An improved Genetic Algorithm For Fuzzy Production PlanningProblems with Application

Jalal Abdulkareem Sultan[1], Omar Ramzi Jasim[2] ,Sarmad Abdulkhaleq Salih[3]
Department of Mathematics, College of Al-Hamdaniah Education, University of Mosul, Al-Hamdaniah, Iraq
Department of Accounting, College of Management and Economic, University of Al-Hayat, Erbil, Iraq
Department of Statistics & Information, College of Mathematics and Computer Sciences, University of Mosul, Mosul, Iraq
Jalalstat2011@gmail.com , omarstatis@yahoo.com sarmadstat2013@gmail.com

Abstract— Production Planning or Master Production Schedule (MPS) is a key interface between marketing and manufacturing, since it links customer service directly to efficient use of production resources. Mismanagement of the MPS is considered as one of fundamental problem in operation and it can potentially lead to poor customer satisfaction. In this paper, an improved Genetic Algorithm (IGA) is used to solving fuzzy multi-objective master production schedule (FMOMPS). The main idea is to integrate GA with local search operator. The FMOMPS was applied in the Cotton and medical gauzes plant in Mosul city. The application involves determine the gross requirements by demand forecasting using artificial neural networks. The IGA proved its efficiency in solving MPS problems compared with the genetic algorithm for fuzzy and non-fuzzy model, as the results clearly showed the ability of IGA to determine intelligently how much, when, and where the additional capacities (overtimes) are required such that the inventory can be reduced without affecting customer service level.

Keywords—Master Production Schedule; Fuzzy Model; Improved Genetic Algorithm; Multi-Objective Optimization.

I. INTRODUCTION

Production planning at tactical level, also known as Master Production Scheduling (MPS), is a philosophy being used for many years and by many industries. One of the difficulties involved in creating tactical production plans is the fact that demand varies over time according to innumerous unexpected reasons.

However, industry resources are limited and a perfect demand forecast is impossible. These things make it very hard for the production planner. One can suggest expanding capacity during high demand periods, but this takes time and requires investment, and besides, at times of low demand, the company will have even more idle capacity.

A production planning system must find a good, perhaps optimal, plan to maximize service levels and, at the same time, minimize inventory levels and resource utilization in face of these difficulties. However, these are conflicting objectives and a tradeoff to reach acceptable values must be made. With this in mind, several techniques have been proposed to perform optimization on production planning problems based on, for instance, linear and non-linear programming, dynamic-lot sizing and meta-heuristics [5].

In general, master scheduling problems are NP-hard problems, that is, unlikely there is no an algorithm that can find an optimal MPS solution in polynomial time. In practical, this means that the processing time required to solve such problem will quickly and enormously grow as the problem size grows. For this reason, truly optimal solution is quite difficult to be found. Therefore, meta-heuristic or evolutionary algorithms such as genetic algorithm, simulated annealing, tabu search, etc. are also employed to obtain optimum solution [12.]

In this paper, an improved Genetic Algorithm (IGA) is used to solving fuzzy multi-objective master production schedule (FMOMPS). The main idea is to integrate GA with local search operator. The purpose of this paper is to apply the FMOMPS in the Cotton and medical gauzes plant. The application involves determine the gross requirements by demand forecasting using artificial neural networks.

The paper is organized as follows: Section 2 present a brief revision of master production scheduling and genetic algorithms. The improved algorithm to solve FMOMPS problems has been proffered in Section 3. The Comparison between IGA and GA is given on Section 4. In Section 5, a FMOMPS has been created for the Cotton and medical gauzes plant. The conclusion of this study has been presented in Section 6.

II. A BIT FROM PRODUCTION PLANNING OPTIMIZATION AND GENETIC ALGORITHMS

This section explains some of the fundamental concepts behind production planning, in particular, master production scheduling and genetic algorithms.

