision variable representing the weight of the production plan u for item j. The objective function (14) minimizes the total cost of the production plans chosen for each item. Let w_{it},γ, y_{it} and v_j be the dual variables with respect to (15), (16), (17) and (18), respectively. During each iteration, the master problem handles the linking constraints (constraints 15, 16, and 17) that connect the subproblems and the convexity constraints (constraint 18) for each subproblem. Constraint (19) expresses the non-negativity constraints. Due to the large number of variables of the master problem, the linear programming relaxation of the master problem is solved using column generation.

The decomposed subproblems for each end item $j|j \in \omega$ are as follows.

Subproblem $\left(SP1_{End}\right)$:

$$Min\sum_{i=1}^{m}\sum_{t=1}^{T}\left[(P_{jt}-w_{it}p_j-\gamma\widehat{p}_j)X_{jt}+(c_j-w_{it}s_j-\gamma\widehat{s}_j)Y_{jt}-\alpha_{jt}y_{it}\right]+\sum\nolimits_{t=1}^{T}\left[(h_j-\gamma\widehat{h}_j)I_{jt}+\beta_jb_{jt}\right]-v_j\forall j \tag{20}$$

Subject to:

$$I_{j0}=b_{j0}=\alpha_{j0}=0 \tag{21}$$

$$I_{jt}=I_{j(t-1)}+X_{jt}+b_{jt}-b_{j(t-1)}-D_{jt}\forall t \tag{22}$$

$$X_{jt}\leq \mathrm{M}\left(Y_{jt}+\alpha_{j(t-1)}\right)\forall i\in\rho(j),t \tag{23}$$

$$Y_{jt}+\alpha_{j(t-1)}\leq 1\forall i\in\rho(j),t \tag{24}$$

$$I_{jt},X_{jt},b_{jt}\geq 0\forall i\in\rho(j),t\geq 1 \tag{25}$$

$$Y_{jt},\alpha_{jt}\in[0{,}1]\forall i\in\rho(j),t\geq 1 \tag{26}$$

The subproblem (20) through (26) is a single-item uncapacitated lot-sizing problem and is solved by dynamic programming recursion (DPR) to determine the production schedule of the end items with strictly external demand (i.e., no successors). In this decomposition, the solutions to the subproblems are the production plans. For a given product, each production plan indicates the production periods and the production quantity along with the inventory level, backlogging, and setup carryover decisions. After all end items are scheduled, the next item, $k|k\in\mu(j)$, is scheduled. The decomposed subproblems for all $k|k\in\mu(j)$ are as follows:

Subproblem $\left(SP1_{\mathrm{Component}}\right)$:

$$Min\sum_{i=1}^{m}\sum_{t=1}^{T}\left[(P_{kt}-w_{it}p_k-\gamma\widehat{p}_k)X_{kt}+(c_k-w_{it}s_k-\gamma\widehat{s}_k)Y_{kt}-\alpha_{kt}y_{it}\right]+\sum\nolimits_{t=1}^{T}\left[(h_k-\gamma\widehat{h}_k)I_{kt}\right]-v_k\cdots\forall k\in\mu(j) \tag{27}$$

Subject to,

$$I_{kt} = I_{k(t-1)} + X_{kt} - \sum\nolimits_{k' \in \Gamma(k)} a_{kk'} X_{k't} \forall t \tag{28}$$

and (21), (23)–(26) for $j = k$.

The internal demand of any item (successor requirement) is placed on the right-hand side of the constraint (28), because $\sum_{k' \in \Gamma(k)} a_{kk'} X_{k't} \forall t$ is the dependent demand for item k due to the production of its successors j that have already been scheduled. Treating the internal demands as constants, $SP1_{\text{Component}}$ is equivalent to $SP1_{\text{End}}$, and hence, item k's production schedule can now be determined. Thus, a production schedule for all the items can be found for a given set of dual variables if the procedure is followed item-by-item in succession to make sure that all requirements resulting from the production of the successor items are calculated before scheduling an immediate predecessor. Equation (28) uses a sequential bill of material approach to pass successors' production requirements between levels. Although it does not guarantee optimality, this procedure will ensure a feasible solution to the full set of inventory constraints.

The CG process begins by creating an initial set of feasible production plans for the master problem by fixing all the dual variables at a value of zero. The initial set of production plans is obtained from the uncapacitated single-item subproblems. However, it is possible that the production requirements of the items in a period may be greater than the available capacity. According to the theory of DW decomposition approach, updating the dual variables w_{it},γ, y_{it} and v_j should take these infeasibilities into account; otherwise, the master problem (MP_1) becomes infeasible because the constraints (15) and (19) may not be satisfied. If demand for one item is greater than the capacity in a period, a split lot is required. Ramsay [36] shows that a feasible solution is often not attainable because an uncapacitated lot-sizing problem does not split the lot-sizes between periods. To avoid infeasibility, we propose a CA heuristic (Sect. 4.2) to obtain a feasible setup plan. Since each of the SIULSP is solved individually, the setup carryover decisions per resource are not coordinated. At most, one item can be carried over from one period to the next and if an item is carried over from period t to $(t + 1)$, this item must have been produced first in period $(t + 1)$. Therefore, it is necessary to generate a feasible solution by incorporating the setup carryover constraints in the solution produced of the single-item subproblems. In this paper, an integer linear programming (ILP) model (Sect. 3.4) is developed to determine the setup carryover decision variables optimally, with the objective of maximizing the savings vis-a-vis setup costs. An LP-based improvement procedure is applied to obtain an optimum production schedule for a given set of setup plans and setup carryover decisions. If there is a production schedule u that makes the reduced cost negative, it is added to U_j. Then the master problem is solved to provide new dual variables. This procedure is repeated until no new column with a negative reduced cost is found.

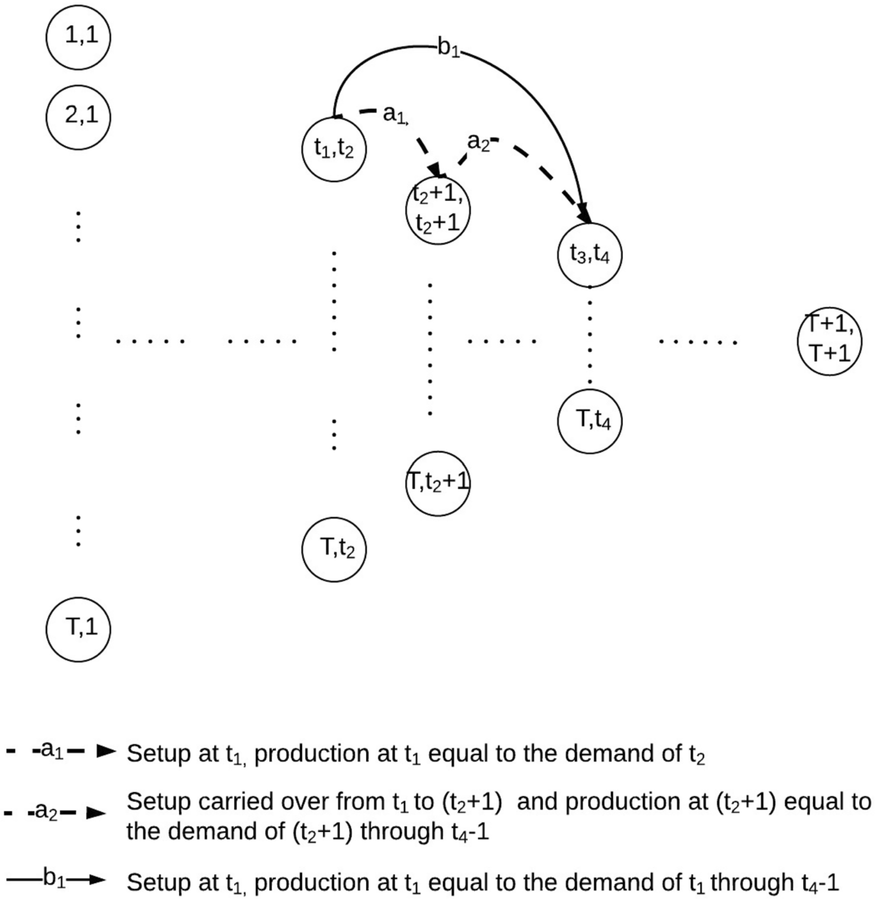

1,1
2,1
T,1
t1,t2
T,t2
t2+1, t2+1
T,t2+1
t3,t4
T,t4
T+1, T+1
b1
a1
a2
- -a1— ► Setup at t1, production at t1 equal to the demand of t2
- -a2— ► Setup carried over from t1 to (t2+1) and production at (t2+1) equal to the demand of (t2+1) through t4-1
—b1→ Setup at t1, production at t1 equal to the demand of t1 through t4-1

Fig. 2 Shortest path network for the subproblem

3.3 Dynamic Programming Recursion (DPR) for Single-Item Subproblem with Setup Carryover and Backlogging

The dynamic programming recursion (DPR) formulation uses a network representation of the single-item lot-sizing problems that integrates the setup status of machines into the state space. The single item uncapacitated sub-problem with setup carryover is shown in the graph $\mathcal{G} = (\mathcal{N}, \mathcal{A})$ in Fig. 2. Each node $\mathcal{N} = [(t_1, t_2)|t_1 \geq t_2]$ represents that production of a particular item starts at period t_1 to meet the demand from period $t_2|t_1 \geq t_2$. The arc $\mathcal{A} = [\left(t_1, t_2\right), \left(t_3, t_4\right)|t_2 \leq t_1 < t_4 \leq t_3]$ indicates the production of a particular item in period t_1 to satisfy demands in periods t_2 through$\left(t_4 - 1\right)$, and t_3 is the next period of production.

