Journal of Engineering and Applied Sciences 12 (3): 628-635, 2017
ISSN: 1816-949X
© Medwell Journals, 2017

Heuristic for Material and Operations Planning in Supply Chains with Alternative Product Structure

[1]Jairo R. Coronado-Hernandez, [2]Daniela Simancas-Mateus,
[3]Karolina Avila-Martinez and [4]Jose P. Garcia-Sabater
[1]Department of Industrial Engineering, Universidad de la Costa,
Cra. 55 #58-66, Barranquilla, Colombia
[2]Department of Informatics, Technische Universitat Munchen,
BoltzmannstraBe 3, 85748 Garching bei Munchen, Germany
[3]Department of Maritime Management, Escuela Naval de Cadetes
"Almirante Padilla", Isla de Manzanillo, Cartagena de Indias, Colombia
[4]Department of Business Management, Universitat Politècnica de Valencia,
Cami Vera, S/n, 46022 Valencia, Spain

Abstract: This research presents a heuristic system to solve the problem of materials and operations planning in supply chains with alternative product structures. The objective is minimizing the costs associated to stock levels, setup and operations, considering products with alternative bill of materials. The heuristic system can be used as a framework to embed one-level rules into a multi-level problem with alternative bill of materials. With this in mind, this framework can be used to apply planning rules as simple as Wager-Whitin, Silver-Meal, FOQ and L×L, among others.

Key words: Supply chain planning, generic materials and operations planning, planning rule, heuristic system, alternative product structure

INTRODUCTION

The globalized world of the 21st century is a dynamic and changing place. Markets are more competitive and companies have to make an effort to deliver the best to their customers. Market is no longer a city, a country or even a region: it's the entire world and in order for operations in this context to research, planning is compulsory. When a company desires to be able to compete in this demanding environment, its supply chain must stand on two fundamental pillars: integration and coordination (Stadtler and Kilger, 2005). Coordination in a supply chain, consists of the synchronization of information flows, material and capital and at the same time integrates three fundamental blocks: information management and communications technology, process orientation and advanced planning. Moreira *et al.* (2014) it is shown that companies with efficient methods of production planning have competitive advantages to respond quickly to the requirements of customers. In this aspect, the concept of time, volume and capacity become crucial variables of a manufacturing system, so customers are sensitive to delivery times and quality of service (Sana *et al.*, 2014).

Planning is about answering an apparently simple question what is the lotsizing level for the production system? However, operations planning are not a simple subject at all, since it involves a series of decisions that affect the entire supply chain. Given the notable role of time in all the operations inside a supply chain; Stadtler and others propose a matrix (Supply Chain Planning Matrix, SCPM) in which the planning depends on the planning horizon (Stadtler *et al.*, 2012). Inside this matrix, on the medium term fringe, the Material Requirements Planning (MRP) is mentioned: process that as input values has demand, inventory registries and the product structure.

A supply chain can be modeled in various ways, one of these is according to the product structure (Ivanov *et al.*, 2010) for this purpose there are numerous lot-sizing rules, designed for product structures of diverse conditions. For example, the Economic Order Quantity (EOQ) a lot-sizing rule proposed by Ford W. Harris in 1913, focuses on determining the optimal lot size for an only product with stationary demand, under the postulation of infinite capacity. Many after Harris developed different variants of the original EOQ, being one of these the ELSP (Economic Lot Size Scheduling

Corresponding Author: Jairo R. Coronado-Hernandez, Department of Industrial Engineering, Universidad de la Costa, Cra. 55 #58 66, Barranquilla, Colombia

Problem) by Bomberger (1966) where a limited-capacity machine has to be programmed to research on multiple products that have a stationary demand; something called single-level planning.

Many researchers have proposed different heuristic algorithms for when the demand is dynamic: Silver-Meal, Wagner-Whitin who proposed dynamic programing. In the 70's, Orlicky (1975) developed the MRP system as a multi-level planning system where the product has a unique material bill. With time this type of product structure was called Gozinto graph. Orlicky introduces the angle from which a supply chain is represented as a multilevel system where the bill of materials, inventory levels and demand are planned using lot-sizing heuristics in a process called explosion of bill of materials. Billington *et al.* (1983) postulated an optimization model to solve the MRP problem, equivalent to a material planning problem with capacity constraints this model is currently known as MRPII. Most of the models designed after Billington's are based on the same Gozinto bill of materials. For solving this problem, people used to adopt heuristics (e.g., Wagner-Whitin, Silver-Meal etc.) (Silver *et al.*, 1998).

