COMBINATION OF GENETIC ALGORITHM WITH LAGRANGE MULTIPLIERS FOR LOT-SIZE DETERMINATION IN CAPACITY CONSTRAINED MULTI-PERIOD, MULTI-PRODUCT AND MULTI-STAGE PROBLEMS

M. Kargari, Z. Rezaee, and H. Khademi Zare

Abstract: *In this paper a meta-heuristic approach has been presented to solve lot-size determination problems in a complex multi-stage production planning problems with production capacity constraint. This type of problems has multiple products with sequential production processes which are manufactured in different periods to meet customer's demand. By determining the decision variables, machinery production capacity and customer's demand, an integer linear program with the objective function of minimization of total costs of set-up, inventory and production is achieved. In the first step, the original problem is decomposed to several sub-problems using a heuristic approach based on the limited resource Lagrange multiplier. Thus, each sub-problem can be solved using one of the easier methods. In the second step, through combining the genetic algorithm with one of the neighborhood search techniques, a new approach has been developed for the sub-problems. In the third step, to obtain a better result, resource leveling is performed for the smaller problems using a heuristic algorithm. Using this method, each product's lot-size is determined through several steps. This paper's propositions have been studied and verified through considerable empirical experiments.*

Keywords: *Production planning, Integer linear programming, Hybrid genetic algorithm, Neighborhood search method, Resource leveling, Lagrange multiplier*

1. Introduction

During the past century, production scheduling problems have evolved significantly. Material Requirement Planning (MRP) is an approach used in production planning to determine parts and materials for final products. Following that, manufacturing resource planning (MRP-II) and enterperacie resource planning (ERP) have been developed based on the hierarchical production plan. In MRP-II and ERP methods, Master Scheduling Planning (MPS) which has been obtained through the customer's predicted demand is generalized to the smallest parts of the products using bill of material (BOM). Despite the extensive application of these methods, all of them are somehow limited. The primary problem of these systems is ignoring the resource constraint [1].

In these systems if there are not enough resources for production, a part of the production activity is delayed or production plan is completed using surplus resources required at the specified time. These delays in the production plan may lead to non-practical programs; on the other hand, usage of surplus resources by each system increases the costs which are in contrast with cost reduction objective [2].

The proposed lot-size determination approach in this paper for multi-stage production planning problems with production capacity constraint, the holding, set-up and production costs has been considered. In another word, the lot-size determination and cost minimization objectives are considered simultaneously while the resource constraint is regarded. The production estimation for each part to meet customer's demand is performed in the production planning horizon. In multi-stage production planning, planning for each product is related to other products plan at the lower level.

Issue literature review indicates that production planning has direct relation with customer demand and

---
Paper first received Dec. 20, 2006 and in revised form Mar. 10, 2007.
M. Kargari, Master of Science, Tarbiat Modarres University, Department of Industrial Engineering, mehrdadkargari@mail.com
Z. Rezaee, Master of Science, Khatam University, Department of Industrial Engineering
H. Khademi Zare, Assistant Professor, Yazd University, Department of Industrial Engineering, hkhademi@yazduni.ac.ir

production capacity and this relationship has been studied extensively, but no more progress has been achieved for resources and products with uniform distribution during the production scheduling [3, 4, 5,6]. Accordingly, most of the production planning problems including resource constrained multi-stage production planning problem with set-up cost are categorized as NP-Hard problems [7].

Many solutions have been developed to solve this kind of problems using branch-and-bound approach [8]. Solution methods for this kind of problems are divided to optimization and heuristic approaches.

To solve these problems, the heuristic algorithms show more effectiveness than optimization approaches [9]. However, increasing usage of computers and needs to correct planning, finding new solutions to obtain the low cost programs seems very crucial and inevitable. Development of the genetic algorithm (GA) is one of the efforts to solve this kind of problems during 1960-1970 [10]. GA has been a successful meta-heuristic solution [11].

GA performs well in general surveys but it is not much quick in obtaining the final solution since it does not perform well in neighborhood search.

However, in most cases, this method provides a final solution. Thus, to accomplish a GA algorithm a neighborhood search algorithm must accompany it and a hybrid genetic algorithm is developed. GA, Neighborhood Search (NS) hybrid approach has been applied as an initial solution by Wang [12].

Bitran and Yanasse [13] have developed a heuristic approach to solve multi-stage single-product production planning problem.

When they added a second product to the problem, it was converted to a NP-Hard problem and when they considered a non-zero set-up time, determination of a feasible solution for the problem was converted to a NP-Hard problem [14].

An extensive issue literature review for lot-sizing has been conducted by Bahl, Kuik and Simpson [15,16,17]. Researchers have developed a multiple heuristic approach according to complexity of multi-stage production planning problem [18,19,20,21]. Katok [22] has extended a heuristic approach based on Harrison and Lewis [23].

Franca [17] developed a heuristic approach consisting four patterns based on the production transfers among the periods.

Their algorithm starts with Wagner-Whitin initial solution [24]. This approach typically develops a non-practical solution.

Following that, various approaches have been proposed to seek a practical, low cost or even a new initial solution. We use these approaches as a basis to develop a hybrid genetic algorithm.

Tempelmeir and Derstroff [25] extended an approach according to Lagrange multipliers.

