Multi-Plant Multi-Product Aggregate Production Planning Using Genetic Algorithm

Fahron Dakka, Muhammad Aswin, Bambang Siswojo

Abstract— **Aggregate production planning (APP) is a mid-term planning tool used to analyze the relationship between the offer and the demand to determine the production levels to satisfy a demand that is not always completely known. It is associated with the determination of inventory, production and work force levels to consider fluctuating demand needs over a planning horizon. This paper presents a Genetic Algorithm approach for solving APP with different selection methods and crossover procedures. Combination of three selection methods and three crossover procedures are taken and compared to choose the best combination for solving APP in this present work. The problem statement depicts multi-plant, multi-product, multi-period APP with forecasted demand. The proposed approach attempts to maximize revenue as well as minimize production costs which includes labor cost, backordering cost,, inventory cost and overtime cost. Results show the outstanding performance of rank selection method and scattered crossover combination.**

Index Terms— **aggregate production planning, genetic algorithm, selection, crossover.**

I. INTRODUCTION

Aggregate Production Planning (APP) is a mid-term planning tool used to analyze the relationship between the offer and the demand to determine the production levels to satisfy demand that is not always completely known. It is associated with the determination of inventory, production and work force levels to meet fluctuating demand over a planning horizon. The planning horizon are divided into periods. Since it is usually impractical to consider every detail related to the production process while maintaining such a long planning horizon, it is required to aggregate the information being processed. The aggregate production approach is forecasted on the existence of an aggregate unit of production, such as the "average" item, or in terms of weight, volume, production time, or dollar value. Plans are based on aggregate demand for one or more aggregate items. Once the aggregate production plan is created, constraints are applied on the detailed production scheduling process, which decides the specific quantities to be produced of each individual item [1].

APP has attracted significant interest from both practitioners and academics. For solving APP problems, certain constraints are imposed which demand constraint optimization. Numerous method have been proposed to model APP. Wang and Fang [5] presented a genetics-based approach to imitate the human decision procedure for a classical product mix problem as an APP problem in a fuzzy environment. Tsoulos [3] introduced the Genetic Algorithm (GA) which is about the problem of constrained optimization and came up with improved version of genetic operators namely crossover and mutation. This improved version conserves the feasibility of the trial solutions of the constrained problem that are encoded in the chromosomes. Tavakkoli-Moghaddam and Biyabani [4] proposed a special design of a GA to work out an APP in order to minimize production costs in a real-case study of a car industry. Bunnag & Sun [5] emerged with the real coded GA based on stochastic optimization method, for solving constrained optimization problems over a compact search domain. This converges in probability to the optimal solution by treating through a repair operator.

It is obvious that there have been a long evolution phase for GA algorithms. Yet the researchers keep on this and they got newer dimension of delevelompent. Here the authors become optimistic enough after reviewing all the literatures since there are good opportunities for future contributions. Here, the authors considered single objectives for multi-plant, multi-product, multi-period APP problem. However, the distinction lies in the followed approach. We used combination of three selection methods and three crossover procedures of GA for solving multi-plant, multi-product, multi-period APP problem. A detailed comparison is also placed to choose the perfect combination of selections methods and crossover procedures. In the previous works with GA for APP, there not any single application of escalating factors for any little uncertainty or imprecise revenue. This work compare combination of three selection methods and three crossover procedures to choose the best combination for solving multi-plant, multi-product, multi-period APP problem. The proposed approach attempts to maximize revenue as well as minimize production costs in terms of inventory levels, labor levels, overtime, backordering levels, machine and warehouse capacity.

GA is a method for solving both constrained and unconstrained optimization problems based on a natural selection process that mimics biological evolution. It is heuristic method which search for the best solution to the problem. In this paper, population of 50 candidates (parents) is taken in which each candidate solution has its own set of chromosomes which can be mutated and altered and follow the biological evolution. These Chromosomes are in the array called strings which are in the form of integer number represent the production level for each product. The more fit individualsare randomly selected from the current population. Then Crossover is then applied on these strings forming new strings called offspring or the next generation. This is followed by Mutation which does random alteration just to create diversity.