Properties 1 and 2 hold.

Property 1 *There are nodes* (t_1, t_2) *with* $t_2 \leq t_1 \forall t_1 = t_2, \dots T - 1$ and $t_2 = 1, \dots T + 1$.

Proof of property 1 We are assuming that another production period will not take place in the window we are satisfying the demand. This is called zero inventory property. If $t_2 > t_1$, which means that production at t_1 meets demand starting from period $t_2 | t_2 > t_1$ and the demand of periods $t_1, \dots, (t_2 - 1)$ is met by production at period $t'_1 < t_1$. Thus, zero inventory property is violated at t_1. In this case, a better production schedule is obtained by shifting the demand of periods $t_1, \dots, (t_2 - 1)$ from period t'_1 to t_1. This modified production schedule saves cost of holding demand of periods $t_1, \dots, (t_2 - 1)$ without increasing any setup cost or any other costs. Therefore, $t_2 \leq t_1$

Property 2 *Now let us consider nodes* (t_1, t_2) *and* (t_3, t_4), *where* $t_2 \leq t_1$ *and* $t_4 \leq t_3$. *There is an arc* (t_1, t_2) to(t_3, t_4) *if and only if* $t_1 < t_4$.

Proof of property 2 The arc from (t_1, t_2) to(t_3, t_4) means that production in t_1 is followed by production in t_3. Production in t_1 meets the demand of periods $t_2, \dots, t_1, \dots, (t_4 - 1)$ and production t_3 meets the demand of periods $t_4, \dots, t_3$ and more. Therefore $t_1 < t_4$

Observation 1: The number of arcs that can be eliminated from any node (t_1, t_2) to $(t_3, t_4) | t_2 \leq t_1 < t_4 \leq t_3$ is $\sum_{r=1}^{t_1} (T - t_r + 1)$.

Given $1 \leq \tau < t < t' \leq T$, let us assume that production in period t satisfies demands in periods $t$through t', and it also satisfies the backlogged quantities from periods τ through $(t - 1)$. Let SC_t^j be the total setup cost of item j in period t, $PC_t^j(\tau, t')$ be the total production cost to satisfy demands of item j in periods τ through $(t' - 1)$ and $HC_t^j(t, t')$ be the total holding cost to hold item j in periods t through $(t' - 1)$ and $BC_t^j(\tau, t)$ be the total backlogging cost to satisfy demands in periods τ through$(t - 1)$ by the production of item j in period t. These cost functions can be defined as follows:

$$\left.\begin{array}{c} SC_t^j = c_j - w_{it} s_j - \gamma \hat{s}_j \\ PC_t^j(\tau, t') = (P_{jt} - w_{it} p_j - \gamma \hat{p}_j) \sum_{r=\tau}^{t'-1} D_{jr} \\ HC_t^j(t, t') = (h_j - \gamma \hat{h}_j) \sum_{r=t}^{t'-1} (r - t) D_{jr} \\ BC_t^j(\tau, t) = \beta_j \sum_{r=\tau}^{t-1} (t - r) D_{jr} \end{array}\right\} \tag{29}$$

For $1 \leq t_2 \leq t_1 < t_4 \leq t_3 \leq T$, let $f_j[(t_1, t_2), (t_3, t_4)]$ be the total cost to satisfy demands in periods t_2 through $(t_4 - 1)$ by the production of item j in period t_1, and t_3 be the next production period.

$$f_j[(t_1,t_2),(t_3,t_4)] = \begin{cases} SC^j_{t_1} + PC^j_{t_1}\left(t_1,t_4\right) + HC^j_{t_1}\left(t_1,t_4\right) if t_1 = t_2, \alpha_{j(t_1-1)} = 0 & (30) \\ SC^j_{t_1} + PC^j_{t_1}\left(t_1,t_1+1\right) + y_{it_1} + PC^j_{t_1+1}\left(t_1+1,t_4\right) + HC^j_{t_1+1}\left(t_1+1,t_4\right) \\ \quad if\ t_1 = t_2, t_1 < t_4 - 1\ and\ \alpha_{jt_1} = 1, t_3 = t_1 + 1, i \in \rho(j) & (31) \\ SC^j_{t_1} + PC^j_{t_1}\left(t_2,t_4\right) + HC^j_{t_1}\left(t_1,t_4\right) + BC^j_{t_1}\left(t_2,t_1\right) if t_2 < t_1 < t_4\ and\ \alpha_{jt_1} = 0 & (32) \\ SC^j_{t_1} + PC^j_{t_1}\left(t_2,t_1+1\right) + BC^j_{t_1}\left(t_2,t_1\right) + y_{it_1} + PC^j_{t_1+1}\left(t_1+1,t_4\right) + HC^j_{t_1+1}\left(t_1+1,t_4\right) \\ \quad if\ t_2 < t_1 < t_4 - 1\ and\ \alpha_{jt_1} = 1, t_3 = t_1 + 1, i \in \rho(j) & (33) \end{cases}$$

$t_1 = t_2$ in expression (30) represents a setup in period $t_{1,}$ followed by the production for the demands of periods t_1 through $(t_4 - 1)$. This schedule does not have any setup carryover from period $(t_1 - 1)$ to t_1 and hence, $\alpha_{j(t_1-1)} = 0$. Expression (31) includes a schedule where there is a setup and production in period t_1 equal to the demand of period t_1, followed by carryover ($\alpha_{jt_1} = 1$) onto period $\left(t_1 + 1\right)$, and production in period $(t_1 + 1)$ equal to the demands of period $(t_1 + 1)$ through $(t_4 - 1)$. The case of $t_2 < t_1 < t_4$ and no carryover ($\alpha_{jt_1} = 0$) is addressed in expression (32) where the setup is done in period t_1, and the production in period t_1 amounts to the demands of periods t_2 though $(t_4 - 1)$, considering the backlogged quantities of periods t_2 though $(t_1 - 1)$ and holding inventories of periods t_1 though $(t_4 - 1)$. Expression (33) indicates a schedule for production and setup in $t_1 | t_2 < t_1 < t_4 - 1$, production of demands of periods t_2 through t_1, along with the backlogged quantities of periods t_2 though $(t_1 - 1)$, setup carryover to the period $(t_1 + 1)$, production in period $(t_1 + 1)$ equal to the demands of periods $(t_1 + 1)$ through $(t_4 - 1)$ and holding inventories of periods $(t_1 + 1)$ though $(t_4 - 1)$.

For $1 \le k \le T + 1$, let $V_j(k)$ be the minimum cost of satisfying demand in periods 1 through $(k - 1)$ for item j. Defining $V_j(1) = 0 \forall j$, we have the following DP recursion:

$$V_j(k) = \min_{1 \le k' \le t' < k \le t''} [V(k') + f_j[(t',k'),(t'',k)]] \tag{34}$$

To analyze the computational complexity of recursion (34), it takes $O(T)$ time to obtain SC^j_t and $O(T^2)$ time to obtain $PC^j_t\left(t',t''\right), HC^j_t\left(t',t''\right)$, and $BC^j_t\left(\tau,\tau'\right)$ for all $1 \le \tau \le \tau' \le t < t' \le t'' \le T$ from Eq. (29). It is noted that after an $O(T^2)$ time preprocessing step, each $f_j[(t_1,t_2),(t_3,t_4)]$ where $1 \le t_2 \le t_1 < t_4 \le t_3 \le T$ can be evaluated in constant time via Eq. (30) through (33). Once these values are available, $V_j(k) \forall 1 \le k \le T + 1$ can be obtained in $O(T^3)$ time.

3.4 Setup Carryover Assignment

The problem of setup carryover assignment can be described as follows. If an item j is produced both in period t and $(t + 1)$ and a setup is performed in both periods, the second setup can be replaced by a setup carryover if the item is produced at the end of period t and at the beginning of period $(t + 1)$. This last condition can be fulfilled by only one item that is produced in both period t and $(t + 1)$. This saves both setup time and setup costs, and such savings are attainable by only one item that is produced in both periods t and $(t + 1)$.