They also used Wagner-Whitin [241] solution as initial solution. Then using Langrage multiplier they tried to find a practical solution.

Other researchers also have been using Lagrange method to solve production planning problems and it's efficiency in solving problems with limited resources have been proved [24,26,27].

Ozdamar and Barbarosoglu [19] developed another approach combining the Lagrange multiplier and annealing simulation.

They have compared their results with the results of the Tempelmeir and Derstroff's approach [25] but unfortunately, their method did not show any improvements.

As the issue literature review indicates, all of the articles are proposed for the single product production planning problem and no significant study has been carried out for multi-product planning problem.

The aim of this paper is to develop a heuristic approach according to the evolution trend of the existing algorithms to solve the multi-stage, multi-product and multi-period production planning problems with limited resources and set-up and installation time and cost. We have extended a more expertise HGA Algorithm.

In addition to a general search to find a near optimal solution, a local search is also used and demonstrated to generate random examples in production planning problems. Local search approaches are based on the Franca search approaches [28].

The paper has the following structure. In section 2, the mathematical model of problem and its decomposition algorithm to define sub-problems accompanying the mathematical model for each product is provided. In section 3, Franca's heuristic approach is described. Section 5 describes the surplus resource leveling and in section 6, the solution algorithm is described.

Section 7 refers to the experiment design using several examples. Finally, in section 8 is devoted to conclusion.

2. Mathematical Model of the Problem

In this section we describe the model of the multi-stage, multi-product and multi-period production planning problem (CMLSP) with production capacity constraint. In this problem we have n products which compete with each other in the limited resource allocation and thus, the production batches in each stage and period must be determined, so all products demands at various periods are satisfied.

decision parameters and variables includes:

N: Number of products $i = 1, 2, ..., N$

T: Number of periods in production planning horizon $t = 1, 2, ..., T$

K: Number of stages required for each product $j = 1, 2, ..., K$

X_{ijt} :   Production lot-size of product i in stage j and period t

I_{ijt} : On hand inventory of product i in stage j and period t

$Y_{ijt}:\begin{cases}1 & \text{if product } i \text{ is produced in stage } j \text{ at the end of period } t \\ 0 & \text{otherwise}\end{cases}$

A_{ijt} : Fixed production set-up cost for product i in stage j and period t

d_{it} : Demand for product i in period t

b_{jt} : Available resource in stage j and period t

a_{ij} : Amount of required resource for product i in stage j and period t

H_{ijt} : Unit holding cost of product i in stage j and period t

S_{ijt} : Production set-up time of product i in stage j and period t

M : Upper limit of the X_{ijt} decision variable.

C_{ijt} : Production cost of product i on machine j in period t

J_{ijt} : Amount of product i stored in stage j at the end of period t

Objective Function:

$$Min\ Z = \sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{t=1}^{T}[A_{ijt}\cdot Y_{ijt} + C_{ijt}\cdot X_{ijt} + H_{ijt}\cdot J_{ijt}] \qquad (1)$$

Constraints:

$$I_{i,m,t-1} + X_{i,m,t} - I_{i,m,t} = D_{it} \qquad (2)$$
$$i = 1,2,...,n \qquad t = 1,2,...,T$$

$$I_{i,j,t-1} + X_{i,j,t} + X_{i,j+1,t} \qquad (3)$$
$$i = 1,2,...,n \qquad t = 1,2,...,T \qquad t = 1,2,...,m-1$$

$$\sum_{i=1}^{n} a_{ijt} X_{ijt} + S_{ijt} Y_{ijt} \le b_{jt} \qquad (4)$$
$$j = 1,2,...,m \qquad t = 1,2,...,T$$

$$X_{ijt} \le M.Y_{ijt} \qquad (5)$$
$$i = 1,2,...,n \qquad t = 1,2,...,T \qquad j = 1,2,...,m$$

$$(X_{ijt}, I_{ijt}) \ge 0 \qquad (6)$$
$$i = 1,2,...,n \qquad t = 1,2,...,T \qquad j = 1,2,...,m$$

$$Y_{ijt} \in (0,1) \qquad (7)$$
$$i = 1,2,...,n \qquad t = 1,2,...,T \qquad j = 1,2,...,m$$

In this model, equation (1) represents the objective function which minimizes the total of set-up, holding and variable production costs. Equation (2) ensures the demand supply in each period. Equation (3) shows that

in a network, total of in-flows to each node (i, j, t) is equal to out-flows from that node. Equation (4) represents the production and set-up times required in each stage for each product and equation (5) ensures that set-up and installation costs are considered as the production process begins. Finally, equations (6,7) represent the type of decision variables.

2-1.Primary Decomposition Algorithm

In problems with several groups of constraints and different structures, typically, this question arises that which one of these constraints must be considered as decomposition factor.

To respond to this question, the following factors must be considered [28]:

    a) Proximity of the resultant solutions from the composition algorithm to the optimal solution.

    b) Facility to decompose the main problem to sub-problems.

    c) Facility to solve each sub-problem and compose the problem solutions.

In this model, it can be seen that only constraint (4) is in relation with all products. In the simplex problem of this constraint, we are facing a set of Lagrange multipliers (λ_{jt}) which makes the objective function to follow the constraints (2, 3, 5) and converts the multi-product hybrid problem to n individual single product problems.