2.1 Master production scheduling: Basics

According to the American Production and Inventory Control Society (APICS), a master production plan is a declaration of what the company expects to manufacture, which become a series of planning decisions that drives the material requirements planning (MRP) system. It represents what the company intends to produce expressed in configuration, quantities and specific dates. The master plan is not a sales forecast, which represents a demand declaration. It should take into consideration the demand, pending orders, material availability, projected ending inventory levels, capacity availability, managerial policies and goals, among others. The master plan is the result of the master production scheduling [7].

MPS problems usually involve conflicting objectives, like minimizing inventory and set-up times, and maximizing service levels. Because of all this, use of heuristics and meta-heuristics are suggested for the resolution of these types of problems. Several artificial intelligence metaheuristics have been applied to optimization, among them, genetic algorithms, taboo search, ant colony, beam search and simulated annealing [12].

Wu et al. (2002) [19] built a mathematical model of MPS and developed a genetic algorithm incorporating several techniques to satisfy constraints for making an optimized MPS of production lines with both assembly and processing. *Vieira & Ribas* (2004) [5] applied simulated annealing to solve production planning problem, more specially, master production schedule. *Soares & Vieira* (2008) [12] introduced new genetic algorithm structure for solving MPS problem. This study formulates the fitness function, which aims to minimize inventory level, maximize service level, minimize overtime and minimize inventory level below safety stock.

2.1.1 Mathematical Model of FMOMPS

The master production schedule problem can be mathematically modeled as a mixed integer program as follows [12]:

K Total quantity of different products

R Total quantity of different productive resources

P Total number of planning periods

TH Total planning horizon

OH_k Initial available inventory (on-hand), at the first scheduling period

GR_{kp} Gross requirement for product k at period p

BS_{kp} Standard lot size for product k at period p

NR_{kp} Net requirement for product k at period p, considering infinity capacity

SS_{kp} Safety inventory level for product k at period p

UR_{kr} Production rate for product k at resource r (units per hour)

AC_{rp} Available capacity, in hours, at resource p at period p

Decision variable for the MPS are:

BN_{kpr} Quantity of standard lot sizes needed for the production of the product k at resource r, at period p (number of lots)

MPS_{kpr} Total quantity to be manufactured of the product k at resource r, at period p

$MPST_{kp}$ Total quantity to be manufactured of the product k at period p (considering all available resources)

BI_{kp} Initial inventory level of the product k at period p

CUH_{rp} Capacity used from the resource r at period p

CUP_{rp} Percent rate obtained from the relation of the number of hours consumed from the resource r at period p, and the available number of hours to the same resource and period

GR_{kp} Gross requirement for product k at period p

RNM_{kp} Requirements not met for product k at period p

SL_{kp} Service level, relation of the requirement met, and the gross requirements for product k at period

According to Supriyanto (2011) [16], the equivalent crisp model for fuzzy multi-objective MPS can be expressed as following.