Along with mass customization and JIT (Just In Time) or lean systems, companies need to be more flexible with their materials, producing process deallocation to attend global suppliers (Chryssolouris *et al.*, 2013). As an outcome of this business model, product structures have national and international participants and the bill of materials has several ways to be manufactured. This is an inconvenience for ERP systems which have too rigid structures to be able to plan accurately in these environments and do not consider alternative bills of materials for taking base on the Gozinto structure. Therefore, Sabater *et al.* (2013) propose a mathematical model for supply chains whose product structures have alternative bills of materials, the representation of alternative ways of manufacture was called "stroke". A stroke represents any operation that transforms (or transports) a series of products (measured as SKUs) into another series of products (also measured as SKUs). This operation and therefore the stroke representing it has an associated cost and lead time and consumes a certain amount of resources during the first of the planning periods however, this aspect could be reconsidered in accordance with the specific case.

MATERIALS AND METHODS

Description of the problem: This study contemplates the mathematical formulation of the model problem and the name put forward for this model is material and operations planning in supply chains with alternative product structure problem based in the stroke concept. To mathematically formulate the problem, it is necessary to define the nomenclature presented in Table 1.

Table 1: Indices, parameters and variables of GMOP's Model

Indices	Model
i	Index set of products
t	Index set of planning periods
r	Index set of resources
k	Index set of strokes
Parameters	
N_{ik}	Number of units i that generates a stroke k
M_{ik}	Number of units i that stroke k consumes
LT_k	Lead time of stroke k
$HC_{i,t}$	Cost of storing a unit of product i in period t
$SC_{k,t}$	Cost of the setup of stroke k in period t
$OC_{k,t}$	Cost of the operation of stroke k in period t
$D_{i,t}$	Demand of product i for period t
$RP_{k,r}$	Capacity of the resource r required for performing one unit of stroke k (in time units)
$RS_{k,r}$	Capacity required of resource r for setup of stroke k (in time units)
$KAP_{r,t}$	Capacity availability of resource r in period t (in time units)
Variables	
$\beta_{k,t}$	= 1 if stroke k is performed in t (0 otherwise)
$y_{i,t}$	Stock level of product i on hand at the end of period t
$z_{i,t}$	Amount of strokes k to be performed in period t

The problem we intend to solve is the same of Generic Materials and Operations Planning (GMOP) proposed by Sabater *et al.* (2013)which consists on planning according to the operation, unlike how Orlicky presented it. GMOP Model Eq. 1 and 2 $\forall r \forall t$ (3) and (4) targets the minimization of costs associated to inventory levels, set-up and operations, considering products with alternative bills of materials:

$$\text{Min} \sum_{i=1}^{CARD(I)} \sum_{t=1}^{CARD(T)} \left(HC_{i,t}\, y_{i,t}\right) + \sum_{k=1}^{CARD(K)} \sum_{t=1}^{CARD(T)} \left(SC_{kt}\, \beta_{k,t} + OC_{k,t}\, z_{k,t}\right) \tag{1}$$

Subject to:

$$y_{i,t} - y_{i,t-1} + \sum_{k=1}^{CARD(K)} M_{i,k}\, z_{k,t} + RPL_{it} + \sum_{k=1}^{CARD(K)} M_{i,k}\, z_{k,t} - LT_k - D_{it} \qquad \forall i \forall t \tag{2}$$

$$\sum_{k=1}^{CARD(K)} \left(RS_{k,r}\, z_{k,t} + RP_{k,r}\, \beta_{k,t}\right) \le KA\,P_{r,t} \qquad \forall r \forall t \tag{3}$$

$$y_{i,t} \ge 0,\ z_{i,t} \ge 0,\ \beta_{k,t} \in \{1, 0\} \tag{4}$$

Equation 1 shows the costs function to be minimized, Eq. 2 is the inventory continuity constraint associated to the alternative bill of materials: matrices M_{ik} and N_{ik}. Finally, Eq. 3 is the constraint associated to the

productive resource's capacity. The problem's complexity can be claimed to be NP-Hard since simplified to a single level, the problem becomes a Capacitated Lot-Sizing Scheduling Problem (CLSP) which is NP-Hard (Florian *et al*., 1980). Therefore, if the CLSP problem is NP-Hard, then GMOP will be NP-Hard too.