Thizy [29] has shown that firstly Lagrange simplification is more precise than other simplification methods.

Secondly, Lagrange simplification of capacity constraint in comparison to other constraints results in the most stable lower limit toward the optimal solution.

Thirdly, applying the decomposition technique based on the limited resource Lagrange multiplier, for multi-stage production models simplifies the main model to n individual problems.

To decompose the main model to n individual single-product problems first we calculate the average required resources in stage j using equation (8):

$$\bar{a}_j = \frac{1}{N}\sum_{i=1}^{N} a_{ij} \quad j = 1,2,...,m \qquad (8)$$

Then, we determine the bottleneck station using equation (9):

$$q = Min\ \left\{\frac{b_1}{\bar{a}_1}, \frac{b_2}{\bar{a}_2}, ..., \frac{b_m}{\bar{a}_m}\right\} \qquad (9)$$

If station j is considered as a bottleneck station, the capacity allocation to products is performed according to station j capacity consumption using equation (10):

$$R_i = \frac{\bar{D}_i . a_{ij}}{\sum_{i=1}^{m}\bar{D}_i . a_{ij}} \qquad j = \text{bottleneck station} \qquad (10)$$

In the equation (10), $\overline{D}_i = \dfrac{1}{T}\sum_{t=1}^{T} D_{it}$ is equal to the average consumption of product i in periods $t = 1, 2, …., T$ based on the economic order quantity (EOQ) concept of Wilson [30].

The average demand for each product in each period is considered to be constant.

According to the ratio of average capacity consumption for each product (R_i) in station (j), C_{ij} matrix is defined as follows:

$$C_{ij} = \begin{bmatrix} C_1 R_1 & C_2 R_1 & \cdots & C_m R_1 \\ C_1 R_2 & C_2 R_2 & \cdots & C_m R_2 \\ \vdots & \vdots & \vdots & \vdots \\ C_1 R_n & C_2 R_n & \cdots & C_m R_n \end{bmatrix} \quad (11)$$

Each row of this matrix represents the allocated capacity to each product in various stages.

2-2. Mathematical Model of Each Product

After decomposing the main problem, the multi-stage, multi-period production planning model with production capacity constraint $(C_1 R_i, C_2 R_i, …., C_m R_i) = (b_1', b_2', …, b_m')$ is as follows:

decision parameters and variables

A_{jt} = Set-up cost of stage j and period t

X_{jt} = Production quantity in stage j and period t

C_{jt} = Variable production cost in stage j and period t

H_{jt} = Inventory holding cost in stage j and period t

I_{jt} = On hand inventory cost in stage j at the end of period t

b_j = Available resource in stage j

D_t = Order quantity of the finished product in stage j

S_j = Set-up time of stage j

$Y_{jt} = \begin{cases} 1 & \text{if machine } i \text{ is setup in period } t \\ 0 & \text{otherwise} \end{cases}$

Objective Function:

$$Min\, Z = \sum_{j=1}^{m}\sum_{t=1}^{T}[A_{jt}.Y_{jt} + C_{jt}.X_{jt} + H_{jt}.I_{jt}] \quad (12)$$

Constraints:

$$I_{m,t-1} + X_{m,t} - I_{m,t} = D_t \;(12) \quad t = 1, 2, …., T$$

$$I_{j,t-1} + X_{j,t} - I_{j,t} - X_{j+1,t} = 0 \quad (13)$$
$$t = 1, 2, …., T \quad j = 1, 2, …., m-1$$

$$\sum_{1}(a_{jt} X_{jt} + S_j Y_{jt}) \le b_j' \quad (14)$$

$$X_{ijt} \le Y_{jt}.b_j' \quad (15)$$

$$(X_{jt}, I_{jt}) \ge 0 \quad (16)$$

$$Y_{ijt} = \begin{cases} 1 & \text{if} \quad X_{j,t} \ge 0 \\ 0 & \text{otherwise} \end{cases} \quad (17)$$

In this model, equation (11) represents the objective function, which seeks to minimize the sum of set-up, holding and variable production costs.

Equation (12) ensures the supply of demand in each period. Equation (13) shows that in a network, total of in-flows to each node (j, t) is equal to out-flows from that node. Relation (14) represents the set-up and production times constraint required in a stage. Relation (15) ensures that set-up and installation costs are considered if the production process begins. Equations (16, 17) represent the type of decision variables.

3. France's Heuristic Approach (H.)

In this section we describe Franca's heuristic approach named (H.).

We get some of the ideas that we have obtained from details definition in our hybrid genetic algorithm (HGA). The main steps of this algorithm are as follows:

3-1. Initial Solution Obtaining Method (P1)

This method provides a primitive solution by repeatedly applying the Wagner-Whitin algorithm.

Wagner-Whitin algorithm is used to determine the optimal lot-size in multi-stage, single-product production planning problems with production capacity constraint. In this method first the capacity constraint is disregarded and lot-size is determined for finished products.

In this stage determined lot-sizes are equal to the previous stage values. The sequence is provided for sustainability.

After reapplying the Wagner-Whitin algorithm for *m* times, an initial solution is obtained.

This solution may be non-practical because in this model, the production capacity constraint is disregarded. If the resultant solution is not practical, return to the second stage; otherwise go to stage 3.

