

# **An Algorithm Based on Theory of Constraints and Branch and Bound for Solving Integrated Product-Mix-Outsourcing Problem**

**Esmaeil Mehdizadeh<sup>a,\*</sup>, Saeed Jalili<sup>a</sup>**

<sup>a</sup> *Faculty of Industrial & Mechanical Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran*

Received 09 October 2016; Revised 22 June 2018; Accepted 25 June 2018

## **Abstract**

One of the most important decision making problems in many production systems is identification and determination of products and their quantities according to available resources. This problem is called product-mix. However, in the real-world situations, for existing constrained resources, many companies try to provide some products from external resources to achieve more profits. In this paper, an integrated product-mix-outsourcing problem (IPMO) is considered to answer how many products should be produced inside of the system or purchased from external resources. For this purpose, an algorithm based on Theory of Constraints (TOC) and Branch and Bound (B&B) algorithm is proposed. For investigation of the proposed algorithm, a numerical example is presented. The obtained results show the optimal result by the new algorithm is as same as the results of integer linear programming.

**Keywords:** Product-mix; Outsourcing; T theory of constraints; Branch and bound algorithm.

## **1. Introduction**

Identification and determination of products and their quantities according to available resources are called product-mix (PM) problem in manufacturing plants. There are many methods for solving this problem. One of these methods is Theory of Constraints which was applied for the first time by Goldratt in 1987 (Goldratt, 1988).

This method is easy to use and understand. So, many researchers have tried to investigate or develop it for different conditions. For example, Plenert (1993) (Plenert, 1993) showed that when there was more than one bottleneck, it might be difficult to identify which one was the dominant bottleneck. Also, Lee and Plenert (1993) (Lee & Plenert, 1993) demonstrated that when evaluating new product introductions, the TOC procedure was inefficient and produced a non-optimal product-mix. For solving these problems. Fredendall and Lea (1997) (Fredendall & Lea, 1997) proposed a revised theory of constraints (RTOC) for product-mix problems when the TOC product-mix heuristic failed previously. RTOC provided the same results as ILP in most of the cases. Aryanezhad and Rashidi Komijan (2004) (Aryanezhad & Komijan\*, 2004) proposed an "improved algorithm", which could reach to an optimum solution of product-mix under TOC. The efficiency of their "improved algorithm" was compared with the RTOC algorithm of Fredendall and Lea (Fredendall & Lea, 1997) through an example. A group of applications of TOC to product-mix issues includes those using meta-heuristic approaches. Among these, genetic algorithms (Onwubolu & Mutingi, 2001),

simulated annealing (Chaharsooghi & Jafari, 2007), Tabu-search (Onwublu, 2001), Immune algorithm (Wang, Sun, Si, & Yang, 2009) and hybrids of such methods such as tabu-simulated annealing (Mishara, Prakash, Tiwari, Shankar, & Chan, 2005) have been applied to analyzing TOC. Another category method for solving the product-mix problem based on TOC algorithm belongs to uncertainty conditions. For example we can mention to Ghazinoory et al. (2010) (GHAZINOORI, Sadeghian, & Samouei, 2010), Azadegan et al. (2011) (Azadegan, Porobic, Ghazinoory, Samouei, & Kheirkhah, 2011), Ghazinoory et al. (2013) (Ghazinoory, Fattahi, & Samouei, 2013) and Hamidi et al. (2012) (Hamidi, Samouei, & Eghbali, 2012) that verified fuzzy logic in their algorithms that they were based on TOC or Revised-TOC. An assumption that all papers that have mentioned to them used it was produced by internal production system and they have not used external sources for satisfying whole demands. But in the competitive real world, managers attempt to satisfy the greater part of the market demand to achieve more sale and profits. But there are usually constrained resources in the production system that they do not allow to produce whole of demands. So, some systems attend to allocate some products to the external sources.

Mehdizadeh et al. (2014) introduced a simple and heuristic method based on TOC to solve IPMO problems in multi-bottleneck mode. At first, it was shown, by an example that available heuristic algorithms for solving the IPMO problem did not necessarily lead to optimum and even feasible solutions. Then, the proposed algorithm was

\*Corresponding author Email address: emehdi@qiau.ac.ir

compared with existing methods in the literature in single and multi-bottleneck modes, and indicated that the proposed algorithm was able to achieve appropriate solutions.