$$max \quad w_1\mu_{z1}(x) + w_2\mu_{z2}(x) + w_3\mu_{z3}(x) + w_4\mu_{z4}(x)$$

$$s.t.$$

$$\mu_{z1}(x) \le \frac{Z_1^1 - Z_1(x)}{Z_1^1 - Z_1^0}$$

$$\mu_{z2}(x) \le \frac{Z_2^1 - Z_2(x)}{Z_2^1 - Z_2^0}$$

$$\mu_{z3}(x) \le \frac{Z_3^1 - Z_3(x)}{Z_3^1 - Z_3^0}$$

$$\mu_{z4}(x) \le 1 - \frac{Z_4^1 - Z_4(x)}{a}$$

$$\mu_{z4}(x) \le 1 - \frac{Z_4(x) - Z_4^1}{b}$$

$$CUH_{rp} - AC_{rp} \le OL_{max}$$

$$w_1 + w_2 + w_3 + w_4 = 1$$

$$\mu_{zi} \in [0,1]; i = 1,2,3,4$$

$$x_{krp} \ge 0 \text{ and integer} \qquad (1)$$

Where

$$Z_1(x) = \frac{\sum_{k=1}^{K}\sum_{p=1}^{P} EI_{kp}}{TH} \qquad (2)$$

$$Z_2(x) = \frac{\sum_{k=1}^{K}\sum_{p=1}^{P} RNM_{kp}}{TH} \qquad (3)$$

$$Z_3(x) = \frac{\sum_{k=1}^{K}\sum_{p=1}^{P} BSS_{kp}}{TH} \qquad (4)$$

$$Z_4(x) = \sum_{r=1}^{R}\sum_{p=1}^{P} OC_{rp} \qquad (5)$$

$$BI_{kp} = \begin{cases} OH_k & if(p = 1) \\ EI_{k(p-1)} & if(p > 1) \end{cases} \qquad (6)$$

$$EI_{kp} = max\left[0, \left(\left(MPST_{kp} + BI_{kp}\right) - GR_{kp}\right)\right] \qquad (7)$$

$$MPST_{kp} = \sum_{r=1}^{R} MPS_{kpr} \qquad (8)$$

$$MPS_{kpr} = BN_{kpr} * BS_{kpr} \qquad (9)$$

$$RNM_{kp} = max\left[0, \left(GR_{kp} - \left(MPST_{kp} + BI_{kp}\right)\right)\right] \qquad (10)$$

$$BSS_{kp} = \max\left[0, \left(SS_{kp} - EI_{kp}\right)\right] \qquad (11)$$

$$CUH_{rp} = \sum_{k=1}^{K} \frac{(MPS_{krp})}{UR_{kr}} \tag{12}$$

$$OC_{rp} = max[0, (CUH_{rp} - AC_{rp})] \tag{13}$$

Where w_i denotes the weighting coefficients that show the relative importance among the fuzzy objectives; *x*: The MPS solution is obtained by solving the above crisp model; $Z_i^0$and Z_i^1: the value of the objective function such that the degree of membership function is 0 or 1 respectively; *a* and *b*: the limit of the admissible violation of overtime objective,; OL_{max}: Allowed maximum overtime.

2.2 Genetic Algorithms

Genetic algorithms (GAs) are a family of computational models developed by Holland [9], which is based on the principles of natural biological evolution. For a specific problem, GA codes a solution candidate as an individual chromosome. The approach begins with an initial chromosome population which represents the set of initial search points in the solution space of the problem. Then the genetic operators such as selection, crossover and mutation are applied to obtain a new generation of chromosomes. Since the operators are under the principle of "survival of the fittest, extinction of the unfitness", it is expected that over all the quality the chromosomes will be improved with the generation increasing. This process is executed iteratively until the termination criterion is met, and the best chromosome of the last generation is reported as the final solution [2].

Execution steps in a genetic algorithm approach can be generically represented by the flowchart in Figure 1. Starting from the use of any, preferably good, heuristic, an initial population of individuals is created. A fitness function measures how good (fit) a solution (individual) is. The process simulating the natural selection starts as each individual has its fitness factor calculated. At this phase, individuals with low fitness are extinct and replaced by individuals with higher fitness, so that the population always remains with the same size.

Next step consists of a mating among individuals. It is important to highlight that individuals with higher fitness will have greater chances for being selected for the crossover; consequently, there will be a tendency to propagate their genes for the next generations [2].

III. IMPROVED GENETIC ALGORITHM TO SOLVE FMOMPS PROBLEMS

In this section, combines Genetic Algorithm with local search operator to present an improved genetic algorithm (IGA). The pseudo-code of the IGA is presented in follow:

```
Step 1: Initializing parameters of IGA
algorithm and set Nd=0;
Step 2:
2.1: Define the parameters of FMOMPS
problem;
2.2: Generate initial population;
2.3: Evaluate the fitness value of the
    initial population;
Step 3: Decade loop Nd=Nd+1
Step 4:
4.1: Selection;
4.2: Crossover;
4.3: Mutation;
4.4  Apply local search operator;
4.5  Evaluate the fitness value of each
chromosome;
Step 5: Terminating Criterion Control;
    Repeat Steps 3-5 until a terminating
    criterion is satisfied;
```