In this paper, we extended the setup carryover problem introduced by Chowdhury et al. [37] to a production system consisting of multiple identical machines. An ILP model can be formulated for each machine to determine the setup carryover assignment variable. The objective of the problem is to maximize savings in setup costs. Suppose we are given $S(i,t)\forall i,t$, where $S(i,t)$ is the set of items produced in machine $i\forall i=1,...,m$ in period $t\forall t=1...T$. Let us assume another set $S'(i,t)|S'(i,t)=S(i,t)\cap S(i,t+1)\forall i=1,...,m$ and $t=1\ldots T-1$. Each element of $S'(i,t)$ represents an item that can be carried over from period t to $(t+1)$ to avoid the machine setup for that item in period $(t+1)$. Since, for a particular machine i, only one item can be carried over to the next period, we must pick exactly one element from $S'(i,t)$. Let us introduce the parameters for the problem as follows:

c_j Setup cost saving associated with element $j|j\in S'(i,t)\forall t$

$$q_{jt}=\begin{cases}1 & if\ item\ j\in S'(i,t)\\ 0 & otherwise\end{cases}$$

$$r'_{jt}=\begin{cases}1 & ifq_{jt}=q_{j(t+1)}=1\ and\ if|S'(i,t+1)|>1\\ 0 & otherwise\end{cases}$$

Decision variable:

$$z_{jt}=\begin{cases}1 & if\ item\ j\in S'(i,t) is\ produced\ at\ the\ end\ of\ period\ t\ and\ in\ the\ beginning\ of\ period\ (t+1)\\ 0 & otherwise\end{cases}$$

Model SC:

$$Max\sum_{j\in\varphi(i)}\sum_{t=1}^{T-1}c_jz_{jt}\ldots\forall i \tag{35}$$

Subject to,

$$z_{jt}\le q_{jt}\forall j,t<T \tag{36}$$

$$\sum_{j\in S(i,t)|q_{jt}=1}z_{jt}\le 1\forall t\le T \tag{37}$$

$$z_{jt}+z_{j(t+1)}\le 1\forall j,t<T-1|r'_{jt}=1 \tag{38}$$

$$z_{jt}\in[0,1]\forall j,t \tag{39}$$

The objective function to maximize the setup cost savings for all $i=1..m$ is expressed in Eq. (35). Constraint (36) ensures that an item, which is produced in two consecutive periods, should be carried over to the next period. Constraint (37) states that at most one item can be carried over to the next period. But for

Procedure 1

```
Input: $Z_{jt}$, S(i, t)
Output: $\alpha_{jt}$
Initialization: $\alpha_{jt} = z_{jt}\forall_{j,t}$
Case 1: If |S(i, t)| = 0 then $\alpha_{jt} = \alpha_{j(t-1)}$
Case 2: if |S(i, t)| = 1 then $\alpha_{jt} = 1 | j \epsilon S(i,t)$
Case 3: let $\epsilon$ = random number between 1 and $n|\epsilon \in S\ (i,t)$
if | S(i, t)> 1 and $\sum_{j\in\varphi(t)}\alpha_{jt} = 0\ then\ \alpha_{\in t} = 1$
```

some t, if $q_{jt} = 0\forall j \in S(t)$, $\sum_{j\in S(i,t)} z_{jt} = 0$. Constraint (38) implies the condition that if the setup state of item j is carried over from period t to $(t+1)$, then j cannot be carried over from period $(t+1)$ to $(t+2)$. Finally, the type of variables is defined in constraint (39). The LP relaxation of the model SC is obtained by relaxing the constraint (39) as $z_{jt} \geq 0\forall j, t$. We determine the setup carryover variable α_{jt} by applying Procedure 1.

4 Proposed DW Decomposition Heuristic Method

4.1 Outline of the Solution Procedure

Generate an initial set of solutions by applying the following procedure:

Step 1.1: From Eq. (29) calculate SC_t^j, $PC_t^j\left(t',t''\right), HC_t^j\left(t',t''\right)$, and $BC_t^j\left(\tau,\tau'\right)$ by fixing the dual variables w_{it}, y_{it}, γ, and v_j a value of zero for the end items j $|j \in \omega$.
Step 1.2: Use SC_t^j, $PC_t^j\left(t',t''\right), HC_t^j\left(t',t''\right)$, and $BC_t^j\left(\tau,\tau'\right)$ as the input for DPR and obtain the optimal production quantity X_{jt} and setup decision Y_{jt} for item j in period t.
Step 1.3: Derive demand for the components $k|k \in \lambda$ as follows: $D_{kt} = \sum_{k\prime\in\Gamma(k)} a_{kk\prime}X_{k\prime t}\forall t$
Step 1.4: Repeat Steps 1.1 and 1.2 for the components. The planned production is determined down to the immediate predecessor level.
Step 1.5: Apply the CA heuristic to make X_{jt} and Y_{jt} feasible (Sect. 4.2).
Step 1.6: Solve the ILP for maximizing setup cost savings presented in Eqs. (31) and (32) and obtain the v alue of the setup carryover decision variable $\alpha_{jt}\forall j, t$ by applying Procedure 1 in Sect. 3.4.
Step 1.7: Use the Y_{jt} values from step 1.5 and α_{jt} from step 1.6 as parameters and solve the model MLCLSP_SCBE to obtain an optimal value for X_{jt}, I_{jt} and B_{jt}.
Solve the LP relaxation of the MP_1 and obtain the dual values of constraints (15) through (18).
Solve the subproblems using the following approach:
Step 3.1: Use the dual values obtained from Step 2 and calculate SC_t^j, $PC_t^j\left(t',t''\right), HC_t^j\left(t',t''\right)$, and $BC_t^j\left(\tau,\tau'\right)$ by using Eq. (29).
Step 3.2: Repeat Steps 1.2 through 1.7.

If there exists at least one new column with negative reduced cost, add such columns to MP_1 and start from Step 2 again. Otherwise, stop.

4.2 Description of the Capacity Allocation (CA) Heuristic

This section explains the CA heuristic. The outline and the numerical illustration of the CA heuristic (Sect. 4.3) use the following symbols:

l Index for levels of product hierarchy (from 0 for the end item, to L).
$\pi(l)$ Set of items positioned in level l of the product hierarchy.
Q_{jt} Production quantity for item j in period t obtained from WW solution (capacity constraint relaxed).
X'_{jt} Production quantity for item j in period t obtained from CA heuristic.
Z'_{jt} Allocated capacity for item j in period t in time units.
Y'_{jt} Setup decision for item j in period t obtained from CA heuristic.
I'_{jt} Inventory level of item j in period t obtained from CA heuristic.
$\text{Req}_{\text{Cap}(i,t)}$ Required capacity of machine i in period t in time units.
$\text{Available}_{\text{Cap}(i,t)}$ Available capacity of machine i in period t in time units.
t' Last period before the next period of production obtained from the WW solution.
$(RQ)_j$ Remaining quantity of item j from the WW solution after the production quantity is adjusted in any period.
$(RD)_{j,t}$ Remaining demand of item j in period t that cannot be satisfied due to the limit of the capacity of resource $i|i \in \rho(j)$.
$\text{Unused}_{\text{Cap}_{(i,t)}}$ Unutilized capacity of machine i in period t.
$\text{Allowable}_{j,t}$ Allowable quantity of item j that can be allocated in period t.

The CA heuristic works as follows: The algorithm starts with $t = 1$ and $l = 0$. Let us consider a machine i that is responsible to produce $j|j \in \pi(l)$ is currently overloaded in period t. This overload of machine i is decreased by shifting the production quantity of an item $j|j \in \varphi(i)$ into an earlier or later period. The production quantity of item j is reduced according to the ratio of the allowable capacity and the required capacity of machine i in period t and assigning the production of item j to period t as follows:

$$\mathrm{Z}^{i}_{jt} = (Q_{jt} \times p_j + s_j) \times \frac{Available_{cap(i,t)}}{Req_{cap(i,t)}} \tag{40}$$

$$X^{i}_{jt} = \max\left(\frac{z'_{jt-Sj}}{p_j}, D_{jt} - I'_{j(t-1)}\right) \tag{41}$$

While decreasing the production quantity of any item, one has to remember that a reduction in the production quantity should not lead to a backlog for this item

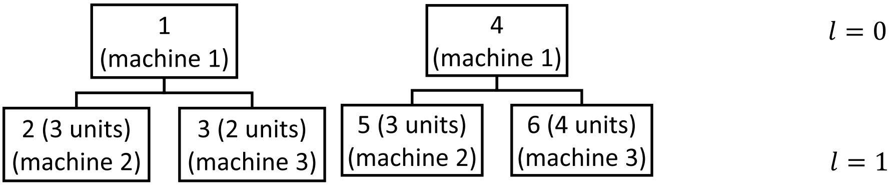


Fig. 3 Product hierarchy structure for the example problem

resulting from successor item demands. That is why it is necessary to adjust the production quantity of the successor item. If there is no further item causing an overload of the resource in question in the current level, then the quantity of the successor items of the product hierarchy is adjusted. For all direct and indirect successors j^i of item j, the maximum quantity that can be decreased is determined according to Eq. (42).

$$X'_{j't} = \max\left(D_{j't} - I'_{j'(t-1)}, \min_{j\epsilon\mu(j')} \frac{X'_{jt}}{a_{jj'}} \right) \tag{42}$$

In the case where the sum of the required production of item j in machine i in period t exceeds the available capacity of machine i in period$t$, we shift the production$[(RD)_{j,t} = D_{jt} - I_{j(t-1)} - X'_{jt}]$ backward into period $\tau|\tau < t$ and$\text{Unused}_{\text{Cap}_{(i,\tau)}} > 0$. Shifting production to the earlier period is possible because the feasibility of the resulting problem instances with respect to the capacity constraints is maintained by ensuring that the cumulative capacity for every period is larger than (or equal to) the cumulative requirement. Because of this shifting to an earlier period, the production quantity of item j in period τ increases. To accommodate the derived demand of the predecessor items$j'|j' \in \mu(j)$, the production quantity of all j' is adjusted as follows: $X'_{j'\tau} = \max(X'_{j'\tau}, D_{j'\tau})$.