RESULTS AND DISCUSSION

Proposed heuristic: Let SS_{kk} be the matrix that represents the relations of input and output among strokes and that is determined by Eq. 5:

$$SS_{kk} - N_{ik}^{-1} \times M_{ik} \tag{5}$$

Let ç = (N, A) too be the graph that draws the same relations and that is built from SS_{kk} with the additional property of modeling in a clear way the and/or relationship between sets of strokes and the products they generate (Line 2). The heuristic solves the problem of determining which products will be generated from which strokes and to what proportion. To do this it is necessary to know of how many forms a single product can be elaborated and estimate every option's accumulated cost (Lines 4 and 5). The way this calculation is performed will be explained later. At the same time the product list must be sorted so that in the case that a product contains in his 'son's list another product of the same level, the son gets executed first (Line 7). To determine in which way the product's requirements will be served which would define the way in which the strokes would be called, each product's MRP is calculated using a lot-sizing rule that can be defined as an input to the system (Line 10).

Algorithm 1: Henuristic's global algorithm

```
GMOPH Henuristic's (OCk, SCk, Hci, Nik, Mik, Dit, Rit, IInici, LTk)
    SSkk = matrix product between Nik y Mik
    Let G = (N, A) be the graph representing input and output
    relationship between product and strocks
    accum Ock←accumOC (G (N, A), OCk)//calculation of
    accumulated costs
    funcation Recursive (productsList, strokeMatrix)
        sort Bill of Materials (productsList)
        For (int i←0; i<productsList size(); i++)
                For (int t←0; t<periodLebgth; t++)
                    productsList (i). mps←get MPSP of (i, Dit,
                    Rit, IInici)
                    If (product list (i). map←(t)>0) Then
                        stroke←selectStroke (i, t, productsList
                        (i).map (t), Ltk)
                        storkeMatrix (stroke.key, stroke.run
                        Time)←
                        storkeMatrix (stroke.key, stroke.run
                        time)+
                        storke repeat number
                        ProductsList(i).mps(t)←0
                    End if
                End for
                For (int i←0; i<productsList.size(); i++)
                    If (productsList (i).has NoRequirements()
                    and
                    productsList.size()>1) Then
                        productsList.remove (i)
                        i←-1;
                    End If
                    Elsif (productsList.(i). has
                    NoRequirements() and
                    productsList size() = 1)
                        return stroke matrix
                    End Elsif
                End for
                    return Recursive (productsList, stroke
                    Matrix)
                End for
                return strokeMatrix
            End Recursive
        End GMOPHeuristic
```

After knowing the best stroke to generate product i in period t and having decided the moment in which will be executed and how many times, the stroke's product requirements are added to the list and the current stroke's requirement are returned to zero (Lines 12-14).

Algorithm 2: Select the best stroke

```
selectStroke(i, t, productsList(i).mps(t), LTk
    mps←productsList(i). mps(t)
    For (int k, k<numStrokes; k←k+1)
        Set C
        c←Iterator(C)
        If (k←i) Then
            If (t-Ltk>0) Then
                C.add (k)
                CTc←accumSCc+(mps×accumOCc)
                c.next ()
            End If
        End If
    End For
    min Cost←argMin(CTc)
    stroke: = c*¤C: CTc* = min
    return stroke
    End selectStroke
```

Accumulated costs: The way the heuristic evaluates each stroke has to do with its most relevant attributes, Specifically Setup (SC) and Operation Costs (OC) and Lead-Time (LT). Figure 1a product-stroke graph can be observed, this allows to explain the logic of the heuristic.

Each stroke's accumulated OC and SC would be calculated according to its own OC and SC plus the accumulated ones of those strokes "under" it. At each level, the production option to be chosen would be that with the least value of total accumulated costs (accumulated SC+ accumulated OC = accumulated TC). For product 'A' in the example, there are three paths that may be taken: trough Stroke 1, 4 or 5. The

heuristic will weigh the accumulated costs of every valid stroke option (feasible by LT) and on each level will pick the cheapest one as shown in lines 6-9. The calculation of accumulated costs is defined by Eq. 6:

$$\text{accumOC}_k = \text{OC}_k + \sum_{h=0}^{H} \text{accumOC}_h + \frac{\sum_{g=0}^{c} \text{accumOC}_g}{G} \quad (6)$$

In words, accumOC of stroke k is equal to k's own OC plus the sum of the accumOC of its descendants in and condition plus the average of the accumOC of its children in or condition. Table 2 is observed a Description of the elements according to the accumulated costs per stroke.

