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

1Jairo R. Coronado-Hernandez, 2Daniela Simancas-Mateus, 3Karolina Avila-Martinez and 4Jose P. Garcia-Sabater  
1Department of Industrial Engineering, Universidad de la Costa, Cra. 55 #58-66, Barranquilla, Colombia  
2Department of Informatics, Technische Universität München, Boltzmannstraße 3, 85748 Garching bei München, Germany  
3Department of Maritime Management, Escuela Naval de Cadetes “Almirante Padilla”, Isla de Manzanillo, Cartagena de Indias, Colombia  
4Department 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 (Sera et al., 2014).

Planning is about answering an apparently simple question what is the lot-sizing 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...
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, proposed dynamic programming. 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, dedication to attend global suppliers (Chrysosouris et al., 2013). As an outcome of this business model, product structures have national and international participation 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 manufacturing 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.

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 ∀∀ ∀∀ (3) and (4) targets the minimization of costs associated to inventory levels, set-up and operations, considering products with alternative bills of materials:

Min \[ \sum_{i=1}^{\text{CARD}(T)} \sum_{t=1}^{\text{CARD}(T)} \left( \text{HC}_{i,t}, y_{i,t} \right) + \sum_{k=1}^{\text{CARD}(k)} \left( \text{SC}_{k,t}, \beta_{k,t} + \text{OC}_{k,t}, z_{k,t} \right) \] (1)

Subject to:

\[ y_{i,t} - y_{i+1,t} + \sum_{k=1}^{\text{CARD}(k)} M_{i,k} z_{k,t} + \text{RPL}_{i,t} + \sum_{k=1}^{\text{CARD}(k)} M_{i,k} z_{k,t} - \text{LT}_{k,t} - D_{i,t} \quad \forall i \forall t \] (2)

\[ \sum_{i=1}^{\text{CARD}(k)} \left( \text{RS}_{i,k}, z_{k+1} + \text{RP}_{k,i}, \beta_{k,i} \right) \leq \text{KAP}_{i,t} \quad \forall i \forall t \] (3)

\[ y_{i,t} \geq 0, z_{k,t} \geq 0, \beta_{k,i} \in [1, 0] \] (4)

Equation 1 shows the cost function to be minimized, Eq. 2 is the inventory continuity constraint associated to the alternative bill of materials: matrices M _k_ and N _k_. 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 $SS_a$ be the matrix that represents the relations of input and output among strokes and that is determined by Eq. 5:

$$SS_a = N^{-1} \times M_a$$  \hspace{1cm} (5)

Let $\zeta = (N, A)$ too be the graph that draws the same relations and that is built from $SS_a$ 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’ 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: Heuristic’s global algorithm

GMOPH Heuristic’s (OC, SC, He, Na, Ma, Da, Ra, IInc, LT)

Let $G = (N, A)$ be the graph representing input and output relationship between product and strokes

Let $G = (N, A)$ be the graph representing input and output relationship between product and strokes

$SS_a$ - matrix product between $N_a \times M_a$

accum OC, accumOC (G (N, A), OC)/calculation of accumulated costs

function Recursive (productsList, strokeMatrix)

sort Bill of Materials (productsList)

For (int i=0; i<productsList.size(); i++)

For (int i=0; i<periodLength; i++)

if (product list (i), map (0)) Then

stroke = selectStroke (0, t, productsList (i), map (0), L4)

strokeMatrix (strokekey, stroke.numTime)

strokeMatrix (strokekey, stroke.numTime)

stroke repeat number

Algorithm 2: Select the best stroke

selectStroke(i, t, productList(i), mps(t), LT, mps=productsList(i), mps(t))

For (int k, k<maxStrokes, k=k+1)

Set C = Iterator (C)

If (k=i) Then

If (t-LT > 0) Then

C.add (k)

C.add (LT)

product = min (mps+accumOC, c, next (t))

End If

End For

min Cost = argMin (C)

stroke = c \times C, LT = 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: through 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:

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

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: EOQ (Fixed Order Quantity) EJOQ (Economic Order Quantity), Silver-Meal, 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_k and N_k.