3-2. Initial Solution Estimation Method (P2)

This method starts with a non-practical initial solution. To find a practical solution, we transfer the production

among the periods. This technique consists of forward progress and backward regression.

In each step, an experiment is carried out for non-practical periods to transfer the production to other sections.

During this transfer, maximum production capacity and required production capacity quantities are compared.

In periods with resource shortage, production is transferred to periods with unused capacity.

Among the possible transfers, the best transfer with the aim of cost minimization and practical solution is selected.

These transfers are continued until the practical solution is obtained for the investigated period.

The new non-practical periods are identified and analyzed. Both steps are carried out until a practical solution or a maximum number of pre-determined iterations is obtained. If a practical solution is not obtained using this method, the method fails.

3-3. Improvement Method (P3)

This method gets a practical solution as an input and tries to improve it.

Cost reduction method uses the forward and backward amounts of production transfers method.

This method in addition to leveling the resource usage, maintains the practicality of the solutions. This method is considered as a local transfer.

Therefore, it begins with a practical solution and using production transfers for adjacent periods, seeks a lower-cost practical solution.

Adjacent transfers are a set of solutions which can be obtained through a production transfer. Transfer steps are recurred frequently until no more improvement is possible after a forward or backward step or maximum number of pre-determined iterations.

Finally, this method ends with a better solution or in the worst state, a solution with equal cost.

3-4. Incorporation Method (P4)

The solution resulted from the improvement method is a start point for incorporation method.

In the case that no improvement is obtained by the previous method, the solution resulted from the estimation method is used as start point.

In this method, over-load for each product at each period is selected and replaced with free time of machinery in other periods. This transfer ends after N steps. There are two different objectives for this kind of transfer. If the initial solution is non-practical, an effort is required to obtain a practical solution or reduce the resource usage in those periods.

If the initial solution is practical, incorporation method is suitable to obtain a low-cost solution.

4. Hybrid Genetic Algorithm (HGA)

Hybrid Genetic Algorithm is an extended genetic algorithm. This algorithm is based on the population of individuals like the rest of the genetic algorithms.

However, this algorithm is less constrained than other methods because there is no production planning structure in this method. In other words, this algorithm is designed for general production planning problems and is only based on the product components.

Unlike the traditional methods, HGA originally investigates all of the related variables [31, 32]. New solutions are obtained at each step of this algorithm through various combinations of these populations [33]. These populations also can be used to classify the genetic algorithm search. Now we describe the HGA steps:

4-1. Initial Solution Representation

Each solution is obtained by a $(T \times 2m)$ matrix (m: number of elements; T: number of periods).

This solution consists of lot-size and inventory for each element in each period.

This solution may be practical or non-practical. Each solution is illustrated as follows:

$$\begin{bmatrix} X_{11} & X_{12} & X_{13} & \cdots & X_{1T} & I_{11} & I_{12} & I_{13} & \cdots & I_{1T} \\ X_{21} & X_{22} & X_{23} & \cdots & X_{2T} & I_{21} & I_{22} & I_{23} & \cdots & I_{2T} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ X_{m1} & X_{m2} & X_{m3} & \cdots & X_{mT} & I_{m1} & I_{m2} & I_{m3} & \cdots & I_{mT} \end{bmatrix}$$
(18)

4-2. Fitness Function

Each solution has a value. This value is related to actual performance of the solution. Accordingly, practical and non-practical solutions can be obtained for each population group. A method to control impossible solutions is to use the cost and feasibility factors simultaneously. This method is shown in relation (19):

$$fitness = \begin{cases} Z > 0 & objective\ function\ value\ for\ a\ practical\ solution \\ Z = 0 & objetive\ function\ value\ for\ a\ non-practical\ solution \end{cases}$$
(19)

In this solution, value of the objective function is equal to the total cost value, if solution is practical; otherwise, value of the objective function is equal to zero.

Thus, fitness function has two modes, one mode represents the cost value for practical solution and the other indicates the practicality of the solution.

4-3. Population Size and Structure

Population reported in this paper consists of m groups. The relationship between these groups is based on the sequential production systems and has m levels. Each group of this population consists of two sub-classes. These two sub-classes represents production value vector (X_{jt}) and inventory value vector (I_{jt}).

Each of these groups stays in successive levels of the production hierarchy. Group m is the main root of the product tree and this forward group has a follower group named m-1. Also, m-1 follower group is a

forward group for group (*m*-2). Therefore, group *m*-2 is a follower group for *m*-1 group.
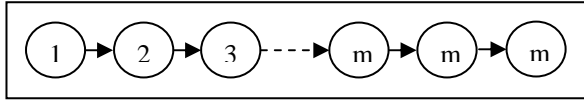
Fig 1. Production steps diagram for each product

Product tree is expanded accordingly to the end. According to the hierarchical structure of the product, solution vector of each follower group is connected to the following group solution vector.

This method transfers the follower group solutions to the forward groups and ensures the best solutions for the upper level elements of the product.

4-4. Initial Population

Each solution of the initial population is obtained using Wagner-Whitin algorithms like P1 in H. heuristic algorithm. Since this solution is non-practical, we apply P2 algorithm after P1.

Our objective is to produce different solutions for fixed set-up costs. These changes are randomly selected for a value between 100 fold of the set-up cost and 0.01 of the set-up cost.