A. Input Information

For the fuzzy multi-objective master scheduling optimization, the software implemented takes into consideration as much parameters as possible found in actual industrial environments:

- aspiration level for every objective function
- number and description of products;
- number and description of productive resources (production lines, workstations, machines);
- number and description of time periods and duration for each period (periods with different durations are allowed);
- initial (on-hand) inventories: product quantities in the beginning of the planning horizon;
- gross requirements: needed quantity per product per period, estimated from forecasting and customers orders;
- batch sizes: production standard lot sizes per product per period;
- safety inventory level per product per period;
- production rate: the quantity a resource can manufacture of a product per time unit;
- setup time per product, non-depended on operation sequence; and
- available capacity per resource per period;

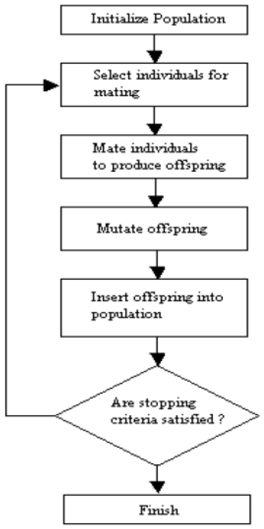
Initialize Population
Select individuals for mating
Mate individuals to produce offspring
Mutate offspring
Insert offspring into population
Are stopping criteria satisfied ?
Finish

Fig. 2. Genetic algorithm flowchart

B. *Chromosome Representation*

The data structure created to represent an individual (an MPS) is composed of several levels of smaller structures (objects), most of them as singly linked lists (Figure 2).

The lowest level structure is the gene, which represents a product quantity to be manufactured. Going up in the hierarchy, the next level is a chromosome, an object containing a list of genes. This represents the quantities of a product that can be manufactured by one of the resources. A list of chromosomes is used to model all the products considered in a certain period. Therefore, a time period object is represented by a list of chromosomes. An individual is a list of periods and, in the highest level, the population is represented by a list of individuals.

This data structure allows for a complete freedom regardless of the problem size to be modeled. An individual can grow and adapt its genetic structure according to the MPS considered. The size-limiting factor will depend only on the computer power (memory and speed). This representation, however, can demand longer processing time.

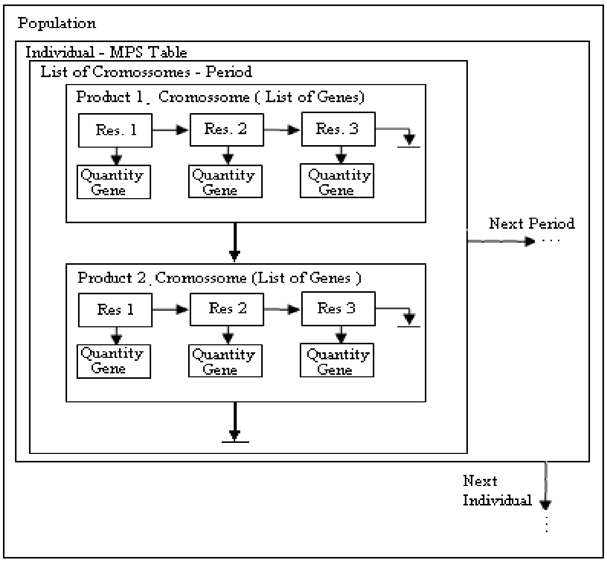

Fig. 2. Chromosome representation

C. *The fitness function*

The goal of equation 1 is to maximize the weighted additive of membership function of objectives. Assuming DM's preference (w_1=w_2= w_3= w_4=0.25), the fitness function for IGA may be written as below.

```
function z=f(x)
$\mu_{z1}(x) = \cdots$
$\mu_{z2}(x) = \cdots$
$\mu_{z3}(x) = \cdots$
$\mu_{z4}(x) = \cdots$
$g(x) = \cdots$
$z=(0.25\mu_{z1}(x)+0.25\ \mu_{z2}(x)+0.25\ \mu_{z3}(x)+0.25\mu_{z4}(x))+$
 $g(x)$;
end
```