**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_k and N_k 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:

$$N_k = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$M_k = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 \\ 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

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.

![Fig. 1: Product-stroke graph of a supply chain whose final product is A](image-url)

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

<table>
<thead>
<tr>
<th>Stroke</th>
<th>LT</th>
<th>SC</th>
<th>OC</th>
</tr>
</thead>
<tbody>
<tr>
<td>k=1, S1</td>
<td>1</td>
<td>2000</td>
<td>2.0</td>
</tr>
<tr>
<td>k=2, S2</td>
<td>2</td>
<td>1800</td>
<td>0.5</td>
</tr>
<tr>
<td>k=3, S3</td>
<td>1</td>
<td>1500</td>
<td>0.5</td>
</tr>
<tr>
<td>k=4, S4</td>
<td>5</td>
<td>18000</td>
<td>0.5</td>
</tr>
<tr>
<td>k=5, S5</td>
<td>1</td>
<td>7300</td>
<td>2.0</td>
</tr>
<tr>
<td>k=6, S6</td>
<td>2</td>
<td>4000</td>
<td>1.0</td>
</tr>
<tr>
<td>k=7, S7</td>
<td>1</td>
<td>1800</td>
<td>0.5</td>
</tr>
<tr>
<td>k=8, S8</td>
<td>3</td>
<td>3000</td>
<td>0.5</td>
</tr>
<tr>
<td>k=9, S9</td>
<td>1</td>
<td>2500</td>
<td>0.5</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Product/material</th>
<th>LT</th>
<th>HC</th>
<th>SOC</th>
</tr>
</thead>
<tbody>
<tr>
<td>i=1, A</td>
<td>200</td>
<td>50</td>
<td>500</td>
</tr>
<tr>
<td>i=2, B</td>
<td>150</td>
<td>60</td>
<td>600</td>
</tr>
<tr>
<td>i=3, C</td>
<td>30</td>
<td>55</td>
<td>400</td>
</tr>
<tr>
<td>i=4, D</td>
<td>20</td>
<td>36</td>
<td>950</td>
</tr>
<tr>
<td>i=5, E</td>
<td>80</td>
<td>21</td>
<td>210</td>
</tr>
<tr>
<td>i=6, F</td>
<td>10</td>
<td>35</td>
<td>350</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Product</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>i = 1, A</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>800</td>
<td>300</td>
<td>300</td>
<td>300</td>
<td>100</td>
<td>200</td>
</tr>
<tr>
<td>i = 2, B</td>
<td>0</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>i = 3, C</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>i = 4, D</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>i = 5, E</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>i = 6, F</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

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 $SS_{ik}$ 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:

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

$$H_{si} = [S3] = 1, \text{ index } h$$

$$G_{si} = [S2, S8] = 2, \text{ index } g$$

$$\text{AccumSC}_{si} = SC_{si} + \sum_{k=1}^{H_{si}} \text{AccumSC}_{sk} + \sum_{i=1}^{G_{si}} \frac{\text{AccumSC}_{si} + \text{AccumSC}_{di}}{2}$$

$$\text{AccumSC}_{si} = 2000 + \text{AccumSC}_{si} + \frac{\text{AccumSC}_{si} + \text{AccumSC}_{si}}{2}$$

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

$$H_{s3} = [\{\} - 0, G_{s3} = [\{\} - 0$$

$$H_{s2} = [S_3, S_8] - 1, G_{s2} = [\{\} - 0$$

$$G_{s2} = [\{\} = 1, G_{s2} = [\{\} = 0$$

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 $\zeta = (N, A)$, $N$ are strokes and $A$ are requirement relationships between a stroke and its descendants, found in $S_{nk}$. In matrix $S_{nk}$ 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):

$$\text{AccumSC}_{s1} = 2000 + \text{AccumSC}_{s1} + \frac{\text{AccumSC}_{s1} + \text{AccumSC}_{s1}}{2}$$