Each new solution tries to achieve its best possible fitness and pass the same best solution to the new generation. This is how the new offsprings are evolved, through various iteration, with more fitness to its predecessor. This fitness is achieved by three operator: selection, crossover and mutation. The fitness is usually the value of the objective function in the optimization problem being solved. The new generation of candidate solutions are then used in the next iteration of the algorithm. Mostly, the algorithm terminates when either a maximum number of generations has been produced, or a satisfactory fitness level has been reached for the population. In other words, it is terminated when best possible fit is achieved.

II. PROBLEM FORMULATION

The multi-plant multi-product multi-period APP problem is formulated as a linear programming model. We assume that the company producing I product families manufactured at J factories to meet the market demand over a planning horizon of T periods. The solution to this APP problem is evaluated using GA which gives optimum levels of labor, inventory, backorder, overtime and regular production rates and other controllable variables. The mathematical model is created based on the characteristics of the APP problem.

The problem statement above and following notation are used after reviewing the literatures (Chakrabortty [1], Leung [9], Fahimnia [10]).

A. Notations

Indices

i : the index of the product
j : the index of the plant
t : the index of the planning period

Sets

I : the set of the products
J : the set of the plants
T : the set of the planning periods

Parameters

r_i: the unit revenue of product i ($/unit)
C_{1ij} : the unit production cost for product i manufactured from factory j by experienced workers at regular time ($/unit)
C_{2ij} : the unit production cost for product i manufactured from factory j by non-experienced workers at regular time ($/unit)
C_{3ij} : the unit production cost for product i manufactured from factory j by experienced workers at overtime ($/unit)
C_{4j} : the labour cost of experienced worker in factory j at regular time ($/man-period)
C_{5j} : the labour cost of experienced worker in factory j at overtime ($/man-hour)
C_{6j} : the labour cost of non-experienced worker in factory j ($/man-hour)
C_{7ij} : the unit inventory cost to hold a product i in factory j at the end of each period ($/unit)
C_{8ij} : the unit back-order cost for a product i in factory j at the end of each period ($/unit)
C_{9j} : the cost to hire one worker at factory j ($/man)
C_{10j} : the cost to lay off one experienced worker at factory j ($/man)
u_i: the labour time for product i by experienced worker (man-hour/unit)
v_i: the labour time for product i by non-experienced worker (man-hour/unit)
δ: the working hour of experienced worker in each period (man-hour/man-period)
α : the fraction of workforce available for overtime use in each period
ε : the fraction of workforce allowable for variation in each period
M_{jt} : the machine time capacity in factory j at period t (machine-hour)
λ_i : the machine time for product i operated by experienced worker (machine-hour/unit)
μ_i : the machine time for product i operated by non-experienced worker (machine-hour/unit)
β : the fraction of machine capacity available for overtime use in each period

Variables

S_{it} : the quantity of product i sold in period t (units)
x_{ijt} : the quantity of product i manufactured from factory j by experienced worker at regular time in period t (units)
y_{ijt} : the quantity of product i manufactured from factory j by non-experienced worker at regular time in period t (units)
z_{ijt} : the quantity of product i manufactured from factory j by experienced worker at overtime in period t (units)
WE_{jt} : the number of experienced workers required in factory j in period t (man-period)
H_{jt} : the number of experienced workers hired in factory j in period t (man-period)
L_{jt} : the number of experienced workers laid-off in factory j in period t (man-period)
TE_{jt} : the overtime of experienced worker in factory j in period t (hour)
TN_{jt} : the labour time of non-experienced worker in factory j in period t (hour)
I_{ijt} : the inventory of product i in factory j at the end of period t (units)
B_{ijt} : the back order of product i in factory j at the end of period t (units)