Framework of rules: One of the main strengths of the proposed heuristic is that it behaves as a framework, allowing the implementation of any sort of lot-sizing rule that the user wants to apply. These will affect the execution time and the final costs of the model. In the numerical shown to follow, we will use the L×L rule, though we might have used as well any of the most common as: FOQ (Fixed Order Quantity) EOQ (Economic Order Quantity), Siver-Meals, Wagner-Whitin, among others.

Numerical illustration: Next, we present a simple example where the heuristic was applied, shown each step taken by the algorithm and the results obtained. The example consists of 6 products (card (i) = 6), generated by 9 strokes (card (k) = 9) with the input/output matrices M_{ik} and N_{ik}.

Input data: In the first place, the product's structure is passed to the heuristic as a parameter as it is done to the GMOP Model, through matrices M_{ik} and N_{ik} which will indicate the precedence and succession relationship among between products and strokes.

In order to ease the reading, products have been enumerated with the letters from 'A' to 'Z' and strokes have been written with and ''s on front. The association of Product-Stroke relationships is represented in Fig. 1 and matrices (Eq. 7 and 8) Table 2 and 3:

$$N_{ik} = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (7)$$

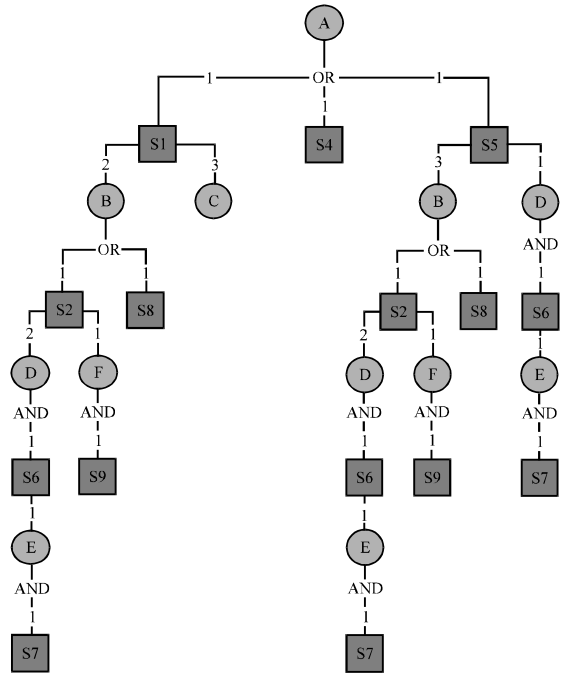

Fig. 1: Product-stroke graph of a supply chain whose final product is A

Table 2: Lead-time, setup costs and operation costs for the strokes in the example

Stroke	LT_k	SC_k	OC_k
k = 1, S1	1	2000	2.0
k = 2, S2	2	1800	0.5
k = 3, S3	1	1500	0.5
k = 4, S4	5	18000	0.5
k = 5, S5	1	7300	2.0
k = 6, S6	2	4000	1.0
k = 7, S7	1	1800	0.5
k = 8, S8	3	3000	0.5
k = 9, S9	1	2500	0.5

Table 3: Initial inventory, holding costs and stock-out costs of each product in the example

Product/material	II_{pici}	HC_i	SOC_i
i = 1, A	200	50	500
i = 2, B	150	60	600
i = 3, C	30	56	400
i = 4, D	20	36	950
i = 5, E	90	21	210
i = 6, F	10	35	356

$$M_{ik} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 \\ 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (8)$$

In second place, it is necessary to introduce necessary information about each stroke its Lead-Time (LT) Operation Cost (OC) and Setup Cost (SC) in Table 2