In this paper, we verify product-mix problems with considering outsourcing factor simultaneously and call this problem as IPMO. The main question in IPMO problem is how many products are supplied within the system and how many from the external sources to achieve the maximum profit (Nazari-Shirkouhi, Eivazy, Ghodsi, Rezaie, & Atashpaz-Gargari, 2010). According to Kutner (2004) (Küttner, 2004), outsourcing problem includes decision making on which product or which parts of the product must be produced inside and which ones must be produced out of borders. But there are a few types of research that investigate TOC in IPMO. A good paper in this area belongs to Coman and Ronen (2000) (Coman & Ronen, 2000) that developed IPMO problem for the first time. They formulated this problem as an integer linear programming and evaluated 4 solving methods (standard accounting, TOC algorithm, integer linear programming and revised-TOC. Their conclusions showed that TOC result was better than standard accounting but not an optimal result. Also, the revised-TOC result was as same as the result of integer linear programming. Another paper about IPMO presented by Nazari-Shirkouhi et al. (2010) (Nazari-Shirkouhi et al., 2010) that used Imperialist Competitive algorithm for this problem.

By reviewing the literature about TOC and IPMO, it will be clear that there are a few attempts for developing these areas, simultaneously (especially in multi bottleneck state). Furthermore, existing methods for solving IPMO problem based on TOC would not arrive at optimal or even justified results in case of multi bottleneck states. So, in this paper, we present a branch and bound algorithm based on TOC to solve the IPMO problem in multi bottleneck states and to find the optimum solution in reasonable time.

The remainder of this paper is structured as follows. In the next section, the classical model of IPMO problem is described. In Section 3, the proposed algorithm will be introduced and in Section 4, an example is presented and solved it with the proposed algorithm. Finally, the conclusions and future researches of this paper are given in Section 5.

## 2. Model Description

Mathematical formulation of the IPMO problem is described here along with the notations:

$P_i$ : decision variable representing the quantity of product type  $i$  which must be produced in the system.

$n$ : number of resources

$m$ : number of product-mix type

$RM_i$ : the raw material cost of product  $i$ .

$AC_j$ : the available capacity of the  $j^{th}$  source during the planning horizon

$CP_i$ : buying price of product  $i$  from the external supplier

$D_i$ : market demand for product type  $i$

$t_{ij}$ : processing time required resource  $j$  to produce a product of type  $i$

The formulation is given as (Coman & Ronen, 2000):

$$\text{Max} = \sum_{i=1}^m P_i(CP_i - RM_i) \quad (1)$$

S.to:

$$\sum_{i=1}^m P_i t_{ij} \leq AC_j \quad j = 1, \dots, n \quad (2)$$

$$P_i \leq D_i \quad i = 1, \dots, m \quad (3)$$

$$P_i \geq 0 \text{ \& integer} \quad i = 1, \dots, m \quad (4)$$

### 2.1. Description of the objective function

Regarding the issue of how to calculate the objective function, which represents the system profit, it is assumed that the demands of all products are satisfied through the internal production system and outsourcing (external supplier). Therefore, the profit of system would be obtained by reducing the summation of the costs of raw materials ( $RM_i$ ), outsourcing ( $CP_i$ ) and fixed operational cost ( $OE$ ) from selling prices ( $MP_i$ ).

$$\text{Profit} = \sum_{i=1}^m [D_i \cdot MP_i - CP_i(D_i - P_i) - P_i \cdot RM_i] - OE \quad (5)$$

In this equation,  $(D_i - P_i)$  is the amount of product  $i$  which is ordered from the external supplier. By rewriting the equation (5), equation (6) will be obtained (Coman & Ronen, 2000):

$$\text{Profit} = \sum_{i=1}^m [(MP_i - CP_i)D_i + P_i(CP_i - RM_i)] - OE \quad (6)$$

Because of the amount of  $(MP_i - CP_i)D_i$  and  $OE$  are fixed values, for gaining maximum profit we can maximize  $\sum_{i=1}^m [P_i(CP_i - RM_i)]$ .