D. *Initial Population Creation Function*

The size of the population (number of countries) and the way the initial population is created have a significant influence in the performance of the algorithm and to the quality of the results. The ideal situation would be to have the greatest possible diversity of countries to better through the search space. The pseudo code of population creation function for multi resources, multi products and multi periods may be written as follows:

```
for k=1:K
    for r=1:R
        if UR(k,r)≠ 0
for p=1:P
IP=randi([0,round(GR(k,p)/BS(k,p))],nPop,1
);
IP=IP*BS(k,p);
Pop=[Pop  IP];
IP=[];
end
        else
  for p=1:P
     IP=zeros(nPop,1);
     Pop=[Pop IP];
     IP=[];
  end
        end
    end
end
```

Where randi([imin,imax],m,n) returns m-by-n matrix containing integer values drawn from the discrete uniform distribution on the interval (imin,imax) [10].

This heuristic approach fills up the most possible diversity, with values always respecting the standard lot size restriction. Consider a hypothetical situation where the gross requirement for a given product at a certain time period (time bucket) is 3,000 units, the standard lot size is 500 units, and there are four possible productive resources available to make the product, the genes for the first country in the population in the first period would be "{0;500;1,000;1500}" second country would have "{2,000; 3,000; 2,500; 500}", and so on, sequentially for all individuals in the population.

In the initial population, we generate N_{pop} of the chromosomes then calculate the fitness function to each of them.

E. *Genetic Operators*

Inspired by GA, genetic operators of selection, crossover, and mutation are exerted on individuals to diversify population of chromosomes.

1) ***Roulette wheel selection Operator:*** In this method, it states that the probability for an individual to be selected is equal to its fitness value normalized with respect to the total fitness of the population. This method takes advantage of individuals that make large jumps in improvement by giving them a much greater chance of reproducing. In other words, roulette decides which individuals will do crossover, and therefore, generate siblings, which will carry their genetic information forward.

2) *Single-point crossover Operator:* In single-point crossover one crossover position (*k*) is selected uniformly at random in the interval [1,2, . . . ,$N_{var} - 1$], and the variables exchanged between the colonies about this point, then two new colonies are produced.

By the operation of crossover, better genetic scheme in a chromosome is inherited from the current generation to the next generation at the predetermined crossover probability (p_c). When two parent chromosomes are located at local optimums, crossover operation provides child country chance of jumping from local optimum. In this algorithm when crossover between two colonies make bad results , we back to the old colonies before crossover.

3) Mutation Operator*:* In each iteration we select some genes of the weakest chromosomes and replace them with new ones, randomly. The Mutation operator can be modeled as:

$$Pop(j,k) = round\left(\frac{unifrnd(VarMin(k),VarMax(k))}{BS}\right) * BS \quad (17)$$

Where $Pop(j,k)$ represent gene(k) in chromosome (j), $VarMin(k)$ and $VarMax(k)$ represent lower and upper bound for gene (k) respectively, and $unifrnd(A,B)$ returns a random number generated from the continuous uniform distributions with lower and upper endpoints specified by A and B, respectively [10]. The chromosomes are selected at the predetermined mutation probability (p_m).In this algorithm when mutation make bad colony, we back to the old chromosome before mutation.

F. Apply Local Search Operator

In the evolutionary progress of (IGA), population diversity may be lost and premature convergence always happens. Local Search is very useful in situations requiring the restoration of population diversity. Include adding or subtracting one production batch size to a specific variable in the chromosome. The local search operator can be modeled as:

$$MPS_{i+1} = MPS_i + BS_{kpr} \quad (18)$$

Where MPS_{i+1} represent master production schedule after modification, MPS_i is master production schedule before modification, BS_{kpr} is Standard lot size for product k at resource *r* at period p.

G. Evaluation

The final step concerns the evaluation of the fitness for the new individuals created.