The 'son's in condition of AND and OR of strokes S3, S2 and S8 are:
Table 5: Accumulated setup and operation costs for S1, S4 and S5

<table>
<thead>
<tr>
<th>Stroke</th>
<th>accum S5</th>
<th>accum OC</th>
</tr>
</thead>
<tbody>
<tr>
<td>k = 1, S1</td>
<td>10050</td>
<td>8.00</td>
</tr>
<tr>
<td>k = 4, S4</td>
<td>18000</td>
<td>0.50</td>
</tr>
<tr>
<td>k = 5, S5</td>
<td>19050</td>
<td>10.25</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Product</th>
<th>t = 1</th>
<th>t = 2</th>
<th>t = 3</th>
<th>t = 4</th>
<th>t = 5</th>
<th>t = 6</th>
<th>t = 7</th>
<th>t = 8</th>
<th>t = 9</th>
<th>t = 10</th>
</tr>
</thead>
<tbody>
<tr>
<td>D1050</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>800</td>
<td>300</td>
<td>300</td>
<td>300</td>
<td>100</td>
<td>200</td>
<td></td>
</tr>
<tr>
<td>Inv0</td>
<td>200</td>
<td>200</td>
<td>200</td>
<td>200</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>L×L</td>
<td>0</td>
<td>0</td>
<td>600</td>
<td>300</td>
<td>300</td>
<td>300</td>
<td>300</td>
<td>100</td>
<td>200</td>
<td></td>
</tr>
<tr>
<td>R1500</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Op1000</td>
<td>0</td>
<td>0</td>
<td>600</td>
<td>300</td>
<td>300</td>
<td>300</td>
<td>300</td>
<td>100</td>
<td>200</td>
<td></td>
</tr>
<tr>
<td>Inv1000</td>
<td>200</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Variables</th>
<th>Product</th>
<th>Units</th>
<th>Setup cost</th>
<th>Operation cost</th>
<th>Total cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>Qp0</td>
<td>S1</td>
<td>S4</td>
<td>S5</td>
<td>S1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>300</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>300</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
<td>300</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>600</td>
<td>100</td>
<td>600</td>
<td>10050</td>
</tr>
<tr>
<td>5</td>
<td>300</td>
<td>300</td>
<td>0</td>
<td>300</td>
<td>10050</td>
</tr>
<tr>
<td>6</td>
<td>300</td>
<td>300</td>
<td>0</td>
<td>300</td>
<td>10050</td>
</tr>
<tr>
<td>7</td>
<td>300</td>
<td>300</td>
<td>0</td>
<td>300</td>
<td>10050</td>
</tr>
<tr>
<td>8</td>
<td>300</td>
<td>100</td>
<td>0</td>
<td>100</td>
<td>10050</td>
</tr>
<tr>
<td>9</td>
<td>100</td>
<td>200</td>
<td>0</td>
<td>200</td>
<td>10050</td>
</tr>
<tr>
<td>10</td>
<td>200</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

\[
\text{accumSC}_{2i} = 2000 + S_{c3} + \frac{(SC_{s2} + \sum_{k=1}^{5} \text{accumSC}_k)}{2} + S_{c3}
\]

\[
\text{accumSC}_{3i} = \frac{2000 + 1500 + 1800 + (\text{accumSC}_{s4} + \text{accumSC}_{s5})}{2}
\]

\[
\text{accumSC}_{4i} = 3500 + \frac{4800 + \text{SC}_{s6} + \sum_{k=1}^{5} \text{accumSC}_k + SC_{s3}}{2}
\]

\[
\text{Stroke S7 has no more children, we have reached the end of the tree:}
\]

\[
\text{accumSC}_{5i} = \frac{3500 + \frac{11300 + \text{SC}_{s7}}{2}}{2} = \frac{3500 + \frac{11300 + 1800}{2}}{2}
\]

\[
\text{accumSC}_{6i} = \frac{3500 + 13100}{2} = 10050
\]

**Selection of the best stroke:** After repeating the process in previous section, accumulated operation and setup 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.
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 $LT_7 = 5$ 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 $L_x$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 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