B. Objective functions

The aim of this study is to find an optimal APP with maximal profit by fulfilling uncertain market demand. The APP consists of production quantities by experienced workers at regular time and at overtime, and by non-experienced workers for each period of time. With the plans, decision-makers can also determine inventory level, back-order level and workforce level. Accordingly, the objective function of the proposed model is formulated as follows:

$$\sum_{i\epsilon I}\sum_{t\epsilon T} r_i \,.\, S_{it} - \sum_{i\epsilon I}\sum_{j\epsilon J}\sum_{t\epsilon T}[C_{1ij}\,.\,x_{ijt} + C_{2ij}\,.\,y_{ijt} + C_{3ij}\,.\,z_{ijt}]$$
$$-\sum_{j\epsilon J}\sum_{t\epsilon T} C_{4j}\,.\,WE_{jt} - \sum_{j\epsilon J}\sum_{t\epsilon T} C_{5j}\,.\,TE_{jt} - \sum_{j\epsilon J}\sum_{t\epsilon T} C_{6j}\,.\,TN_{jt}$$
$$-\sum_{i\epsilon I}\sum_{j\epsilon J}\sum_{t\epsilon T} C_{7ij}.I_{ijt} - \sum_{i\epsilon I}\sum_{j\epsilon J}\sum_{t\epsilon T} C_{8ij}.B_{ijt} - \sum_{j\epsilon J}\sum_{t\epsilon T}(C_{9j}.H_{jt} + C_{10j}.L_{jt})$$

This first component in expression is the total revenue based on the quantity of product sales. The second component is the production cost. The third component is the labour cost of experienced workers at regular time. The fourth component is the labour cost of experienced workers at overtime. The fifth component is the labour cost of non-experienced workers. The sixth and seventh components are the inventory cost and the back-order cost respectively. The eighth components is the cost of hiring extra workers at each period of time and the cost of laying-off redundant workers at each period of time. It is noted that no non-experienced will be recruited to work at overtime in order to maintain the quality of products.

C. Constraints

The objective functions formulated in the previous section are restricted by three sets of constraints. They are the inventory-level constraints, the relationship among the number of workers and the production capacity constraints.

The inventory-level constraints

$$\sum_{j\in J}(I_{ijt-1} - B_{ijt-1} - x_{ijt} + y_{ijt} + z_{ijt} - I_{ijt} + B_{ijt}) = S_{it} \quad i \in I, t \in T$$

$$S_{it}^{min} \le S_{it} \le S_{it}^{max} \qquad i \in I, t \in T$$

$$\sum_{i\in I} I_{ijt} \le BUF_{jt} \qquad j \in J, t \in T$$

$$\sum_{j\in J} B_{ijt} \le STK_{it} \qquad i \in I, t \in T$$

Relationship among the number of workers

$$WE_{jt} = WE_{jt-1} + H_{jt} - L_{jt} \qquad j \in J, t \in T$$

$$WE_{jt}^{min} \le WE_{jt} \le WE_{jt}^{max} \qquad j \in J, t \in T$$

$$\sum_{i\in I} u_i x_{ijt} \le \delta WE_{jt} \qquad j \in J, t \in T$$

$$\sum_{i\in I} u_i y_{ijt} \le TN_{jt} \qquad j \in J, t \in T$$

$$\sum_{i\in I} u_i z_{ijt} \le TE_{jt} \qquad j \in J, t \in T$$

$$TE_{jt} \le \delta\alpha WE_{jt} \qquad j \in J, t \in T$$

$$H_{jt} + L_{jt} \le \varepsilon WE_{jt-1} \qquad j \in J, t \in T$$

The production capacity constraints

$$\sum_{i=1}^{n} \lambda_i . x_{ijt} + \sum_{i=1}^{n} \mu_i . y_{ijt} \le M_{jt} \qquad j \in J, t \in T$$

$$\sum_{i=1}^{n} \lambda_i . z_{ijt} \le \beta M_{jt} \qquad j \in J, t \in T$$

D. Outline of Basic GA Mode

1) Generate random population of n chromosomes
2) Compute the value of the fitness function f(x) for each chromosome in the population
3) Create a new population by repeating four steps (Selection, Crossover, Mutation and Acceptation) until the new population is complete
4) Replace the old chromosomes with new ones and use this to find the generation with better fitness
5) If the stopping condition is satisfied, stop and return the best solution in current population. But if the stopping condition is not satisfied, then go to step 2 and follow the next step (loop).