### 2.2. Description of the constraints

Equation (2) shows the maximum total time that used in each internal production system resource is its available capacity. Furthermore, equation (3) shows maximum demands of each product.

## 3. Proposed algorithm

Proposed branch and bound algorithm is an exact method for integrated product-mix and outsourcing problems that are presented in this section.

At the first, it is necessary to define the mechanism of branching procedure.

### 3.1. Branching procedure

Branching must be performed on a defined constrained set of  $S$ . So, the definition of set  $S$  for the integrated product-mix-outsourcing problem is necessary. This set has the number of each product types that produced inside of production system and the rest assignment for the external supplier. Bottleneck sources and the maximum market

demand in an integrated product-mix-outsourcing problem are the explicit constraints of the proposed algorithm. In the first step and in node 0, all products are produced by internal production system without according to the explicit constraints. Branching procedure will be as follows:

We consider to the main bottleneck which has the highest capacity lack in node 0 and obtains the sequence of this production based on the proposed constraints theoretical method proposed by Coman and Ronen (2000) (Coman & Ronen, 2000). This sequence shows the priority of each product that can be produced inside the system. Each product which has the lowest priority in case of the main bottleneck must be assigned to external source sooner than the others and it is the first choice for cutting to create a justifiable space. There are two cuts off in this algorithm:

- 1- One unit of selected product will be assigned to the external source and the released source will be calculated. After this, if the required capacities of bottlenecks would be lower than their available capacities, a justifiable space and answer are reached.
- 2- The amount of assigned product to the external source is determined.

### 3.2. Bounding Rule

The second characteristic of the branch and bound algorithm is bounding which through cutting off some branches in upper levels which do not arrive at an optimal answer.

Bounding rule in the proposed algorithm is as follows:

1. Continue branching procedure among nodes until a node with the justifiable answer is reached. If we reach at a justifiable answer, that node will be constrained.
  2. A lower bound of each node will be obtained based on the number of products that produced in the internal production system and the amount of assignment to an external source. Whenever we arrive at a node, it's lower bound that shows the system profit is calculated.
- If a lower bound is less than the specified limit of the algorithm, that node will be constrained.

The steps of proposed the branch and bound algorithm will be as follows:

#### **Step 1: Initial adjustments**

At the first, in node 0 all products are produced by the internal production system. So, bottleneck sources have not been observed which results in a justifiable answer in node 0. In this step, the main bottleneck which has the highest lack is selected, too. Also, because of the objective function is maximization, the preliminary lower bound ( $L=0$ ) is considered for branch and bound algorithm.

#### **Step 2: choosing node for cutting off**

Based on the first depth, we choose the node to be cut off and if the chosen node has been constrained according to step 5, the next node through a backward method is chosen. If no node is found to be cut off, the algorithm will be stopped and the lower bound shows the optimal answer.

#### **Step 3: Choosing product for being assigned to the external supplier**

Based on the sequence of priorities in the main bottleneck, the product which has the lowest priority is selected for assigning to the external source. If the product's condition is clear (producing by internal production system or supplying from the external source), the next product with the lowest priority and undetermined condition is chosen.

#### **Step 4: Cutting the node**

*First cut:* We assign one unit of selected product to the external source.

*Second cut:* The number of products that produced inside of production system and the amount of assignment to the external source will be fixed to determine this cut. Based on these values, the objective function can be calculated for decision making about lower bound.

#### **Step 5: Constraining the node**

If the node has been reached to a justifiable answer we will constrain the objective function of this node because by cutting this node the amount of objective function (profit) will only be reduced and we will be far from an optimal answer. Then, we compare the objective function of the mentioned node with the lower bound of the algorithm. If the objective function of the node is more than the lower bound of the algorithm, the mentioned objective function will be chosen as a new lower bound.

If the node reaches to an unjustifiable answer we compare the objective function of that node with the lower bound of the algorithm. If the node's objective function is less than lower bound of the proposed algorithm, it is possible to cut this node because there is already a node that reaches to a better answer. So, the mentioned node will be constrained.

Refer to the 2nd step.