H. Stopping Criteria

Different conditions may be selected as termination criteria including reaching a maximum number of iterations or having negligible improvement in objective functions.

IV. CREATE FMOMPS FOR COTTON AND MEDICAL GAUZES PLANT

In this section, we create a fuzzy master production schedule to Cotton and medical gauzes plant. The plant was established in 2008, and currently provides a wide range of products, including different sizes of medical cotton and different types of gauzes.

A. Determine Input Parameters of MPS

Table 1 shows the details of MPS parameters required to create MPS problem of towels plant.

TABLE 1. THE INPUT PARAMETERS OF MPS

Parameters	Value
K	4 (C500,C400,C100,C50)
R	6 (S1,S2,S3,S4,S5,S6)
P	4 weeks (1st contain day off)
BS_{kp}	12 unit for all products and all periods
UR_{rp}	Table 2.
OH_k	(600, 480, 360, 360)
SS_{kp}	120 unit for all products and all periods
AC_{rp}	40 hours/period for each resource
OL_{max}	25 hours/period for each resource

B. Determine Gross Requirements

During the process of formulate MPS, as the most important input, gross requirements (demand) forecast has play great role in the final formation of MPS. BP neural network model has strong fault-tolerant performance, learning performance, self-adaptive performance and non-linearity map ability, and it is adaptive to solve some problems like non-determinacy inference of complex causal relation, judgment, recognition, classification and so on [1] .So try to use BP network algorithm in the MPS gross requirements forecast model.

TABLE 2. THE PRODUCTION RATES (UNIT/HOUR)

	S1	S2	S3	S4	S5	S6
C500	20	20	20	20	20	20
C400	25	25	25	25	25	25
C100	100	100	100	100	100	100
C50	180	180	180	180	180	180

We used the weekly demand data for each product. The sample data covered the time period from 31 January 2009 to 31 December 2013 and contains a total of 202 observations. The data were divided into training, validation and testing set for conducting the experiment in order to determine the best neural network structure.

In general, a three layer feed forward neural network model has been used for this study. The logistic and identity function have been used as activation function for the hidden nodes and output node respectively. Since multi-step ahead forecasting will be done using iterative procedure, only one output node is employed. Hence, the model uncertainty is associated only with the number of input nodes *(p)* which is the number of lagged observations in this case and number of hidden layer nodes *(q)*. The number of input nodes and hidden nodes were determined with the help of Monte Carlo simulation. We varied the number of input units from 1 to 12 as it plays a significant role in mapping the autocorrelation structure. The number of hidden units varied from 2 to 12.

By using Pletarion Synapse Neural Networks development environment, we find the best structure and gross requirements for each product as seen in table 3.

TABLE 3. THE BEST ANN STRUCTURES AND GROSS REQUIREMENTS

Prod.	Str.	Weeks			
		1st	2nd	3rd	4th
C500	7,9,1	3856	5780	4980	5322
C400	6,7,1	2167	1978	2377	2578
C100	5,4,1	381	369	425	289
C50	9,8,1	468	521	584	596

C. Create Fuzzy Model

For evaluation, the model FMOMPS will be applied. First of all, the lower and upper bound of aspiration level for all objectives must be defined. The values of $Z_i^0$and Z_i^1 are presented in Table 4. It is important to note that the aspiration level must be reasonable values to avoid infeasible solution. The admissible violation parameters for overtime objective (a *and* b) are set as 5 and 4 hours respectively.