E. Genetic Algorithm Parameters

1) Selection

a. **Tournament Selection** select individual from population by running a tournaments between the selected chromoshomes. The winner of each tournament is selected for crossover. Weak individual has a smaller chance to get selected in larger tournament.

b. **Roulette Selection** select individual according to their fitness value. The size of each segment in the roulette wheel is proportional to the value of the fitness of the individual. Stronger individual has a greater segment on the wheel and therefore a bigger chance to get selected [8].

c. **Rank Selection** sorts the population according to their fitness value and rank them. Then every chromosome is allocated selection probability with respect to its rank. Individuals are selected as per their selection probability [8].

2) Crossover

Crossover options specify how the GA form a new individual (child) by combining two individuals (parents) for the next generation.

a. **Single point crossover** choose a random integer x between 1 and number of variables and then selects vector entries numbered less than or equal to n from the first parent and selects vector entries numbered greater than x from the second parent. E.g: Parent P1 have [a b c d e f g h] and parent P2 have [1 2 3 4 5 6 7 8] and the randm number is 3, the function returns the following child = [a b c 4 5 6 7 8]

b. **Two Points Crossover** choose two random integer x and y between 1 and number of variables. The function selects Vector entries numbered less than or equal to x from the first parent, vector entries numbered from x+1 to y, from the second parent, vector entries numbered greater than y from the first parent. E.g.: Parent P1 have [a b c d e f] and parent P2 have [1 2 3 4 5 6] and the random value are 3 and 5. Then, child would have [a b c 4 5 f].

c. **Scattered Crossover** creates a random binary vector as a mask and selects the genes where the vector value is 1 from the first parent, and the genes where the vector value is 0 from the second parent, and combines the genes to form the child. E.g: Parent P1 have [a b c d e f] and parent P2 have [1 2 3 4 5 6], and the binary vector is [0 1 0 0 0 1]. Then, the child would have [1 b 3 4 5 f].

III. MODEL IMPLEMENTATION

A. Case Description

In this paper, we modelling and analyse the planning problem of a clothing company manufacturing a number of product types, which has manufacturing factories and sales branches located in different regions. Labor cost vary for different products and for different factories. The headquarter collects orders through its sales branch offices. The orders consist of the type of products, quantity and location preference. Decision-makers then develop an initial aggregate production plan every specified time period. In the planning process, they have to consider the manufacturing capacity, workforce level, inventory level, and other factors to fulfil forecasted demand. Based on the production plans, the factories are assigned a list of products with quantities to be produced for each period of time.

The aim of this paper is to formulate a model for aggregate production planning problems to maximize revenue as well as minimize production cost, labour cost, inventory cost, back-order cost and other relevant costs. Here, we propose the use of Genetic Algorithm as one of the approach to maximize revenue as well as minimize production cost.

The production cost and inventory cost and back-order cost for different products in each factory are shown in table 1. The labour cost, hiring cost and lay-off cost are shown in table 2. The forecast for product quantities are shown in table 3. The revenue of each product are also shown in table 4. The factories production capacities in the planning horizon are shown in table 5. The other parameters used are δ=8, α=0.2 and β=0.3. The model has 10 months planning horizon. The production cost, inventory cost and back-order cost for different products in each factory are shown in table 1. The population size, number of generations, and the number of runs which have been considered for the experimental run for the above equations are 50, 50 and 10 respectively.