## **4. Numerical Example**

Assume that four products that are shown in Table 1 can be produced in a factory. This table has other information such as the name of stations; available capacity; required capacity; the difference between available and required capacity; market demands; sale prices; purchasing prices from the external supplier; raw material costs and contribution margins. It is necessary to define which product type and how many of these should be produced inside of the system and supplied from the external supplier (outsourcing).

Table1  
The information of the numerical example

Product	Stations					Market demand	Sale price (\$)	The purchase price of the external supplier(\$)	Raw material cost(\$)	CM (\$)
	1	2	3	4	5					
A	7	14	20	23	29	30	99	27	13	14
B	32	20	15	47	36	30	109	92	65	17
C	6	1	49	2	13	10	61	40	32	8
D	46	3	35	40	36	10	73	53	52	1
available capacity (minutes/week)	2400	2400	2400	2400	2400					
required capacity (minutes/week)	1690	1060	1890	2520	2440					
Difference	710	1340	510	-120	-40					

Proposed branch and bound algorithm are used for solving this problem as follows:

**Step 1: Initial adjustments**

In the root node (node1) it is assumed that all products are produced inside of this factory. So, stations 4 and 5 will be identified as bottlenecks because they do not have

enough capacity to produce all products. Also, Station 4 will be identified as the main bottleneck due to having the greatest lack. Table 2 shows the sequence of priorities in the main bottleneck. According to this station, the best priorities will be C, A, B, and D.

The initial lower bound is 0 (L=0). Supposing all products are produced by the internal production system, the upper bound of node 0 will be 4400 (U= 4400).

Table 2  
The main Bottleneck (Station 4) priorities

Product	Material cost	Purchase price	CM <sub>i</sub>	Processing time	PR <sub>id</sub>
A	13	27	14	23	0.609
B	65	92	27	47	0.575
C	32	40	8	2	4
D	52	53	1	40	0.025

In step 2, Node 1 is selected. In step 3, product D is selected to be assigned to the external supplier based on the last sequence of priorities. In step 4, a unit of product D is assigned to the external supplier. In this situation, the upper bound of created node is as equal as 4399 (U= 4399). It is also decided not to assign product D to the external supplier at all (Priority cut). In this situation, its upper bound is 4400 (U = 4400). In step 5, regarding both upper bounds of nodes are upper than the lower bound, there is no constrained node.

After this, node 2 is selected and product D is selected to be assigned to the external supplier based on the sequence of priorities. If one unit of product D is assigned to the external supplier the upper bound of created node is 4398 (U=4398). But if it is decided not to assign product D to the external supplier the upper bound of created node is 4399 (U=4399). Regarding both upper bounds of nodes are upper than the lower bound, there is no constrained node, too.

After this stage, Node 4 is selected and similar to previous stages product D is selected to be assigned to external

source based on priorities. If one unit of product D is assigned to the external supplier the upper bound of created node is 4397. If product D is not assigned to the external supplier (Priority cut) the upper bound of created node is (U=4398). According to the answer of node 6 is justifiable, this node will be constrained and the previous lower bound (L=0) changes to (L=4397).

In the next stage, Node 7 and Product B are selected because product D will not be assigned to external sources anymore. If one unit of product B is assigned to an external source, upper bound will be 4371 (U=4397)but if product B is not assigned to external source this value 4398. Regarding that, the answer is justifiable in node 8 and its objective function is less than the lower con bound (L=4397) this node will be constrained.

If we repeat this procedure, the best answer of objective function value will be 4397 and 30A, 30B, 10C, 7D will be produced by internal production system and 3D is assigned to the external supplier. The following diagram shows these details.