These changes in some cases lead to high set-up costs and in other cases lead to low set-up costs. On the other hand, to generate more solutions, we use randomly uniform distribution for lot-size and inventory in allowable intervals.

These methods are studied as hybrid methods in subsections 4-6 and 4-7. Using these methods we obtain more solutions and also gain access to production process leveling.

4-5. Combination

In this step, each follower group is combined with a forward group and each combination generates a new solution. This group of new solution is added to the existing population.

For example, according to Figure 1; group 1 is concluded from the combination of the follower group 1 and forward group 2 and the new solution of group 2 is achieved from the combination of the follower group 2 and forward group 3.

This process continues until the final stage in the follower group *m* and amount of the demand. Since there is only one follower group for sequential hierarchy structure of this sub-set, a crossover action is generated. Because of this combination, *m* new groups are obtained.

In this investigation to obtain a combination, an algorithm is designed and experimented. New hybrid groups are added to the initial population. However, we notice that these hybrid groups are extremely related to product structure.

Therefore, we consider them as a proposed solution. In this algorithm, we start with the final elements that lie at the lowest level of the final product and then we deal with the highest level products.

This indicates the practicality of the solution with respect to constraints.

4-6. Memetic Algorithm

The amount of the production and inventory for each offspring is calculated as follows:

$$\begin{cases} X_{jt}^{offspring} = 0, \; I_{jt}^{offspring} = -a_{jt} & if & a_{jt} \leq 0 \\ X_{jt}^{offspring} = u\left[a_{jt}, X_{jt}^{parent1}\right], \; I_{jt}^{offspring} = X_{jt}^{offspring} - a_{jt} & if & a_{jt} > 0 \end{cases} \quad (19)$$

where

$$a_{jt} = d_{jt} - X_{jt}^{offspring} - I_{j,t-1}^{offspring} \quad (20)$$

Equation (20) refers to the element *j* production capacity in period *t*. Production of this element in period *t* is not necessary if $a_{jt} > 0$.

For this problem to be practical, the minimal amount of the product *j* in period *t* must be equal to a_{jt}. (a,b) distribution function shows the uniform random production values in (a,b). This function is used for a variety of solutions in the society.

Stochastic production comparison is carried out using minimum required production values (a_{jt}) and forward group production value in the hierarchical structure ($X_{jt}^{parent\ 1}$).

Then using new lot-size ($X_{jt}^{offspring}$) and equation (19), inventory (I) is determined. We continue this action according to method P1 until the inventory and production values are determined for all elements.

Then we start a leveling trend.

The objective of this trend is to change and update the inventory of all elements in periods (0-*T*) [34].

4-7. Wagner-Within Combination

This combination uses the Wagner-Within (WW) algorithm. We change set-up costs randomly for some elements and some periods according to relations (21) and (22):

$$X_{jt}^{parent1} = 0 \quad and \quad X_{jt}^{parent2} = 0 \quad if \quad S_{jt} \times 100 \quad (21)$$

$$X_{jt}^{parent1} > 0 \quad and \quad X_{jt}^{parent2} > 0 \quad if \quad S_{jt}/100 \quad (22)$$

We use Wagner-Within algorithm for each production stage. These set-up costs changes affect the production cost of the next level parts estimation for other periods. These effects refer to the production state of the upper level parts in previous periods.

The solution generated here is a practical solution according to (12) and (13). But it may not be a practical solution with respect to the resource capacity constraint (14).

In this case we use Frank et al. leveling trend. As described earlier, this leveling trend is used to find a practical solution.

In case the obtained solution is practical, we use the improvement trend. To select the elements in Wagner-Whitin combination, $u(1, m)$ uniform distribution is used [24].

4-8. Mutation

Hybrid genetic algorithm sometime uses the stochastic approaches to change the solutions regardless of the amount of fitness.

We evaluate the fitness and consistency of the solutions before mutation using P4 method.

After combination, we apply mutation operations in each period with $u(0,1)$ probability.

In this operation, m random numbers are generated between 0 and 1 which is smaller than 0.1 for each group. Mutation is applied according to equations (23), (24) and (25):

$$\begin{cases} X'_{jt} = X_{jt} + 0.1(b_j - X_{jt}) & \text{if} \quad X_{jt} \leq b_j & (23) \\ \\ X'_{jt} = X_{jt} - 0.1(X_{jt} - a_{jt}) & \text{if} \quad X_{jt} > b_j & (24) \end{cases}$$

$$I'_{jt} = X_{jt} - a_{jt} \qquad (25)$$

Equations (23) and (24) are used according to limited resource and production constraints.

These new groups are also added to the initial population.

4-9. Restart (Selection)

In this algorithm, we use restart strategy, because the existing population shows a few of the evaluated solutions.

We implement all of the existing population solutions and in each group a solution with minimum objective function value is selected.

Since the obtained solution is the best solution until now, they are similar to the initial population solutions except that they may have better objective function values.

To increase the number of solutions, some steps of the H. method must be repeated. The generated values by restart method are used when we use the return approach.

In these experiments, restart is used 20 times.

Stop criteria in HGA could be equal to the maximum number of the generated solutions or implementation time constraint.

If this algorithm does not obtain a practical solution, we will not be able to say with certainty that this is a non-practical problem.