Table 1. Production, inventory and back-order cost

Fact, j	Prod, i	Prod. cost of regular time	Prod. cost of overtime	Inventory cost	Back-order cost
1	1	50	140	15	30
	2	60	150	20	35
	3	80	170	30	45
	4	90	180	35	50
	5	110	200	45	60
2	1	55	145	13	30
	2	65	155	18	35
	3	85	175	28	45
	4	95	185	33	50
	5	115	205	43	60
3	1	60	150	10	30
	2	70	160	15	35
	3	90	180	25	45
	4	110	190	30	50
	5	120	210	40	60

Table 2. Labor cost

Factories, j	Labor cost of regular time	Labor cost of overtime	Hire cost	Layoff cost
1	250	10	100	120
2	225	9	90	110
3	200	8	80	100

Table 3. Forecasted demand

Period, t	Product, i				
	1	2	3	4	5
1	3000	3000	2800	2800	2600
2	4000	4000	3800	3800	3600
3	3500	3500	3300	3300	3100
4	3800	3800	3500	3500	3200
5	4000	4000	3800	3800	3600
6	3600	3600	3400	3400	3200
7	3300	3300	3200	3200	3000
8	3300	3300	3200	3200	3100
9	3000	3000	2800	2800	2600
10	3700	3700	3500	3500	3300

Table 4. Product revenue

Product, i	1	2	3	4	5
Sales Revenue	300	350	450	500	600

Table 5. Production capacities

Period, t	Factories, j		
	1	2	3
1	5000	4000	3000
2	4000	3500	3000
3	4500	3500	2500
4	5500	5000	4500
5	4000	3000	2000
6	4500	3500	2500
7	5000	4000	3000
8	4000	3500	3000
9	5000	4000	3000
10	5500	4500	3500

The purpose of this work is to check the effect of selection and crossover strategy on multi-plant multi-product APP problem. Various combinations of crossover and selection procedures are tested for the APP problem. Results are obtained for 10 runs and compared based on different statistical values like best solution, mean solution and worst solution. The population size, number of generations, and the number of runs which have been considered for the experimental run.

IV. RESULT AND FINDINGS

The proposed GA approach can solve most real-world multi-plant multi-product APP problems through an interactive decision making process. In this proposed work, 9 different combinations of crossover and selection procedures are tested and results are obtained from 10 runs for each combination. Multiple solutions are came up with the use of GA. Table 6 shows the fitness value results for all the combinations. From the results, it can be observed that combination of rank selection and scattered crossover procedure performs better than all other combinations. statistics of fitness value of all combinations are shown in table 7. Statistics values show that maximum revenue is $ 40.346.811.

Table 6 Fitness value obtained by using different combinations of selection and crossover procedures

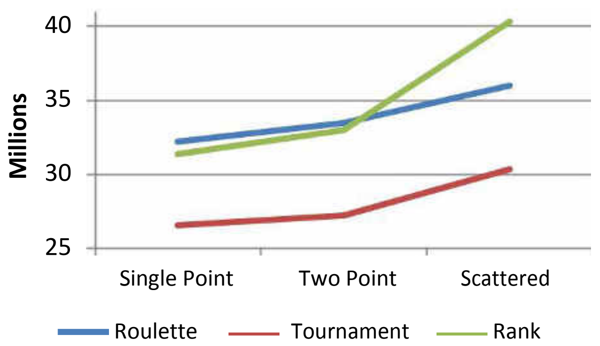
Selection	Crossover		
	Single Point	Two Point	Scattered
Roulette	32244660	33507534	40208638
Tournament	26574425	27233400	30386465
Rank	31370475	32970670	40346811

Table 7 Statistics of the fitness obtained by using best combination of selection and crossover procedures

Best	32244660
Mean	26574425
Worst	31370475

Figure 1 is plotted, to check the behavior of different combination of crossover and selection procedure on multi-plant multi product APP problem. From Figure 1 several characteristics of this proposed approach can be drawn. Rank selection procedure gives best performance when combiuned with scattered crossover. Rank selection gives best performance with scattered crossover but performs worst with single point crossover. Roulette selection gives best result when combined with scattered crossover, but getting worst with single point crossover. Tournament selection procedures are also best performs with scattered crossover.