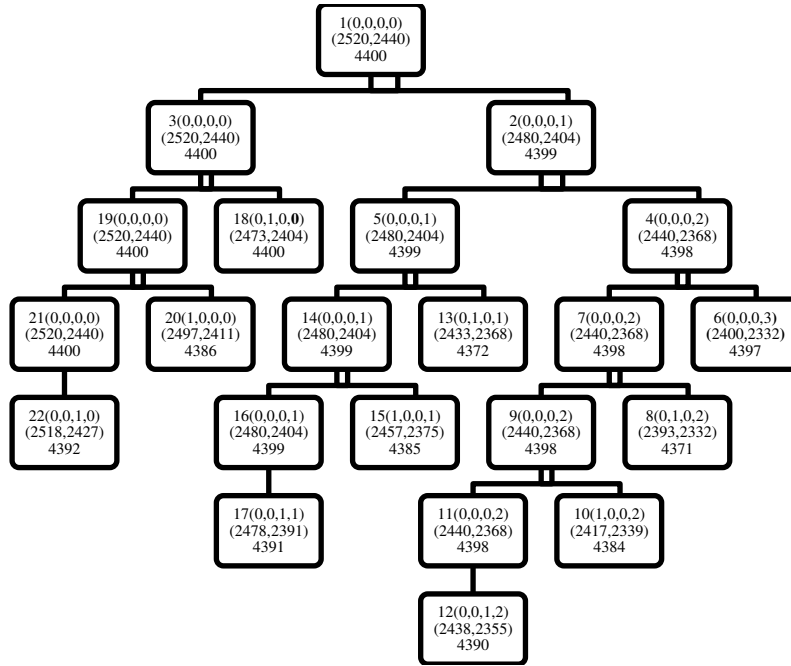


Fig. 1. Diagram of the proposed algorithm for solving the numerical example

If  $P_i$  shows the number of product  $i$ , ( $i = A, B, C, D$ ), the integer linear programming model of the above example will be as follows:

$$Max = 14 P_A + 27 P_B + 8 P_C + P_D$$

Subject to:

$$7P_A + 32P_B + 6P_C + 46 P_D \leq 2400$$

$$14P_A + 20P_B + 1P_C + 3P_D \leq 2400$$

$$20P_A + 15P_B + 49P_C + 35P_D \leq 2400$$

$$23P_A + 47P_B + 2P_C + 40 P_D \leq 2400$$

$$29P_A + 36P_B + 13P_C + 36P_D \leq 2400$$

$$P_A \leq 30$$

$$P_B \leq 30$$

$$P_C \leq 10$$

$$P_D \leq 10$$

$$P_i \geq 0 \quad \& \quad integer \quad i = A, B, C, D$$

If we solve the above model with Lingo software, the optimal result will be 30A, 30B, 10C, and 7D. It shows that these amounts of products should be supplied by the external supplier. So, 3 units of product D will be bought from the external supplier (outsourcing unit). In this situation, the objective function value will be \$4397 that is as equal as the proposed algorithm. The following figure shows the details:

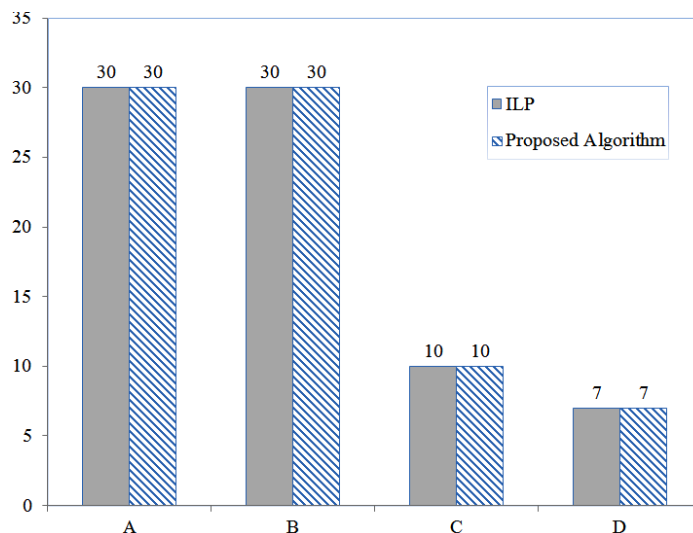


Fig. 2. Comparing ILP and proposed

### 5. Conclusion

In this research, the integrated problem of product-mix-outsourcing has been investigated, and an algorithm based on the theory of constraints and branch and bound

procedure has been developed for solving it. For comparing the efficiency of our proposed algorithm, we solve an example with both integer linear programming and our proposed algorithm. The obtained results show the maximum profit of our algorithm is as same as integer

linear programming. Studying the problem in uncertain conditions like as fuzzy environments and also verifying the other exact method such as dynamic programming and comparing their results with our algorithm seems attractive fields for future researches.