If, for all $j|j \in \pi(l)$ and for all $i|i \in \rho(j)$, the available capacity of machine i in period t is allocated among all $j|j \in \varphi(i)$, then we move into the next level of the product hierarchy. When the production quantity of all items j is allocated according to the available capacity of machine i in period t, shift forward the remaining quantity $(RQ)_j$ to period $t'|t' > t$ and assign the production quantity of item j in period t' as follows: $X'_{jt} = \min\left[\text{Allowable}_{jt'}, D_{jt'}, (RQ)_j\right]$. Update $(RQ)_j$. Next, shift the rest of the quantity backward for all $t' = t' - 1, ..t + 1$. A numerical example is presented in Sect. 4.3 to illustrate the CA heuristic.

Table 3 Parameters for the example problem of size ($T \times n \times m = 4 \times 6 \times 3$)

Item (j)	1	2	3	4	5	6
Processing time (p_j)	2	3	2	3	1	2
Setup time (s_j)	15	20	25	30	20	20
Holding cost(h_j)	0.5	0.6	0.2	0.3	0.6	0.2

Table 4 WW solution for end items 1 and 4

Period (t)	1	2	3	4
X_{1t}	45	0	60	0
$Z\prime_{1t}$	105	0	135	0
X_{4t}	110	0	0	0
$Z\prime_{4t}$	360	0	0	0

4.3 Illustrative Example for CA heuristic

Let us consider the following (Fig. 3) product hierarchy structure where there are two end items (items 1 and 4), and each of the end items has two components over the course of four periods. Demand of items 1 and 4 are D_{1t}=(20, 25, 30, 30) and D_{4t}=(25, 20, 30, 35), where t=1.4. Table 3 provides additional parameters for this example problem.

WW for end-items: Each subproblem is an SIULSP. Let, X_{jt} and Z'_{jt} be the production quantity and allocated capacity for item j in period t, respectively. The WW solution and the required capacity for item 1 and 4 at each period is given in Table 4.

Derive demands for components are given in Table 5.

WW for components

The WW solution for items 2, 3, 5, and 6 is given in Table 6.

Feasibility procedure:

Step 4.1: Capacity allocation of the WW solution is shown in Fig. 4. Let $t = 1$ and $l = 0$.

Item 1 and 4 are at level 0 and both of these items are processed by machine 1. The required capacity of machine 1 in period 1 (105 + 360 = 465) exceeds its available capacity in period 1 (Fig. 4a). That is why the production quantity of items 1 and 4 in period 1 is shifted to the later periods.

The available capacity of machine 1 in period 1 is allocated for items 1 and 4 as follows:

$Z'_{11} = \left(105 \times \frac{300}{465}\right) = 67.74$ and $Z'_{41} = \left(360 \times \frac{300}{465}\right) = 232.258$. As a result, the production quantity for item 1 and 4 in period 1 is decreased as follows:

$X'_{11} = [(67.74 - 15)] = 26 \textit{ and } X'_{41} = [(232.258 - 30)/3] = 67$

Derived demand and required capacity for items 2, 3, 5, and 6 in period 1 are 78, 52, 201, and 268, respectively.

Step 4.2: Let $l = 1$. Items 2, 3, 5, and 6 are produced in the next level. The derived demand for item 2 in period 1 is 78 (=26 × 3). But the WW quantity for item 2 in period 1 is 135 (Table 6). Therefore, we take the maximum of 78 and 135 as the

Table 5 Derive demands for components

Period (t)	1	2	3	4
D_{2t}	135	0	180	0
D_{3t}	90	0	120	0
D_{5t}	330	0	0	0
D_{6t}	440	0	0	0

Table 6 WW solution for items 2, 3, 5, and 6

Period (t)	1	2	3	4
X_{2t}	135	0	180	0
X_{3t}	210	0	0	0
X_{5t}	330	0	0	0
X_{6t}	440	0	0	0

production quantity of item 2 in period 1. The production quantity of items 2, 3, 5, and 6 and the required capacity of machines 2 and 3 in period 1 is computed and shown in Table 7.

Thus, the required capacity of machines 2 and 3 in period 1 is 775 and 1345 time units, respectively, which exceeds the available capacity of machines 2 and 3 in period 1 (Fig. 4b).

Step 4.3: The available capacity of machine 2 in period 1 is allocated for items 2 and 5 as follows:

$Z'_{21} = \left(425 \times \left(\frac{400}{775}\right)\right) = 219.35$ and $Z'_{51} = \left(350 \times \left(\frac{400}{775}\right)\right) = 180.65$. As a result, the production quantity for items 2 and 5 in period 1 decreases as follows: $X'_{21} = [(219.35 - 20/3)] = 66$ and $X'_{51} = [(180.65 - 20)/1] = 160$. Similarly, the production quantity for items 3 and 6 in period 1 is decreased to $X'_{31} = 70$ and $X'_{61} = 157$.

The required capacity of machine 2$= 66 \times 3 + 20 \times 1 + 160 \times 1 + 20 \times 1 = 398 < 400$

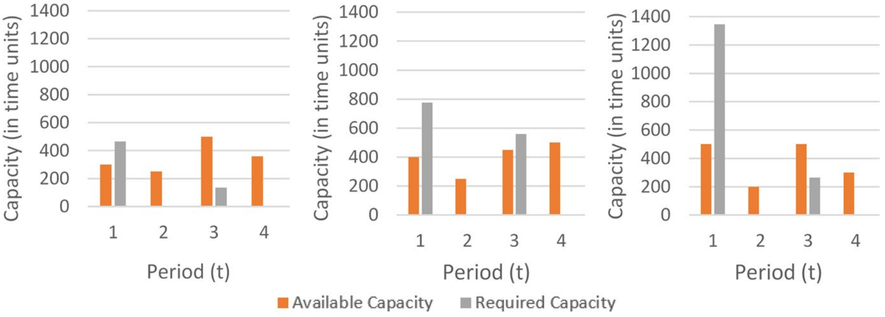
The required capacity of machine 3$= 70 \times 2 + 25 \times 1 + 157 \times 2 + 20 \times 1 = 499 < 500$

If the required capacity exceeds the available capacity, then repeat steps 4.2 and 4.3.

Step 4.4: Compute production quantity of the successor items: $X'_{11} = \max\left(20, \min\left(\frac{66}{3}, \frac{70}{2}\right)\right) = 22$ and $X'_{41} = \max\left(25, \min\left(\frac{160}{3}, \frac{157}{4}\right)\right) = 39$

Step 4.5: Update the production quantity of predecessors.

$$\begin{aligned} X'_{21} &= \max(66, 22 \times 3) = 6, 6X'_{31} = \max(70, 22 \times 2) \\ &= 70, X'_{51} = \max(160, 39 \times 3) = 160, X'_{61} \\ &= \max(157, 39 \times 4) = 157. \end{aligned}$$



(a) Machine 1 (b) Machine 2 (c) Machine 3

Fig. 4 Capacity allocation of WW solution

Table 7 Production quantity of item 2, 3, 5 and 6 and the required capacity of machine 2 and 3

Item (j)	X'_{j1}	Z'_{j1}	Machine
2	max(78,135) = 135	425	2
3	max(52,210) = 210	445	3
5	max(201,330) = 330	350	2
6	max(268,440) = 440	900	3

Step 4.6: Shift the production quantity for each item to the period (t') before the next production period obtained from the WW schedule, and then shift the excess production forward. For any item j, if $X_{jt} = 0 \forall t > 1$, then assign $t' = T$.

The CA heuristic procedure for the example problem is illustrated in Fig. 5.

Similarly, the capacity allocation for items 3 and 6 is shown in Table 8.

Step 4.7: $t = t + 1$ and repeat steps 4.1 to 4.6 until $t = T$. The feasible solution after the capacity allocation is completed is shown in Table 9, and the capacity allocation of a feasible solution is shown in Fig. 6.

Assign setup decision variables. For the example problem, $Y'_{jt} = 1$ for all j and t except $Y'_{3,4} = Y'_{5,2} = 0$.

LP-based improvement procedure: Solve the original problem as an LP given the setup variables. The setup decisions (Y'_{jt}) provided by the CA heuristic are used as parameters in the relaxed LP model to refine the solution. As a result, the refined solution becomes optimum for a particular setup decision (Table 10) Furthermore, if the setup decisions are correct, then the solution obtained using the local search method provides the optimum solution.