TABLE 4. THE LOWER AND UPPER BOUND OF ASPIRATION LEVEL

Objective	Z_i^0	Z_i^1
Ending Inventory Z_1	240	3200
Requirement not met Z_2	0	800
Inventory below safety stock Z_3	50	1000
Overtime Z_4	0	3

Having substituted the parameter values into Equation 1, the fuzzy multi-objective model for the MPS problem is obtained as follows:

$$max \;\; 0.25\mu_{z1}(x) + 0.25\mu_{z2}(x) + 0.25\mu_{z3}(x) + 0.25\mu_{z4}(x)$$

$$s.t.$$

$$\mu_{z1}(x) \le \frac{3200 \; - Z_1(x)}{2960}$$

$$\mu_{z2}(x) \le \frac{800 - Z_2(x)}{800}$$

$$\mu_{z3}(x) \le \frac{1000 \; - Z_3(x)}{950}$$

$$\mu_{z4}(x) \le 1 - \frac{3 - Z_4(x)}{5}$$

$$\mu_{z4}(x) \le 1 - \frac{Z_4(x) - 3}{4}$$

$$CUH_{rp} - AC_{rp} \le 5$$

$$\mu_{zi} \in [0,1]; i=1,2,3,4$$

$$x_{krp} \ge 0 \; ; k = 1 \ldots 5; r = 1 \ldots 4; p = 1 \ldots 6 \qquad (20)$$

D. Fuzzy Solution

Solving the above crisp single objective using HPSO algorithm, the degree satisfaction and achievement level of objectives for the optimum solution are obtained as follows.

$$\mu_{z1}(x) = 0.71 \,, \mu_{z2}(x) = 0.74, \mu_{z3}(x) = 0.84, \mu_{z4}(x) = 0.7$$

$$Z_1 = 1093 \,, Z_2 = 205.5 \,, Z_3 = 196 \,, Z_4 = 1.8$$

The best MPS solution is presented in Table 5. The value “zero” indicate that there are no products to be manufactured in the related resource and at the corresponding time period.

TABLE 5. THE BEST MPS SOLUTION FOUND

		Week			
	Res.	1st	2nd	3rd	4th
C500	S1	624	540	552	684
	S2	492	744	588	384
	S3	432	732	516	684
	S4	564	408	384	624
	S5	612	276	504	792
	S6	696	540	588	564
	Total	3420	3240	3132	3732
C400	S1	204	312	264	120
	S2	348	60	264	504
	S3	432	60	348	120
	S4	264	456	504	204
	S5	228	636	360	0
	S6	108	300	240	276
	Total	1584	1824	1980	1224
C100	S1	60	60	108	0
	S2	72	36	0	72
	S3	96	0	0	60
	S4	0	72	84	60
	S5	0	12	36	0
	S6	24	108	132	12
	Total	252	288	360	204
C50	S1	12	0	144	180
	S2	144	48	24	0
	S3	36	180	60	72
	S4	228	120	0	48
	S5	48	120	24	72
	S6	120	0	12	120
	Total	588	468	264	492

V. Comparison between IGA and GA

Soares *et al.* (2008) [12] and Supriyanto (2011) [16] solved the same MPS problem using genetic algorithm. To provide benchmarking, the result of Soares's study will be cited here as non-fuzzy solution, and the result of Supriyanto's study will be cited as fuzzy solution.

The parameter settings of IGA algorithm are described as follow: p_c is equal to 0.9, p_m is 0.7, the size of population is equal to 500 and the maximum number of iterations is equal to 1000 generations.

As a final result, the comparison of performance measures between IGA and GA for fuzzy model and non-fuzzy model for the given production scenario is presented by Table 6.

Table 6. The comparison between HPSO and GA

Al g.	Solution	EI	RNM	BSS	OC
GA	Fuzzy	4943	842	821	3
	Non Fuzzy	5228.5	985.7	585.7	4.33
IGA	Fuzzy	4785.7	833.85	450	0
	Non Fuzzy	4828.5	942.8	528.5	2.3

The non-fuzzy solution of GA yields high levels of all objectives while the non-fuzzy solution of HPSO yields lower inventory level, lower requirement not met, lower inventory below safety stock and lower overtime.

The Fuzzy solution of GA yields highest levels of all objectives compared with fuzzy solution of HPSO. It seems that the GA may be not able to assigned properly overtime ("where and when" question is not addressed accurately). Theoretically, overtime should be able to reduce inventory level if it is placed on appropriate resource and right time. In contrary, the HPSO can effectively address when the additional capacity must be substituted, how much it is required and at which resource should be added.