## References

- Aryanezhad, M., & Komijan\*, A. (2004). An improved algorithm for optimizing product mix under the theory of constraints. *International Journal of Production Research*, 42(20), 4221-4233.
- Azadegan, A., Porobic, L., Ghazinoory, S., Samouei, P., & Kheirkhah, A. S. (2011). Fuzzy logic in manufacturing: A review of literature and a specialized application. *International Journal of Production Economics*, 132(2), 258-270.
- Chaharsooghi, S. K., & Jafari, N. (2007). A Simulated Annealing Approach for Product Mix Decisions. *Scientia Iranica*, 143(3), 230-235.
- Coman, A., & Ronen, B. (2000). Production outsourcing: a linear programming model for the theory-of-constraints. *International Journal of Production Research*, 38(7), 1631-1639.
- Fredendall, L. D., & Lea, B. (1997). Improving the product mix heuristic in the theory of constraints. *International Journal of Production Research*, 35(6), 1535-1544.
- Ghazinoori, S. S., Sadeghian, R., & Samouei, P. (2010). Comparing Of Utilizing "Theory Of Constraints" versus Fuzzy Linear Programming In Fuzzy Product-Mix Problems.
- Ghazinoory, S., Fattahi, P., & Samouei, P. (2013). A hybrid FRTOC-SA algorithm for product mix problems with fuzzy processing time and capacity. *The International Journal of Advanced Manufacturing Technology*, 65(9-12), 1363-1370.
- Goldratt, E. M. (1988). Computerized shop floor scheduling. *The International Journal of Production Research*, 26(3), 443-455.
- Hamidi, N., Samouei, P., & Eghbali, M. (2012). Proposed fuzzy revised theory of constraints algorithm for product mix problems with fuzzy capacity, profit and processing times. *International Journal of Industrial Engineering*, 22(4), 359-368.
- Küttner, R. (2004). *Optimal planning of product mix for subcontracting companies*. Paper presented at the Proc. 4th International DAAAM Conference "Industrial Engineering.
- Lee, T. N., & Plenert, G. (1993). Optimizing theory of constraints when new product alternatives exist. *Production and Inventory Management Journal*, 34(3), 51.
- Mehdizadeh, E., Jalili, S., Eghbali, M., (2014). Presenting a Heuristic Algorithm to Integrated Product Mix-Outsourcing Problem under Theory of Constraint, *Sharif Industrial Engineering and Management Journal*, 30(1/2), 101-107( In Persian)
- Mishara, N., Prakash, Tiwari, M. K., Shankar, R., & Chan, F. T. S. (2005). Hybrid Tabu-Simulated Annealing Based Approach to Solve Multi-Constraint Product-Mix Decision Problem. *Expert Systems with Application*, 29(2), 446-454.
- Nazari-Shirkouhi, S., Eivazy, H., Ghodsi, R., Rezaie, K., & Atashpaz-Gargari, E. (2010). Solving the integrated product mix-outsourcing problem using the imperialist competitive algorithm. *Expert Systems with Applications*, 37(12), 7615-7626.
- Onwublu, G. C. (2001). Tabu Search-Based Algorithm for the TOC Product-Mix Decision. *International Journal of Production Research*, 39(10), 2065-2067.
- Onwubolu, G. C., & Mutingi, M. (2001). Optimizing the multiple constrained resources product mix problem using genetic algorithms. *International Journal of Production Research*, 39(9), 1897-1910.
- Plenert, G. (1993). Optimizing theory of constraints when multiple constrained resources exist. *European Journal of Operational Research*, 70(1), 126-133.
- Wang, J., Sun, S., Si, S., & Yang, H. (2009). Theory of constraints product mix optimisation based on immune algorithm. *International Journal of Production Research*, 47(16), 4521-4543.

**This article can be cited:** Mehdizadeh E., & Jalili S. (2019). An Algorithm Based on Theory of Constraints and Branch and Bound for Solving Integrated Product-Mix-Outsourcing Problem . *Journal of Optimization in Industrial Engineering*. 12 (1), 167- 172

[http://www.gjie.ir/article\\_543684.html](http://www.gjie.ir/article_543684.html)

DOI: 10.22094/JOIE.2018.664.1429