5 Computational Study

5.1 Dataset Used

The performance of the proposed DW decomposition and the CG procedure with the CA heuristic is tested using the test data set introduced by Tempelmeier and Derstroff [10]. This is a 75 problem-instance set of class B and class D as shown in Fig. 7. Class B test cases are comprised of 10 items, 4 periods, and 3 machines whereas class D problems are comprised of 40 items, 16 periods, and 6 machines. The 75 instances were generated using combinations of the following coefficients of variance, setup cost structures, and capacity utilization profiles:

Three demand structures with varying coefficients of variance (CV = 0.1, 0.4, 0.7).

Five setup cost structures resulting in different profiles of average time between orders (TBO).

Five capacity utilization profiles (90%, 70%, 50%, 90%/70%/50%, 40%/70%/90%).

Regarding the demand structures, the average demand per period for the end product in the assembly structure is set to 100. As for the setup cost structures, the numbers delimited by slashes are TBO values for higher, middle, and lower levels of the product hierarchy. Note that TBO is an acronym introduced by Tempelmier and Destroff (1996), which

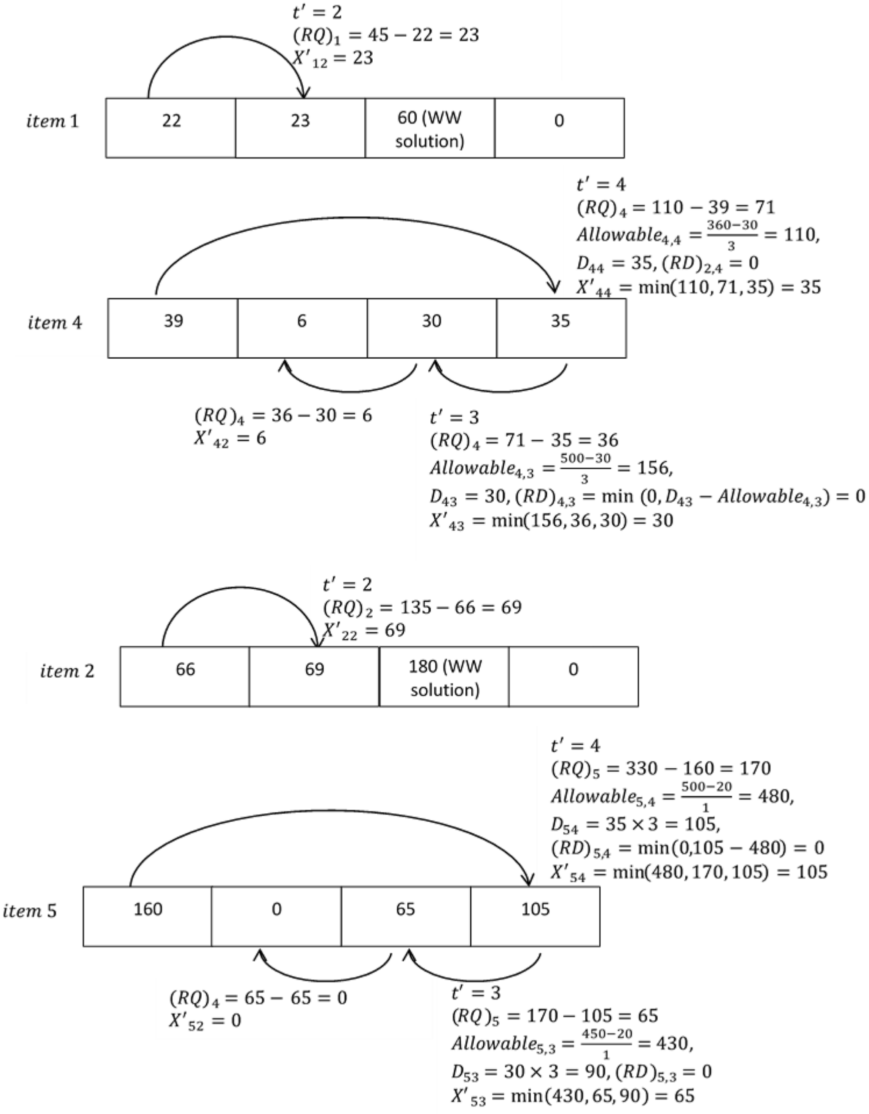

$t' = 2$
$(RQ)_1 = 45 - 22 = 23$
$X'_{12} = 23$
item 1
22
23
60 (WW solution)
0
$t' = 4$
$(RQ)_4 = 110 - 39 = 71$
$Allowable_{4,4} = \frac{360-30}{3} = 110,$
$D_{44} = 35, (RD)_{2,4} = 0$
$X'_{44} = \min(110, 71, 35) = 35$
item 4
39
6
30
35
$(RQ)_4 = 36 - 30 = 6$
$X'_{42} = 6$
$t' = 3$
$(RQ)_4 = 71 - 35 = 36$
$Allowable_{4,3} = \frac{500-30}{3} = 156,$
$D_{43} = 30, (RD)_{4,3} = \min\ (0, D_{43} - Allowable_{4,3}) = 0$
$X'_{43} = \min(156, 36, 30) = 30$
$t' = 2$
$(RQ)_2 = 135 - 66 = 69$
$X'_{22} = 69$
item 2
66
69
180 (WW solution)
0
$t' = 4$
$(RQ)_5 = 330 - 160 = 170$
$Allowable_{5,4} = \frac{500-20}{1} = 480,$
$D_{54} = 35 \times 3 = 105,$
$(RD)_{5,4} = \min(0{,}105 - 480) = 0$
$X'_{54} = \min(480, 170, 105) = 105$
item 5
160
0
65
105
$(RQ)_4 = 65 - 65 = 0$
$X'_{52} = 0$
$t' = 3$
$(RQ)_5 = 170 - 105 = 65$
$Allowable_{5,3} = \frac{450-20}{1} = 430,$
$D_{53} = 30 \times 3 = 90, (RD)_{5,3} = 0$
$X'_{53} = \min(430, 65, 90) = 65$

Fig. 5 Illustration of the working principle of CA heuristic for items 1, 2, 4, and 5

Table 8 Capacity allocation for items 3 and 6

Item	Period (t)			
	1	2	3	4
3	70	20	120 (WW solution)	0
6	157	36	107	140

Table 9 A feasible solution after the CA heuristic is completed

Item(j)	Period (t)			
	1	2	3	4
2	22	23	30	30
2	66	69	115	65
3	70	20	120	0
4	39	6	30	35
5	160	0	65	105
6	157	36	107	140

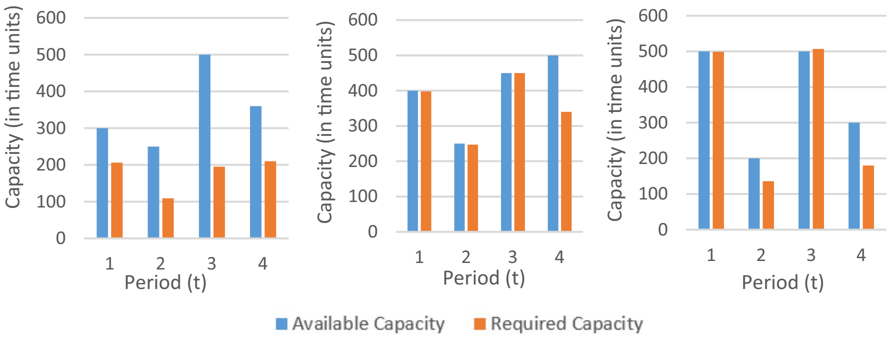
they defined as the average length of a production cycle. Setup cost is computed using the following formula: Setup cost = 0.5 × holding cost × average demand × $(TBO)^2$. Available capacity per period is computed by dividing the mean demand by the target capacity utilization. The setup time profile and the resource assignment for problem classes B and D are shown in Tables 11 and 12, respectively.

For more details about these test instances, see Tempelmeier and Derstroff [10]. In order to allow backlogging, the following modifications of the data made by Wu et al. [14] are being considered:

A ratio of backlogging costs to inventory costs such that $\beta_j = 10h_j \forall j \in \omega$.

The demand for all items is increased by 20% for the first half of the time horizon, while the resource capacities are increased by 10% over the entire time horizon for each test instance.

However, these data sets have no allowance for an emission cap. We, therefore, alter the problem instances to limit the emission while making lot sizing decision. The emission due to setup $(\widehat{c}_j)$, production $(\widehat{p}_j)$ and holding $(\widehat{h}_j)$ activities are all generated from discrete uniform distributions DU (0,20). The emission cap is set to 25,000 and 60,000 tons of carbon for the problem classes B and D, respectively.