Table 4: Demand forecast for products in the example from day 1-10

	$D_{i,t}$									
Product	1	2	3	4	5	6	7	8	9	10
i = 1, A	0	0	0	0	800	300	300	300	100	200
i = 2, B	0	0	100	0	0	0	230	100	347	900
i = 3, C	0	0	0	0	0	0	0	0	0	0
i = 4, D	0	0	0	0	0	0	0	0	0	0
i = 5, E	0	0	0	0	0	0	0	0	0	0
i = 6, F	0	0	0	0	0	0	0	0	0	0

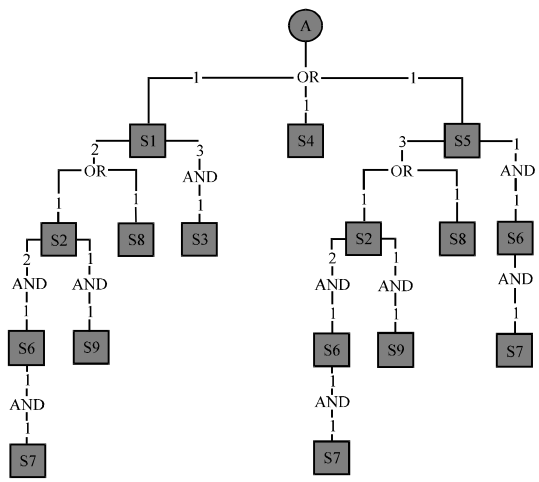

Fig. 2: Stroke-stroke graph for a supply chain whose final product is A

and 3. In third place, information about products and materials on the bill of materials is introduced: their demand forecast, initial inventory, holding costs and opportunity cost due to stock-outs (Table 3 and 4).

Data processing: The interaction between stroke and products is eliminated in matrix SS_{kk} and a better sight of stroke to stroke requirements is obtained (Fig. 2). Important data as 'number of option's or ways of generating a product (or stroke in this case) can be reached from matrix N_{ik} like this:

$$SS_{kk'} - N_{ik}^{-1} \times M_{ik} -$$

$$\begin{bmatrix} & S1 & S2 & S3 & S4 & S5 & S6 & S7 & S8 & S9 \\ S1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ S2 & 2 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 \\ S3 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ S4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ S5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ S6 & 0 & 2 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ S7 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ S8 & 2 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 \\ S9 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (9)$$

$$\text{numOpc}_i = \sum_{k=0}^{\text{card}(k)} N_{ik} \quad (10)$$

Arcs in the graph hold information about how many times a lower level stroke needs to be executed in order to its direct parent to be runnable. For this reason in ζ = (N, A), N are strokes and A are requirement relationships between a stroke and its descendants, found in S_{kk}. In matrix S_{kk}' parents are columns and sons will be the rows, this way, stroke S2 requires two runs of S6 and one of S9.

Calculation of accumulated costs per stroke: In order to select a stroke to generate product i we first need to calculate operation and setup accumulated costs for each stroke that can generate it. On the first level, product A, may be generated by strokes S1, S4 and S5. As shown with Eq. 6 accumulated costs are calculated recursively. An shown is made using S1. The size of the set of S1's 'son's in condition of and (H) and or (G):

$$H_{s1} = \{S3\} = 1, \text{indax: h}$$
$$G_{s1} = \{S2, S8\} = 2, \text{indax: g}$$

$$\text{accumSC}_{s1} = SC_{s1} + \sum_{h=0}^{H_{S1}} \text{accumSC}_h + \frac{\sum_{g=u}^{G_{S1}} \text{accumSC}_g}{G_{S1}}$$

$$\text{accumSC}_{s1} = 2000 + \text{accumSC}_{s3} + \frac{\text{accumSC}_{s2} + \text{accum}_{s8}}{2}$$

The 'son's in condition of AND and OR of strokes S3, S2 and S8 are:

$$H_{S3} = \{\ \} = 0, G_{S3} = \{\ \} = 0$$
$$H_{S2} = \{S_6, S_9\} = 1, G_{S2} = \{\ \} = 0$$
$$G_{S8} = \{\ \} = 1, G_{S8} = \{\ \} = 0$$

Table 5: Accumulated setup and operation costs for S1, S4 and S5

Stroke	accum SS_k	accum OC_k
k = 1, S1	10050	8.00
k = 4, S4	18000	0.50
k = 5, S5	19650	10.25