Even, one solution does not ensure the practicality of this problem. HGA implementation steps are illustrated in Figure 2.

4-10. Computational Results

Hybrid genetic algorithm is written with Visual Basic programming language.

300 problems with various dimensions have been considered for the program testing.

Domains, which have been used to generate the examples, are provided in Table 1.
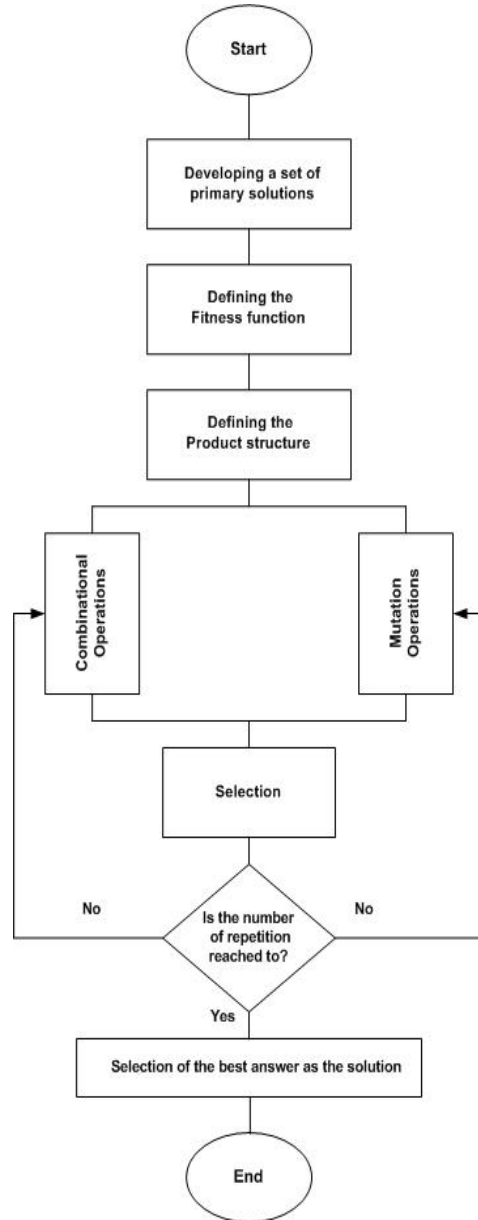
Fig 2. HGA flowchart

These domains are used by Rigna [34]. In these problems, the sequential production structure has been used. Sequential structure means that each element has exactly one previous and one next sample.

Number of steps in each problem and comparison results are provided in Table (4).

In each row of Table (4) 60 test problems are generated using distribution functions in Table (3) and objective function values for three methods of HGA, MA and H. have been compared.

Table 3. Uniform distribution of the stochastic examples

Parameter	Interval
C_{jt}	u(1.5,2)
A_{jt} For Low setup cost	u(5,90)
A_{jt} For high setup cost	u(50,950)
H_{jt}	u(0.2,0.4)
d_{jt} For find items	u(0,180)
d_{jt} For no find items	u(0,18)

Table 4. Comparative results of HGA method versus MA and H.

Number of steps	H.	MA	HGA
10	9.3	8.7	7.4
20	11.5	10.9	9.8
30	12.8	12.2	11.1
40	16.1	15.2	14.4
50	25.2	22.8	21.3

From the above table, it can be seen that HGA method has lower cost in comparison with H. and MA.

5. Resource Leveling

To implement resource leveling, the surplus capacity is calculated for each sub-problem using equation (26):

$$RC_i = (b_i^t - \sum_{j=1}^{m} \overline{D}_i.a_{ij}) \qquad (26)$$

The total remaining capacity is calculated using equation $(RCT = \sum_{j=1}^{m} RC_i)$.

This remaining capacity is distributed among sub-problems according to the used resource capacity. In this method, less capacity is allocated to sub-problems with more remaining capacity and vice versa.

Resource leveling is implemented according to equation (27) to achieve better feasible solutions:

$$BA_i = RCT.\frac{\overline{D}_i.a_{ij}}{\sum_{i=0}^{n} b_i^t.a_{ij}} \qquad (27)$$

Each sub-problem with the new justified capacity is resolved using hybrid genetic algorithm.

Therefore, according to the resource capacity increase in problems that need more capacity and resource capacity reduction in problems that require less capacity (resource leveling), the total cost decreases. This operation continues until the total cost difference between two successive steps (ΔTC) is less than ε. ε is determined according to the required accuracy.

6. Original Problem Solution Algorithm

Step 1: Decompose the original problem to *n* individual problems using Lagrange multiplier in limited resource.
Step 2: Solve each sub-problem using hybrid genetic algorithm.
Step 3: Calculate the remaining capacity of each sub-problem with respect to the allocated capacity.
Step 4: Implement the resource leveling operation for all total remaining capacities (TRC).
Step 5: Return to step 2 and continue until the stop criteria is reached.

7. Design of Experiment

To evaluate the proposed algorithm's performance, 300 stochastic problems with various dimensions have been designed. Their characteristics are as follows:

1. Problem dimensions: (N.M.T) = $(3 \times 3 \times 5)$ up (N.M.T) = $(5 \times 8 \times 15)$. List of the problems are provided in Table (3).
2. Set-up time and cost for each product in each period are determined in random and from (0,10) uniform distribution.
3. Inventory holding and production variable costs for each product in each period are also determined from (0,10) uniform distribution.