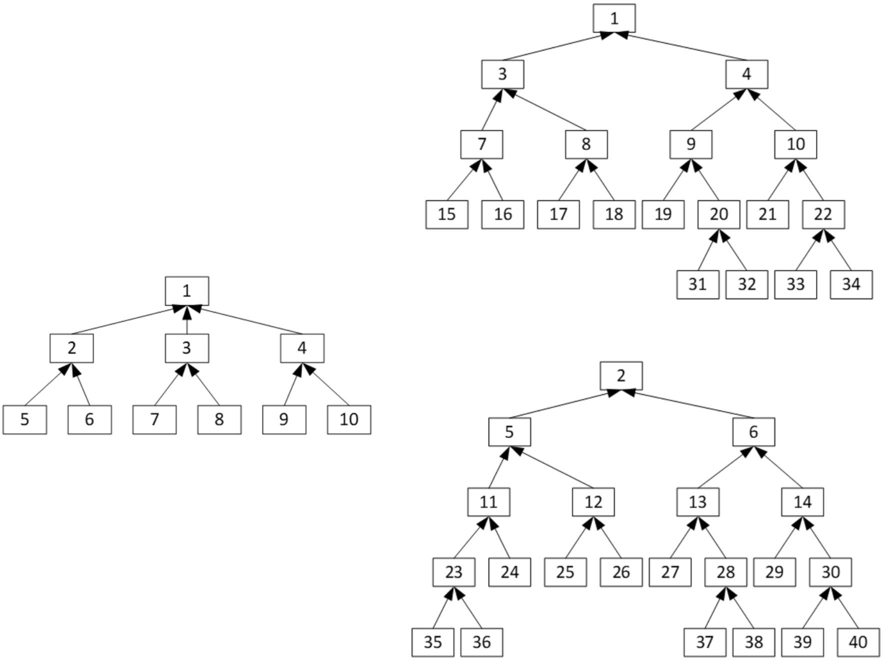


(a) Machine 1 (b) Machine 2 (c) Machine 3

Fig. 6 Capacity allocation of a feasible solution

Table 10 Production schedule (X'_{jt}) after the improvement procedure

Item (j)	Period (t)			
	1	2	3	4
2	20	25	30	30
2	60	75	90	90
3	40	59	111	0
4	46	0	29	35
5	138	0	87	105
6	184	36	116	140



(a) Problem class B

(b) Problem class D

Fig. 7 The assembly product structure used for the experiment [10]

Table 11 Setup time profiles of problem class B and D for the assembly product structure [10]

	Setup time						
	Class B			Class D			
	5	10	15	10	15	20	25
Items	7, 8, 9, 10	1, 2, 5, 6	3, 4	1…10	11….20	21…30	31…40

Table 12 Resource assignment of problem class B and D for the assembly product structure [10]

Resource	Problem class B	Problem class D
A	1…4	1,2
B	5…7	3…6
C	8…10	7…14
D	-	15…22
E	-	23…30
F	-	31…40

6 Performance Evaluation

Our developed DW decomposition and the CG procedure with the CA heuristic are coded using FICO's Mosel (Xpress-IVE Version 1.24.02 64 bit) algebraic modeling language and the commercial MILP solver Xpress 7.6 is used to establish its efficiency. All the test instances are run on a PC with an Intel Core i7 1.8 GHz processor, 8 GB of RAM, and an L2 cache of 512 KB.

Table 13 Average deviation from optimality (%) of the proposed heuristic solutions for class B test cases of 10 items, 4 periods, and 3 machines (C_{cap} = 25,000 tons of carbon)

TBO profile	CV	Utilization profile (%)					Mean
		90	70	50	90/70/50	50/70/90	
1	0.1	0.19	0.19	0.19	0.19	0.19	0.19
	0.4	0.20	0.13	0.19	0.19	0.65	0.27
	0.7	0.64	0.53	0.65	0.19	0.19	0.44
	mean	0.34	0.28	0.34	0.19	0.34	0.30
2	0.1	0.76	1.78	1.71	1.26	0.69	1.24
	0.4	1.08	1.66	0.97	1.15	1.10	1.19
	0.7	1.09	2.81	2.43	1.75	1.20	1.85
	mean	0.98	2.08	1.70	1.39	1.00	1.43
4	0.1	1.81	2.52	1.89	1.04	4.56	2.36
	0.4	1.45	1.69	3.39	1.63	0.95	1.82
	0.7	2.04	2.30	4.49	1.57	4.93	3.07
	mean	1.77	2.17	3.26	1.41	3.48	2.42
1/2/4	0.1	3.14	3.64	3.45	2.26	2.95	3.09
	0.4	5.24	2.26	4.04	2.12	2.12	3.16
	0.7	5.28	3.46	5.07	2.66	0.99	3.49
	mean	4.55	3.12	4.19	2.34	2.02	3.24
4/2/1	0.1	1.15	0.50	2.37	0.35	0.50	0.97
	0.4	0.51	0.50	1.19	0.50	0.93	0.73
	0.7	0.50	1.36	1.46	0.49	0.50	0.86
	mean	0.72	0.79	1.67	0.45	0.64	0.85
	Overall mean (75 problem instances)						1.65

Table 14 The solutions (values of the cost objective function employed) obtained by the proposed heuristic for class B test cases of 10 items, 4 periods, and 3 machines (C_{cap} = 25,000 tons of carbon)

TBO profile	CV	Utilization profile (%)									
		90		70		50		90/70/50		50/70/90	
		MILP	DW	MILP	DW	MILP	DW	MILP	DW	MILP	DW
1	0.1	13,205	13,230	13,205	13,230	13,205	13,230	13,205	13,230	13,205	13,230
	0.4	13,299	13,325	13,207	13,224	13,205	13,230.2	13,287	13,312	13,300.8	13,386.6
	0.7	13,304	13,388.5	13,232	13,302.5	13,144.2	13,230	13,296	13,321	13,364.7	13,389.7
2	0.1	14,470	14,580	14,325	14,580	14,095.3	14,336.8	14,397.9	14,580	14,480	14,580
	0.4	14,578.1	14,736	14,345	14,583	16,284.8	16,442.4	14,494.9	14,662	14,576.4	14,736.6
	0.7	14,579	14,738	14,252.5	14,652.5	14,234.2	14,580.1	14,419.2	14,671	14,565	14,739.7
4	0.1	33,965	34,580	17,538.8	17,980	25,106.3	25,580	17,795.3	17,980	19,108	19,980
	0.4	18,130	18,392	17,680.8	17,980	17,069.4	17,648	17,772.8	18,062	18,064	18,235.6
	0.7	18,526	18,904	17,645.9	18,052.5	17,206.8	17,980	17,792.3	18,071	17,752.6	18,627.6
1/2/4	0.1	17,321.5	17,865	16,273.4	16,865	15,916.2	16,465	16,492.8	16,865	17,353.4	17,865
	0.4	16,954.2	17,843	16,426.9	16,798.7	15,752.8	16,389.8	16,497.8	16,847	16,669	17,021.6
	0.7	17,031.5	17,930.4	16,370.9	16,937.5	16,051.8	16,865	16,517.3	16,956	16,477	16,640.9
4/2/1	0.1	12,436.4	12,600	13,970	14,040	12,308.2	12,600	13,970	14,040	13,970	14,040
	0.4	14,061.5	14,133	13,972.5	14,043	13,875.3	14,040	14,052	14,122	14,066.1	14,196.6
	0.7	14,131	14,201	13,922.5	14,112.5	13,831.7	14,034	14,062	14,131	14,129.7	14,199.7

With respect to the class B test instances, the computational results are given in Tables 13 and 14. Since the test instances of class B are relatively small problems, the commercial MILP solver Xpress 7.6 is able to optimally solve the MLCLSP with SCBE model (Eqs. 1 and 13) for all the test instances in this class. The optimality gap is calculated using the formula, $\%\text{gap} = \frac{\text{Heuristic Solution}-\text{Optimum solution}}{\text{Optimum Solution}} \times 100$. According to the results listed in Table 13, it can be seen that the overall mean deviation from optimality is 1.65%. We provide the solutions (values of the cost objective function employed) in Table 14. Table 15 demonstrates the computational time (seconds) required by the DW heuristic solutions for class B test cases. It is noted from Table 15 that the average solution time using DW heuristic is improved by 54.9% over the MIP solver solution time.

The computational results of class D problem instances are given in Tables 16 and 17. Since the problem size of the class D test instances is relatively larger, MIP solver Xpress 7.6 could not solve any of the instances. Table 16 shows the average percent deviation of the proposed heuristic solutions from the lower bounds for problem class D. The gaps are calculated using the formula, $\%\text{gap} = \frac{UB_{\text{heuristic}} - LB_{MILP}}{LB_{MILP}} \times 100$. Here, $UB_{\text{heuristic}}$ denotes the upper bounds achieved by our developed DW decomposition heuristic. According to the results listed in Table 16, it can be seen that the overall mean deviation from optimality is 9.23%.