Table 6: MPS for final product A (i = 1) using lot by lot rule

Product	t = 1	t = 2	t = 3	t = 4	t = 5	t = 6	t = 7	t = 8	t = 9	t = 10
D_{1t}	0	0	0	0	800	300	300	300	100	200
Inv_{1t}	200	200	200	200	0	0	0	0	0	0
L×L	0	0	0	0	600	300	300	300	100	200
R_{1t}	0	0	0	0	0	0	0	0	0	0
OP_{1t}	0	0	0	0	600	300	300	300	100	200

Ii_{nic1}: 200

Table 7: Costs of setup, operation and its sum to decide which the best stroke to generate product A is

Variables	Product	Units			Setup cost			Operation cost			Total cost		
T	OP_{1t}	S1	S4	S5	S1	S4	S5	S1	S4	S5	S1	S4	S5
1	0	0	300	0	0	18000	0	0	150	0	0	18150	0
2	0	0	300	0	0	18000	0	0	150	0	0	18150	0
3	0	0	300	0	0	18000	0	0	150	0	0	18150	0
4	0	600	100	600	10050	18000	19650	4800	50	6150	14850	18050	25800
5	600	300	200	300	10050	18000	19650	2400	100	3075	12450	18100	22725
6	300	300	0	300	10050	0	19650	2400	0	3075	12450	0	22725
7	300	300	0	300	10050	0	19650	2400	0	3075	12450	0	22725
8	300	100	0	100	10050	0	19650	800	0	1025	10850	0	20675
9	100	200	0	200	10050	0	19650	1600	0	2050	11650	0	21700
10	200	0	0	0	0	0	0	0	0	0	0	0	0
											74700	90600	136350

$$\text{accumSC}_{S1} = 2000 + S_{C3} + \frac{(SC_{S2} + \sum_{h=0}^{H_{S2}} \text{accumSC}_h) + S_{C8}}{2}$$

$$\text{accumSC}_{S1} = 2000 + 1500 + \frac{1800 + (\text{accumSC}_{S6} + \text{accumSC}_{S9}) + 3000}{2}$$

$$\text{accumSC}_{S1} = 3500 + \frac{4800 + (\text{accumSC}_{S6} + \text{accumSC}_{S9})}{2}$$

The 'son's in condition of AND and OR, of strokes S6 and S9 are:

$$H_{S6} = \{S_7\} = 1, G_{S6} = \{\ \} = 0$$
$$H_{S9} = \{\ \} = 1, G_{S9} = \{\ \} = 0$$

$$\text{accumSC}_{S1} = 3500 + \frac{4800 + SC_{S6} + \sum_{h=0}^{H_{S6}} \text{accumSC}_h + SC_{S9}}{2}$$

$$\text{accumSC}_{S1} = 3500 + \frac{4800 + 4000 + \text{accumSC}_{S7} + 2500}{2}$$

Stroke S7 has no more children, we have reached the end of the tree:

$$\text{accumSC}_{S1} = 3500 + \frac{11300 + SC_{S7}}{2} = 3500 + \frac{11300 + 1800}{2}$$

$$\text{accumSC}_{S1} = 3500 + \frac{13100}{2} = 10050$$

Selection of the best ssroke: After repeating the process in previous section, accumulated operation and setup are costs for S1, S4 and S5 are obtained (Table 5). Now needs to be decided which one to choose.

Construction of the MPS: Manufacture costs of products are directly proportional to the amount that needs to be produced. Next step, for the selection of the best stroke is determining the Master Production Plan (MPS) here is where the lot-sizing rule is applied to know how much to produce on each period.The Order for Production (OP) for product A (I = 1) using L×L (Lot by Lot) rule can be observed on Table 6.

As may also be noticed, L×L implies that exactly what is required will be ordered, this way at the end of the period product A's inventory be null.

Table 8: Final stroke execution

S_1	t = 1	t = 2	t = 3	t = 4	t = 5	t = 6	t = 7	t = 8	t = 9	t = 10
S_1	0.0	0.0	0.0	600.0	300.0	300.0	300.0	100.0	200.0	0.0
S_2	0.0	0.0	0.0	0.0	80.0	100.0	347.0	900.0	0.0	0.0
S_3	0.0	0.0	1770.0	900.0	900.0	900.0	300.0	600.0	0.0	0.0
S_4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
S_5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
S_6	0.0	0.0	140.0	200.0	694.0	1800.0	0.0	0.0	0.0	0.0
S_7	0.0	50.0	200.0	694.0	1800.0	0.0	0.0	0.0	0.0	0.0
S_8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
S_9	0.0	0.0	0.0	70.0	100.0	347.0	900.0	0.0	0.0	0.0

Costs comparison: Next, total costs for each stroke are compared, verifying the feasibility of using each in the given period. This means that if the lead time chosen for every stroke allows time to fulfill the order in time. The 7 can be observed that $LT_4 = 5$ and therefore the first 100 units order of A hasn't time to be executed by S4 on the required time. In the end, the result throws that for product A, the cheapest feasible stroke to use is S1.