Order quantity of each product in each period is selected randomly from (0,10) uniform distribution.

4. Machinery production capacity in each step is randomly determined from (15,30) uniform distribution.

To solve the problems above, two programs have been written in Visual Basic environment. The first program is written combining the genetic algorithm and Lagrange multiplier (HGA-LR) and the second one is written with Memetic Algorithm (MA).

The total cost and time for each method are provided in Table (5). Comparison results demonstrate that HGA-LR costs solves the problem in much less time, in addition to better solutions and lower costs. This algorithm also solves large size problems in less than 10 hours with near optimal solutions while it takes 10 hours (maximum time) to solve these problems by MA and Lingo algorithms. The improvement obtained by this algorithm is 25.8 percent in time reduction and 19.3 percent in cost reduction.

8. Conclusion

In this approach a meta-heuristic approach has been developed to decompose large and complex problems to small sub-problems based on Lagrange multipliers and combining them with hybrid genetic algorithm to determine the dynamic lot-size in multi-stage, multi-product and multi-period production planning problems with limited resources and minimizing the total of set-up, production and inventory holding costs. This heuristic approach starts with decomposing the main problem to *n* sub-problems. After solving each sub-problem using hybrid genetic algorithm (Genetic Algorithm + local search), remaining capacities are calculated and resource leveling is carried out.

Table 5. Comparison among HGA-LR, MA And lingo methods results

Problem Size (N.M.T)	Number of Problems Solved	MA		Lingo		HGA-LR	
		Solution time (minute)	Total cost ($)	Solution time (minute)	Total cost ($)	Solution time (minute)	Total cost ($)
3×3×5	5	0.45	49	0.32	45	0.22	45
3×3×10	8	0.69	84	0.59	82	0.69	71
3×3×15	10	1.58	97	1.12	91	0.89	92
3×4×5	5	2.12	85	2.45	74	1.85	84
3×4×10	8	3.82	99	3.14	82	2.95	96
3×4×15	10	5.89	135	8.15	131	7.11	122
3×5×5	5	14.59	235	20.28	220	11.12	219
3×5×10	8	17.62	315	36.12	245	15.15	212
3×5×15	10	19.42	328	40.11	280	15.85	215
3×6×5	5	127.32	299	245.19	240	112.11	215
3×6×10	8	215.15	315	385.19	310	175.24	288
3×6×15	10	218.15	339	-	-	215.16	315
4×3×5	5	315.16	341	485.12	321	225.84	321
4×3×10	8	319.42	483	459.18	425	252.18	418
4×3×15	10	325.42	519	-	-	251.25	482
4×4×5	5	428.42	485	411.51	325.12	322.17	352
4×4×10	8	432.15	496	-	-	324.16	480
4×4×15	10	-	-	-	-	392.17	491
4×5×5	5	545.81	568	-	-	412.19	542
4×5×10	8	592.58	592	-	-	431.19	594
4×5×15	10	-	-	-	-	480.15	612
4×6×5	5	661.15	625	-	-	482.51	1605
4×6×10	8	-	-	-	-	511.16	594
4×6×15	10	-	-	-	-	520.82	624
5×7×5	5	-	-	-	-	525.92	722
5×7×10	8	-	-	-	-	548.22	748
5×7×15	10	-	-	-	-	565.17	827
5×8×5	5	-	-	-	-	522.19	752
5×8×10	8	-	-	-	-	582.15	812
5×8×15	10	-	-	-	-	592.15	915

These steps are continued until we reach the stop criteria. Computational results indicate that composition and decomposition approach based on the limited resource Lagrange multipliers and hybrid genetic algorithm is a suitable solution for lot-size determination in similar problems. Also, combination of the composition and decomposition approaches based on the limited resource Lagrange multipliers and meta-heuristic approaches provides better results and more suitable solutions in resource allocation and resource leveling operations.

References

[1] Tempelmeier, H., *"Resource–constrained materials – requirements planning –MRP"*, Production Planning and Control 8(5), 1997, PP. 451-461.

[2] Franca, P.M., Armentano, V.A., Berietta, R.t., & Clark, A.., "*A heuristic qm for Lot-sizingn multi-stage systems*". Computers and Operations Research 24(9), 1992, PP. 861-874.

[3] Carvalho, M.F., Fermandes, C.A.O., & Ferreira, P.A.V., *"Multi-product multi-stage production scheduling for manufacturing systems"*. Production Planning and Control, 10(7), 1995, PP. 671-681.

[4] Chu, S.C.K., *"A mathematical programming approach towards optimized master production scheduling"*, International Journal of Production Economics, 38, 1995, PP. 269-279.

[5] Tang, L., Liu, J., Rong, A., & Tang, Z., *"A mathematical programming model for scheduling steelmaking-continuous casting production"*. European Journal of Operational Research, 120, 2000, PP. 423-435.

[6] Weinstein, L., & Chung, C.H., *"Integrating maintenance and production decisions in a hierarchical production planning environment computers and operations research"*, 1999, 26, PP. 1059-1074.