Table 15 Solution time (seconds) required by the proposed heuristic solutions for class B test cases of 10 items, 4 periods, and 3 machines (C_{cap} = 25,000 tons of carbon)

TBO profile	CV	Utilization profile (%)									
		90		70		50		90/70/50		50/70/90	
		MILP	DW	MILP	DW	MILP	DW	MILP	DW	MILP	DW
1	0.1	8.83	2.56	5.68	4.27	9.33	4.43	10.65	4.62	9.90	3.32
	0.4	9.82	3.36	5.41	3.85	8.08	4.81	8.93	2.16	9.15	3.57
	0.7	8.78	4.35	4.26	4.40	7.49	2.74	8.20	4.17	9.45	2.49
2	0.1	9.38	4.40	4.79	4.71	9.35	2.18	10.49	2.18	8.59	3.79
	0.4	7.64	3.73	6.81	4.90	10.06	2.68	9.59	3.89	11.51	3.40
	0.7	10.10	2.75	6.94	3.08	7.99	2.74	9.88	3.74	8.20	2.67
4	0.1	8.25	4.22	5.66	3.29	9.81	2.87	11.41	3.50	8.82	3.80
	0.4	10.71	2.68	5.56	3.39	8.01	3.94	11.06	4.84	11.19	2.00
	0.7	9.91	3.47	5.33	4.83	9.64	2.91	7.34	2.69	11.77	3.59
1/2/4	0.1	10.96	3.75	4.30	3.94	9.79	3.15	10.46	3.07	11.19	2.31
	0.4	7.56	3.78	4.40	3.93	8.70	3.48	9.81	4.89	8.12	3.68
	0.7	8.12	2.79	5.24	4.17	7.55	4.03	9.25	3.17	10.59	3.60
4/2/1	0.1	8.73	2.92	5.34	3.27	10.18	3.21	7.93	4.20	9.12	3.58
	0.4	8.83	2.56	5.68	4.27	9.33	4.43	10.65	4.62	9.90	3.32
	0.7	9.82	3.36	5.41	3.85	8.08	4.81	8.93	2.16	9.15	3.57

Table 16 Average deviation (%) of the proposed heuristic solutions from the lower bounds for class D test cases of 40 items, 16 periods, and 6 machines (C_{cap} = 60,000 t/MWh)

TBO profile	CV	Utilization profile (%)					
		90	70	50	90/70/50	50/70/90	Mean
		DW	DW	DW	DW	DW	DW
1	0.1	4.08	7.65	7.88	9.98	11.36	8.19
	0.4	6.98	7.85	6.76	8.77	10.67	8.21
	0.7	7.56	8.45	6.56	8.45	10.45	8.29
	Mean	6.21	7.98	7.07	9.07	10.83	8.23
2	0.1	8.98	8.96	8.09	9.87	10.87	9.35
	0.4	7.44	9.76	8.76	10.27	11.54	9.55
	0.7	6.95	10.35	8.98	9.56	9.48	9.06
	Mean	7.79	9.69	8.61	9.90	10.63	9.32
4	0.1	7.45	9.52	9.65	8.67	9.34	8.91
	0.4	8.97	9.65	8.54	9.44	9.59	9.24
	0.7	9.55	10.30	9.47	10.35	8.86	9.71
	Mean	8.66	9.79	9.22	9.49	9.26	9.28
1/2/4	0.1	7.34	7.56	10.45	9.49	10.34	9.04
	0.4	7.29	8.34	10.77	10.87	9.78	9.41
	0.7	9.95	8.96	10.89	11.55	9.56	10.18
	Mean	8.19	8.29	10.70	10.64	9.89	9.54
4/2/1	0.1	7.98	8.67	9.56	8.78	10.66	9.13
	0.4	8.76	9.46	10.14	9.89	11.43	9.94
	0.7	8.45	9.65	10.65	11.66	10.78	10.24
	Mean	8.40	9.26	10.12	10.11	10.96	9.77
	Overall mean (75 problem instances)						9.23

For reference, we provide the solutions (values of the cost objective function employed) in Table 17. The column LB_{MILP} in Table 17 denotes the lower bound obtained by solving the LP relaxation of the corresponding MLCLSP_SCBE formulation after a running time of 1800s. The columns associated with UB_{DW} provide the upper bounds achieved by the DW decomposition heuristic. It was mentioned in Sect. 5.1 that we consider the same modifications to problem classes B and D made by Wu et al. [14] to allow backlogging. Wu et al. [14] show that their progressive time-oriented heuristic (PTH) lowers the optimality gaps by around 7.5% for the modified class B and 25% for the modified class D when compared to CPLEX. In contrast, as compared to the MIP solver's solution, the DW decomposition with the CA heuristic reduces the optimality gaps by roughly 1.65% for class B and 9.23% for class D test cases.

Table 17 The solutions (values of the cost objective function employed) obtained by the DW heuristic for class D test cases of 40 items, 16 periods, and 6 machines (C_{cap} = 60,000 tons of carbon)

TBO profile	CV	Utilization profile (%)									
		90		70		50		90/70/50		50/70/90	
		LB_{MILP}	UB_{DW}	LB_{MILP}	UB_{DW}	LB_{MILP}	UB_{DW}	LB_{MILP}	UB_{DW}	LB_{MILP}	UB_{DW}
1	0.1	1,045,678	5,312,044	1,048,656	9,070,874	1,039,567	9,231,355	1,056,754	11,603,159	1,032,872	12,766,295
	0.4	1,047,854	8,361,875	1,045,765	9,255,020	1,043,567	8,098,080	1,058,975	10,346,186	1,054,546	10,302,918
	0.7	1,056,397	9,042,758	1,052,876	9,949,678	1,046,985	7,915,207	1,067,878	10,091,447	1,095,094	10,348,635
2	0.1	1,167,543	11,652,079	1,067,548	10,632,778	1,034,564	9,404,187	1,033,724	11,236,584	1,674,560	18,202,465
	0.4	1,154,678	9,745,482	1,087,654	11,703,157	1,055,643	10,303,076	1,055,150	11,891,535	1,562,971	17,614,680
	0.7	1,187,546	9,440,991	1,123,432	12,750,953	1,096,545	10,943,519	1,095,725	11,570,858	1,562,304	16,497,933
4	0.1	1,203,543	10,169,938	1,676,578	17,637,600	1,675,788	17,847,139	1,675,265	16,199,814	1,686,009	16,303,702
	0.4	1,209,856	12,062,264	1,564,786	16,664,971	1,564,046	14,920,997	1,563,500	16,322,938	1,455,126	15,191,519
	0.7	1,213,456	12,801,961	1,564,345	17,677,099	1,563,759	16,372,555	1,562,970	17,739,712	1,433,919	16,274,977
1/2/4	0.1	1,267,987	10,575,012	1,687,654	14,446,318	1,687,005	19,316,207	1,686,357	17,689,888	1,342,059	14,078,194
	0.4	1,276,543	10,582,541	1,456,543	13,604,112	1,455,988	17,136,979	1,455,513	17,276,942	1,303,652	15,474,352
	0.7	1,268,709	13,892,364	1,435,467	14,297,251	1,435,118	17,063,547	1,434,676	18,005,178	1,374,960	17,255,753
4/2/1	0.1	1,308,657	11,751,740	1,343,876	12,995,281	1,343,425	14,186,564	1,342,551	13,130,144	1,324,345	12,952,094
	0.4	1,376,545	13,435,079	1,305,650	13,657,099	1,305,203	14,539,961	1,304,332	14,204,172	1,398,767	15,232,573
	0.7	1,387,659	13,113,378	1,376,548	14,660,236	1,376,166	16,032,336	1,375,459	17,413,316	1,675,866	21,216,464

7 Conclusion

This paper addresses the extended classical MLCLSP where setup carryover, backlogging, and emission control are being incorporated (MLCLSP_SCBE). An item DW decomposition technique is developed to decompose the MLCLSP_SCBE into a master problem and a number of uncapacitated dynamic single-item lot-sizing problems, which are solved by combining dynamic programming and a multi-step iterative capacity allocation (CA) heuristic approach. An ILP model is developed to determine the setup carryover variable for a given production schedule. An LP-based post-improvement procedure is implemented to refine the solution. The capacity constraints are being taken into consideration implicitly through the dual variables, which are updated using a column generation procedure. The performance of the heuristic is tested by comparing the average percentage of deviation from optimality. The quality of the heuristic for MLCLSP_SCBE is tested using the two test data sets namely class B and class D introduced by Tempelmeier and Derstroff [10]. The results show that the overall mean deviation from optimality is 1.65% (for class B test cases) and 9.23% (for class D test cases) as the proposed heuristic-based decomposition approach is applied.

For future work, the proposed model can be extended to include constraints of limited inventory storage capacity, lot transportation, to name a few. In the future, period decomposition and resource decomposition can be considered for comparing the effectiveness of each of these techniques. Also, metaheuristic approaches, math-heuristics, or the newer hyper-heuristics may be employed to investigate if the percentage gap from optimality improves using the sub-optimal solutions obtained by such solution approaches.

Acknowledgements I would like to express my sincere gratitude to my supervisors, Professor Dr. Fazle Baki and Professor Dr. Ahmed Azab for their valuable guidance and support throughout the research process.

Author Contribution Nusrat T. Chowdhury is the major contributor in writing the manuscript. M.F. Baki and A. Azab did the overall supervision while writing the manuscript. All authors read and approved the final manuscript.