Once S1 is picked, the costs are transferred to the costs Table, inventory data is stored and the process continues to the following level. It is important to not forget that this is a recursive process, repeated for each material required to generate the final products. In this context final products are those that have an external forecasted demand which in this example would be A and B. It is also necessary to take into account that when proceeding to the next level, demand for B will have been added the internal demand caused by A's production for the best stroke for A, S1, counts B among its materials.

Heuristic's results: After following the previous steps, using L×L lot-sizing rule, total stroke execution in the end of the planning horizon results as in Table 7. Total production cost is $538013,00 related to the total stroke production costs along the planning horizon, specified on Table 8.

CONCLUSION

In this research was presented a heuristic model to solve the material and operations planning in supply chains with alternative product structure problem. The optimal solution to this is obtained through GMOP Model which targets the minimization of costs associated to inventory levels, setup and operations, considering products with alternative bill of materials. This heuristic resolves the problem of determining which products will be generated from which strokes and at which proportions for this it is necessary to know of how many ways a same product can be generated and estimate which is the accumulated cost of each option. To define in which way the requirements will be attended, the MPS of every product is calculated using a lot-sizing rule that can be defined as an input data of the system. After determining the best stroke to generate a product in each period the moment for it to run is defined until all the products in each period is planned. At last, a numerical illustration of the heuristic was shown, explaining the steps taken by the algorithm and the results obtained computationally.

REFERENCES

Billington, P.J., J.D. McClain and L.J. Thomas, 1983. Mathematical programming approaches to capacity-constrained MRP systems: Review, formulation and problem reduction. Manage. Sci., 29: 1126-1141.

Bomberger, E.E., 1966. A dynamic programming approach to a lot size scheduling problem. Manage. Sci., 12: 778-784.

Chryssolouris, G., K. Efthymiou, N. Papakostas, D. Mourtzis and A. Pagoropoulos, 2013. Flexibility and complexity: Is it a trade-off?. Int. J. Prod. Res., 51: 6788-6802.

Florian, M., J.K. Lenstra and K.A.H.G. Rinnooy, 1980. Deterministic production planning: Algorithms and complexity. Manage. Sci., 26: 669-679.

Ivanov, D., B. Sokolov and J. Kaeschel, 2010. A multi-structural framework for adaptive supply chain planning and operations control with structure dynamics considerations. Eur. J. Oper. Res., 200: 409-420.

Moreira, E., A.C.P. Junior, O. Pacifico and A.P.S. Junior, 2014. Contribution of production planning and control to business competitiveness: A study in a furniture company. Mag. Spaces Venezuela, 35: 1-5.

Orlicky, J., 1975. Material Requirements Planning. McGraw-Hill, New York, USA.,.

Sabater, G.J.P., J. Maheut and M.J.A. Garcia, 2013. A new formulation technique to model materials and operations planning: The Generic Materials and Operations Planning (GMOP) problem. Eur. J. Ind. Eng., 7: 119-147.

Sana, S.S., J.A. Chedid and K.S. Navarro, 2014. A three layer supply chain model with multiple suppliers, manufacturers and retailers for multiple items. Appl. Math. Comput., 229: 139-150.

Silver, E.A., D.F. Pyke and R. Peterson, 1998. Inventory Management and Production Planning and Scheduling. 3rd Edn., Wiley, New York, ISBN: 9780471119470, Pages: 754.

Stadtler, H. and C. Kilger, 2005. Supply Chain Management and Advanced Planning. 3rd Edn., Springer, Berlin, Germany,.

Stadtler, H., B. Fleischmann, M. Grunow, H. Meyr and C. Suirie, 2012. Advanced Planning in Supply Chains. 1st Edn., Springer, Berlin, Germany,.