[7] Florian, M., Lenstra, J.K., & Rinnooy Kan, A.H.G., *"Deterministic production planning and complexity"*. Management Science 26, 1980, PP. 669-679.

[8] Rardin, R.L., "*Optimization in operations research*", Prentice-Hall, NJ, 1998.

[9] ROY, R., & Meilke, S.E., *"The role of discrete event simulation techniques in finite capacity Scheduling"*. Journal of the Operational Research Society, 46, 1995, PP. 1310-1321.

[10] Holland, J.H., *"Adaptation in natural and artificial systems'*, 2nd Edition, University of Michigan Press, MIT Press, 1992.

[11] Goldberg, D.E., *"Genetic algorithms in search optimization and machine learning"*, Addison-Wesley, MA. 1989.

[12] Wang, H.F., & Wu., K.Y., *"Hybrid genetic algorithm for models with permutation property"*. Submitted to Computers and Operations Research, 2001.

[13] Bitran, G.R., & Yanasse, H.H., *"Computational complexity of the lot size problem"*. Management Science 28, 1982, PP. 44-56.

[14] Maes, J., Meclain, J.O., & Van Wassen hove, L.N., *"Multi-level capacitated lot sizing complexity and LP based heuristics"*. European Journal of Operational Research 53, 1991, PP.131-148.

[15] Bahl, H.C., Ritzman, L.P., & Gupta, J.N.D., *"Determining lot-sizes and resource requirement"*: A Review Operations Research 34(3), 1987, PP. 237-249.

[16] kuik, R., Solomon, M., & Van Wassen hove, L.N., *"Batching decisions: structure and models"*. European Journal of Operational Research 75, 1994, PP. 243-263.

[17] Simpson, N.C., *"Questioning the relative virtues of dynamic lot sizing rules"*, Computers & Operations Research 28, 2001, PP. 899-914.

[18] Franca, P.M., Armentano, V.A., Berretta, R.E., & Clark, A., "*A heuristic for lot-sizing in multi-stage systems*". Computers and Operations Research 24(9), 1997, PP. 861-874.

[19] Ozdamar, L., & Barbarosoglu, G., *"An integrated lagrangean relaxation-simulated annealing approach to the multi-level"*. Multi-Item capacitated lot-sizing problem International Journal of Production Economics 68, 2000, PP. 319-331.

[20] Dellaert, N.P., & Jeunet, J., *"Randomized multi level lot sizing heuristics for general product structures"*, European Journal of Operational Research, 148, 2003, PP. 211-228.

[21] Ouenniche, J., & Fayez, F.B., *"The two – group heuristic to solve the multi-product"*, Economic Lot-Sizing and Scheduling Problem in Flow Shops, European Journal of Operational Research, 129, 2001, PP. 539-554.

[22] Katok, E., Lewis, H.S., & Harrison, T.P., *"Lot sizing in general assembly systems with setup costs, setup times and multiple constrained resource"*. Management Science 44(6), 1998 , PP. 859-877.

[23] Harrison, T.P., & Lewis, H.S., *"Lot sizing in serial assembly systems with multiple constrained resources"*, Management Science 42(1), 1996, PP. 19-36.

[24] Wagner, H.M., & Whitin, T.M., "*Dynamic version of the economic lot size model*". Management Science 5(1), 1958, PP. 89-96.

[25] Tempelmeir, H., & Derstroff, M., *"A Lagrange an – based heuristic for dynamic multi–level multi–time constrained lot sizing with setup times"*. Management Science 42(5), 1996, PP. 738-757.

[26] Linet, O., & Gulay, B., "*An integrated lagrangean relaxation-simulated annealing approach to the multi-item capacitated lot sizing problem*", International Journal of Production Economic. 68, 2000, PP. 319-331.

[27] Hsiao, F.W., Kuang, "*Modeling and analysis for multi-period*", Multi-product and Multi-resource Production Scheduling, Journal of Intelligent Manufacturing, 14, 2003, PP. 297-309.

[28] Geoffrion, A.M., *"Lagrangean. relaxation for integer programming"*, Mathematical, Programming Study, 2. 1974, PP. 82-114.

[29] Thizy, J.M., *"Lagrangean heuristics for the capacitated multi-item capacitated lot sizing problem"*, Infor, 29(4).1991, PP. 271-283.

[30] Wilson, R.H., *"A scientific routine for stock control"*, Harvard Business Review, B(1), 2002, PP. 116-128.

[31] Moscato, P., *"Memetic algorithms: A short introduction"*. In: Crone, D., Glover, p., Dorigo, M.(EDS), New Ideas in Optimization. McGraw-Hill, Maidenhead, UK, 1999, PP. 219-234.

[32] Moscato, P., Cotta, C., *"A gentle introduction to meme tic algorithms. In: glover"*,F., Kochenberger, G.(Eds), Handbook of Metahearistics. Kluwer Academic Publisher, Boston, 2003.

[33] Berretta, R.E., & Moscato, P., *"The number partitioning problem":* An open challenge for Evolutionary Computations In: Corne, D., Glover, P., Dorigo, M. (Eds), New Ideas in Optimization. McGrow-Hill, Maiden head, UK. 1999, PP. 261-278.

[34] Regina, B., & Luiz, F.R., "*A meme tic algorithm for a multistage capacitated lot–sizing problem"*, International Journal of Production Economics, 2004.