Funding This research is partially funded by the Natural Sciences and Engineering Research Council's (NSERC) Discovery Grants (A/C # 811008). This research is also partially funded by the Research and Teaching Innovation Fund (RTIF) and the internal faculty funds.

Data Availability The authors confirm that the data supporting the findings of this study are available within the article.

Declarations

Ethical Approval All authors certify that for this manuscript the following is fulfilled. This material is the author's own original work, which has not been previously published elsewhere. The paper is not currently being considered for publication elsewhere. The paper reflects the authors' own research and analysis in a truthful and complete manner. The paper properly credits the meaningful contributions of co-authors and co-researchers.

Competing Interests The authors declare no competing interests.

References

1. Billington PJ, McClain JO, Thomas LJ (1983) Mathematical programming approaches to capacity-constrained MRP systems: review, formulation and problem reduction. Manage Sci 29(10):1126–1141. https://doi.org/10.1287/mnsc.29.10.1126
2. Buschkühl L, Sahling F, Helber S, Tempelmeier H (2010) Dynamic capacitated lot-sizing problems: a classification and review of solution approaches. OR Spectrum 32(2):231–261. https://doi.org/10.1007/s00291-008-0150-7
3. Briskorn D (2006) A note on capacitated lot sizing with setup carry over. IIE Trans 38(11):1045–1047. https://doi.org/10.1080/07408170500245562
4. Haase K (1998) Capacitated lot-sizing with linked production quantities of adjacent periods. Beyond Manufacturing Resource Planning (MRP II). Advanced Models and Methods for Production Planning. Springer Berlin Heidelberg., Berlin, Heidelberg, pp 127–146
5. Chen X, Benjaafar S, Elomri A (2013) The carbon-constrained EOQ. Oper Res Lett 41(2):172–179. https://doi.org/10.1016/j.orl.2012.12.003
6. Wagner HM, Whitin TM (1958) Dynamic version of the economic lot size model. Manage Sci 5(1):89–96. https://doi.org/10.1287/mnsc.5.1.89
7. Federgruen A, Tzur M (1991) A simple forward algorithm to solve general dynamic lot sizing models with n periods in 0(n log n) or 0(n) time. Manag Sci 37(8):909–925. https://doi.org/10.2307/2632555
8. Florian M, Lenstra JK, Kan AHGR (1980) Deterministic production planning: algorithms and complexity. Manag Sci 26(7):669–679. https://doi.org/10.1287/mnsc.26.7.669
9. Bitran GR, Yanasse H (1982) Computational complexity of the capacitated lot size problem. Manag Sci 28(10):1174–1186. https://doi.org/10.1287/mnsc.28.10.1174
10. Tempelmeier H, Derstroff M (1996) A Lagrangean-based heuristic for dynamic multilevel multiitem constrained lotsizing with setup times. Manage Sci 42(5):738–757
11. Sox CR, Gao Y (1999) The capacitated lot sizing problem with setup carry-over. IIE Trans 31(2):173–181. https://doi.org/10.1023/a:1007520703382
12. Tempelmeier H, Buschkühl L (2009) A heuristic for the dynamic multi-level capacitated lotsizing problem with linked lotsizes for general product structures. OR Spectrum 31(2):385–404. https://doi.org/10.1007/s00291-008-0130-y
13. Sahling F, Buschkühl L, Tempelmeier H, Helber S (2009) Solving a multi-level capacitated lot sizing problem with multi-period setup carry-over via a fix-and-optimize heuristic. Comput Oper Res 36(9):2546–2553. https://doi.org/10.1016/j.cor.2008.10.009
14. Wu T, Akartunalı K, Song J, Shi L (2013) Mixed integer programming in production planning with backlogging and setup carryover: modeling and algorithms. Discrete Event Dynamic Systems 23(2):211–239. https://doi.org/10.1007/s10626-012-0141-3
15. Carvalho DM, Nascimento MCV (2018) A kernel search to the multi-plant capacitated lot sizing problem with setup carry-over. Comput Oper Res 100:43–53. https://doi.org/10.1016/j.cor.2018.07.008
16. Ghirardi M, Amerio A (2019) Matheuristics for the lot sizing problem with back-ordering, setup carry-overs, and non-identical machines. Comput Ind Eng 127:822–831. https://doi.org/10.1016/j.cie.2018.11.023
17. Fiorotto DJ, del Carmen J, Neyra H, de Araujo SA (2020) Impact analysis of setup carryover and crossover on lot sizing problems. Int J Prod Res 58(20):6350–6369. https://doi.org/10.1080/00207543.2019.1680892
18. Benjaafar S, Li Y, Daskin M (2013) Carbon footprint and the management of supply chains: insights from simple models. IEEE Trans Autom Sci Eng 10(1):99–116. https://doi.org/10.1109/TASE.2012.2203304
19. Retel Helmrich MJ, Jans R, van den Heuvel W, Wagelmans APM (2015) The economic lot-sizing problem with an emission capacity constraint. Eur J Oper Res 241(1):50–62. https://doi.org/10.1016/j.ejor.2014.06.030
20. Mashud AHM, Roy D, Daryanto Y, Mishra U, Tseng M (2022) Sustainable production lot sizing problem: a sensitivity analysis on controlling carbon emissions through green investment. Comput Ind Eng 169(108143)
21. Mashud AHM, Roy D, Daryanto Y, Chakrabortty RK, Tseng M (2021) A sustainable inventory model with controllable carbon emissions, deterioration and advance payments. J Clean Prod 296(126608)

22. Sepehri A (2021) Controllable carbon emissions in an inventory model for perishable items under trade credit policy for credit-risk customers. Carbon Capture Sci Technol 1(100004)
23. Konur D, Schaefer B (2014) Integrated inventory control and transportation decisions under carbon emissions regulations: LTL vs. TL carriers. Trans Res Part E-Log 68:14–38. https://doi.org/10.1016/j.tre.2014.04.012
24. Ghosh A (2021) Optimisation of a production-inventory model under two different carbon policies and proposal of a hybrid carbon policy under random demand. Int J Sustain Eng 14(3):280–292
25. Ahmadini AAH, Modibbo UM, Shaikh AA, Ali I (2021) Multi-objective optimization modelling of sustainable green supply chain in inventory and production management. Alex Eng J 60(6):5129–5146
26. Manne AS (1958) Programming of economic lot sizes. Manage Sci 4(2):115–135. https://doi.org/10.1287/mnsc.4.2.115
27. Degraeve Z, Jans R (2007) A new Dantzig-Wolfe reformulation and branch-and-price algorithm for the capacitated lot-sizing problem with setup times. Oper Res 55(5):909–920. https://doi.org/10.1287/opre.1070.0404
28. Pimentel CMO, Alvelos FPE, de Carvalho Valério JM (2010) Comparing Dantzig-Wolfe decompositions and branch-and-price algorithms for the multi-item capacitated lot sizing problem. Optimization Methods and Software 25(2):299–319. https://doi.org/10.1080/10556780902992837
29. Caserta M, Voß S (2012) A math-heuristic Dantzig-Wolfe algorithm for the capacitated lot sizing problem. Learning and Intelligent Optimization: 6th International Conference, LION 6, Paris, France, January 16–20, 2012, Revised Selected Papers, 31–41.
30. Fiorotto DJ, de Araujo SA, Jans R (2015) Hybrid methods for lot sizing on parallel machines. Comput Oper Res 63(Supplement C):136–148. https://doi.org/10.1016/j.cor.2015.04.015
31. Ioannis F, Degraeve Z, De Reyck B (2016) A horizon decomposition approach for the capacitated lot-sizing problem with setup times. INFORMS J Comput 28(3):465–482. https://doi.org/10.1287/ijoc.2016.0691
32. Wu T, Shi Z, Liang Z, Zhang X, Zhang C (2020) Dantzig-Wolfe decomposition for the facility location and production planning problem. Comput Oper Res. https://doi.org/10.1016/j.cor.2020.105068
33. Gören HG, Tunali S (2018) Fix-and-optimize heuristics for capacitated lot sizing with setup carryover and backordering. J Enterp Inf Manag 31(6):879–890. https://doi.org/10.1108/JEIM-01-2017-0017
34. Vanderbeck F (2000) On Dantzig-Wolfe decomposition in integer programming and ways to perform branching in a branch-and-price algorithm. Oper Res 48(1):111–128. https://doi.org/10.1287/opre.48.1.111.12453
35. Vanderbeck F, Savelsbergh MWP (2006) A generic view of Dantzig-Wolfe decomposition in mixed integer programming. Oper Res Lett 34(3):296–306. https://doi.org/10.1016/j.orl.2005.05.009
36. Ramsay TEJ (1981) Integer programming approaches to capacitated concave cost production planning problem. (Ph.D Dissertation), Georgia Institute of Technology.
37. Chowdhury NT, Baki F, Azab A (2023) The setup carryover assignment problem. Oper Res Forum 4(25):1–13. https://doi.org/10.1007/s43069-023-00207-6