Figure 1 Variation of fitness value by using different combinations of crossover and selection procedures



V. CONCLUSION

The proposed GA approach can solve most real-world multi-plant multi-product APP problem. This APP model can easily expanded by adding parameters, decision variables and constraints as required for practical use in industries. The modification Parameter of GA are rarely investigated for the optimization of APP problems. Hence, in this paper, combination of three different types of crossover and three different types of selection procedures are experimented to optimize multi-plant multi-product APP problem with the forecasted demand, related operating costs, production capacity and storage capacity. The results reveal best performance obtained by combining rank selection procedure with scattered crossover. Moreover scattered crossover gives near optimal fitness value with all the selection procedures. The worst performance for the APP problem among all these 9 combinations is the combination of tournament selection procedure with single point crossover.

REFERENCES

[1] R. K. Chakrabortty and Md. A. Akhtar, "Solving an aggregate production planning problem by using multi-objective genetic algorithm (MOGA) approach" in *International Journal of Industrial Engineering Computation*, vol. 4, 2013, pp. 1-12.

[2] D. Wang, and S. Fang, , "A genetics-based approach for aggregated production planning in a fuzzy environment", in *IEEE Transactions On Systems, Man, and Cybernetics*, vol. 27, 1997, pp. 636-645.

[3] I. G. Tsoulos. "Solving constrained optimization problems using a novel genetic algorithm", in *Applied Mathematics and Computation*, vol. 208, 2009, pp. 273–283.

[4] R. Tavakkoli-Moghaddam, and H. Biyabani, "The Use of Genetic Algorithms for Aggregate Production Planning", in *Proceedings of the 4th International Symposium on Intelligent Manufacturing Systems,Sakarya, Turkey, September 6-8*, 2004, pp. 683-692.

[5] D. Bunnag, & M. Sun, "Genetic algorithm for constrained global optimization in continuous variables" in *Applied Mathematics and Computation*, vol. 171, 2005, pp. 604-636.

[6] D. K. Sharma, and R. K. Jana, "Fuzzy goal programming based genetic algorithm approach to nutrient management for rice crop planning", in *International Journal of Production Economics*, vol. 121, 2009, pp. 224–232.

[7] J. E. Baker, "Adaptive selection methods for genetic algorithms", In *Proceedings of an International Conference on Genetic Algorithms and their applications*, 1985, pp 101-111.

[8] R. Kumar, "Blending Roulette Wheel Selection & Rank Selection in Genetic Algorithms", in *International Journal of Machine Learning and Computing,* vol. 2 no. 4, 2012.

[9] S. C. H. Leung, Y. Wu and K. K. Lai, "Multi-site aggregate production planning with multiple objectives: a goal programming approach", in *Production Planning & Control*, vol. 14, no. 5, July–August 2003, pp. 425–436.

[10] B. Fahimnia, L. H. S. Luong, and R. M. Marian, "Modeling and Optimization of Aggregate Production Planning - A Genetic Algorithm Approach", in *International Journal of Mechanical, Aerospace, Industrial, Mechatronic and Manufacturing Engineering*, vol. 2, no. 9, 2008.

[11] P. Savsani, G. Banthia, J. Gupta and R. Vyas, "Optimal Aggregate Production Planning by using Genetic Algorithm", in *Proceedings of the 2016 International Conference on Industrial Engineering and Operations Management*, 2016.

[12] G. Jiang, J. Kong, and G. Li, "Aggregate Production Planning Model of Production Line in Iron and Steel Enterprise Based on Genetic Algorithm", in *Proceedings of the 7th World Congress on Intelligent Control and Automation*, 2008.

[13] R. Tavakkoli-Moghaddam, "Solving a generalized aggregate production planning problem by genetic algorithms", in *Journal of Industrial Engineering International*, vol. 2, no. 1, 2006, pp. 53–64.

[14] J. H. Holland. "Adaptation in natural and artificial systems". Ann Arbor: The University of Michigan Press, 1975.

[15] S. M. Masud, & C. L. Hwang. "An aggregate production planning model and application of three multiple objective decision methods". In *International Journal of Production Research*, vol. 18, 1980, pp. 741–752.

[16] V. Gaba and A. Prashar, "Comparison of processor scheduling algorithms using Genetic Approach", in *International Journal of Advanced Research in Computer Science and Software Engineering,* vol. 2, 2012, pp. 37-45.