VI. Conclusions

This paper proposes an improved genetic algorithm which integrates GA with local search operator for solving fuzzy production planning, in particular, fuzzy multi-objective MPS problems and its performance is evaluated by create FMOMPS for Cotton and medical gauzes plant. The IGA can solve efficiently the crisp single objective model equivalent to the FMOMPS. It has ability to determine intelligently how much, when, and where the additional capacities (overtimes) are required such that the inventory can be reduced without affecting customer service level. The comparison results show the efficiency and capabilities of the improved algorithm in finding the optimum. Its performance has shown to be better than GA. The performance achieved is quite satisfactory and promising for solving fuzzy and non-fuzzy models of MPS.

References

[1] B. Zhang, and M. Hue, "Forecasting with artificial neural networks", The state of the art," *International Journal of Forecasting*, vol. 14, 1998.

[2] D.E. Goldberg," Genetic Algorithms in Search Optimization & Machine Learning", Addison-Wesley. Reading, 1989.

[3] G.E. Vieira, F. Favaretto, and P.C. Ribas," Comparing genetic algorithms and simulated annealing in master production scheduling problems", Proceeding of 17th International Conference on Production Research. Blacksburg, Virginia, USA, 2004.

[4] G.E. Vieira, and C.P. Ribas," A new multi-objective optimization method for master production scheduling problems using simulated annealing" International Journal of Production Research, Vol. 42, No. 21, Pp.4609-4622 , 2003 .

[5] G.E. Vieira," A practical view of the complexity in developing master production schedules: fundamentals, examples, and implementation", In J. W. Herrmann. Handbook of Production Scheduling, Springer, Maryland, USA pp. 149-176, 2004.

[6] J.A. Sultan,"Proposed Hybrid Techniques for Solving Fuzzy Multi-Objective Linear Programming with Application. Master thesis", university of Mosul, Mosul, Iraq ,2013.

[7] J. Cox, J. Blackstone," APICS Dictionary", 10th edn. APICS, Alexandria ,2001.

[8] J. F. Proud," Master Scheduling", 2nd Edition, John-Wile-Sons Inc., 1999.

[9] J.H. Holland," Adoption in Natural and Artificial Systems", University of Michigan, Ann Arbor ,1975.

[10] MathWorks. Matlab Documentation. MathWorks Inc., 2004.

[11] M. Garey, and D. Johnson," Computer, complexity and intractability. A guide to theory of NP-Completeness", Freeman, San Francisco, USA ,1979.

[12] M.M. Soares, and G.E. Vieira," A New multi-objective optimization method for master production scheduling problems based on genetic algorithm", International Journal of Advanced Manufacturing Technology, DOI 10.1007/s00170-008-1481-x. ,2008.

[13] N. S. Slack, " Operation Management. Prentice Hall", 3rd Edition, New Jersey, USA ,2001.

[14] P. Higgins, and J. Browne," Master production scheduling a concurrent planning approach. Production Planning & Control", Vol. 3, no.1.pp 2-18 ,1992 .

[15] P. Ribas," Análise do uso de têmpera simulada na otimização do planejamento mestre da produção", Pontifícia Universidade Católica do Paraná, Curitiba, 2003.

[16] S. Supriyanto," Fuzzy Multi-Objective Linear Programming and Simulation Approach to the Development of Valid and Realistic Master Production Schedule", Doctor thesis, university of Duisburg, Essen, Germany, 2011.

[17] T. Bäck ," Evolutionary algorithms in theory and practice", New York, Oxford Univ Press ,1996.

[18] T.E. Vollmann, W.L. Berry, and D.C. Whybark," Manufacturing planning and control system", New York. 4rd Edition, McGraw-Hill, 1997.

[19] Y. Wu, M. Liu, and C. Wu," A genetic algorithm for optimizing the MPS of a processing-assembly production line with identical machines", Proceedings of the First International Conference on Machine Learning and Cybernetics. Beijing, 2002.