1

Production Planning Using Genetic Algorithm

S.K. Kumar[1] and M.K. Tiwari[2]

[1] *Wolfson School of Mechanical and Manufacturing Engineering, UK*
[2] *Indian Institute of Technology Kharagpur, India*

Abstract
Production planning and control, in manufacturing industries, generally addresses the issues of acquisition, utilization and allocation of resources to satisfy customer requirements in the most efficient and effective way. Therefore efficient management of the production function is of the utmost importance to achieve this objective. The production function includes production level, inventory level, work force level, assignment of overtime and transhipments. Mathematical models developed in this context are widely accepted and can act as decision support systems. This chapter initially focuses on developing the optimization model in different scenarios such as multi-item, multi-period, un-capacitated, capacitated, backorder, fixed and variable workforce. Improving the decision quality in those fields gives rise to complex combinatorial optimization problems, which mostly fall into the class of NP-hard problems. Finding a satisfactory solution in an acceptable time is also an important factor. Genetic Algorithms (GA) provide potent methods for solving such optimization problems and steps for GA implementation, and an example optimization problem solved by using GA, are also included in this chapter.

Keywords: Production planning, inventory, work force, genetic algorithm

1.1 Introduction

Production planning is a set of functions to undertake the efficient and effective utilization of available resources over a time horizon,

Manoj Tiwari and Jenny A. Harding (eds.) Evolutionary Computing in Advanced Manufacturing, (1–18) © Scrivener Publishing LLC

with the motivation of satisfying customer needs whilst creating profits for the organization. In manufacturing industries these functions can be defined as acquisition, utilization and allocation of resources such as raw material, man power, machines, cash flow, information etc. In today's globalized environment, with rapid technological advancement, highly customized demand, short life cycles and high competitiveness, market demand is highly volatile. Production planning therefore plays a crucial tool for any organization trying to satisfy highly customized demand and to remain competitive in the market. Poor production planning can be devastating for any organization and can lead to excessive inventory, high operational cost, poor cash flow, stock outs or lost sales and their subsequent effects. Therefore, managers must utilize their resources efficiently to avoid such consequences.

Production planning in general develops an operational plan based on the required level of manufacturing output. The plan broadly includes the raw material planning, demand management, capacity planning, work force management, inventory management, scheduling and sequencing of jobs etc., with the objective of reducing work in progress, optimizing shop floor timing, reducing inventory, managing lead time and delivery date. Production planning works across a broad domain, but this chapter will concentrate on planning of raw material, man power, inventory, production and transhipment levels. Further production planning realms will be discussed in the subsequent chapters.

The intention of this chapter is to provide an overview of developing the production planning optimization models in different scenarios and the application of genetic algorithms (GA) for solving them. There is an exhaustive range of problem contexts and solution methodologies for production planning problems, but these are not covered in detail here as this chapter concentrates more on mathematical model development in different scenarios. The next section presents various production planning models.

1.2 Production Planning Models

In general, production planning starts with a fulfilment of customer demands whilst minimizing the overall production costs. In most cases, future demand forcasting is done over a short period of time compared to the total planning horizon, in order to minimize the errors. Subsequently production planning is also divided into short time intervals to follow the forcasted demand. Demand is assumed to be deterministic, known and updated periodically.

The first mathematical model presented below has infinite manufacturing capacity i.e. the market demand is always assumed to be less than the capacity. Following are the notations used in this model:

1.2.1 Mathematical Model

A production planning problem can be formulated as follows. Suppose that I types of items are to be produced from R resources for T (1,2,3,.....T) periods. A deterministic demand D_{it} for the i^{th} item in the t^{th} period is assumed. a_{ir} and b_{rt} are the required input of resources 'r' for the production of the i^{th} item and its availability in the t^{th} period. Let C_{it} and h_{it} be the production and inventory cost. Now, the integer vectors $X_t\{x_{1t},........x_{it}\}$ and $Y_t\{y_{it},.......,y_{It}\}$ must be determined, as these represent the production and inventory quantity respectively, that will minimize the total production cost. This problem can be represented by a mathematical equation as:

1.2.1.1 Model 1: (Assuming Infinite Capacity of the Plant)

$$\text{Min} \sum_{t=1}^{T}\sum_{i=1}^{I}[X_{it}C_{it} + Y_{it}h_{it}] \tag{1.1}$$

Subject to the following constraints

$$Y_{i,t-1} + X_{it} - Y_{it} = D_{it}, \quad i = 1,2,..,I, \quad t = 1,2,3,...T \tag{1.2}$$

$$\sum_{i=1}^{I} a_{ir}X_{it} \le b_{rt} \quad \forall r,t \tag{1.3}$$

$$X_{it}, Y_{it}, D_{it} \ge 0 \quad \forall i,t \tag{1.4}$$

Equation (1.1), also called the objective function, minimizes the total cost. Constraint (1.2) ensures that demand is met. In any production system, the required resources should not exceed the available resources and this has been shown in the constraint (1.3), finally constraint (1.4) is the non-negative constraint for demand, production and inventory.

Results obtained from the above model will determine the production plan of

1. Production level of each item in each time period.
2. Inventory level of each item in each time period.

It can easily be seen that model 1 assumes that the plant has an infinite production capacity or that the demand will never exceed the production capacity (mathematically both are same). However, in realistic conditions this assumption is unfounded and it is therefore imperative that capacity is included in the model.

In most of cases, e.g. Fast Moving Commercial Goods (FMCG products), unmet demand for any time period is considered as lost sales, leading to penalty costs for the lost opportunity. Nonetheless, demand for any period can be deferred to subsequent periods and can be considered as back orders with discounts or reduced prices, indirectly increasing the production cost through penalties.

In next mathematical model we will consider the lost sale condition.

1.2.1.2 Model 2: (Assuming Lost Sale)

Extending the model 1, let MC_{it} be the maximum production capacity for the i^{th} item in the t^{th} time period. If L_{it} and δ_{it} are the penalty cost and unmet demand for the i^{th} item in the t^{th} time interval, then the objective function will be:

$$\text{Min} \sum_{t=1}^{T} \sum_{i=1}^{I} \left[C_{it} X_{it} + h_{it} Y_{it} + L_{it} \delta_{it} \right] \tag{1.5}$$

Subject to

$$Y_{it-1} + X_{it} - Y_{it} - D_{it} = \delta_{it} \quad \forall i,t \tag{1.6}$$

$$X_{it} \leq MC_{it} \tag{1.7}$$

$$X_{it}, Y_{it}, D_{it}, \delta_{it} \geq 0 \quad \forall i,t \tag{1.8}$$

and constraint (1.3).

In the above model, constraints (1.6) and (1.8) are the extensions of constraint (1.2) and (1.4), considering the lost sales. Constrai1. nt (1.7) restricts the production under its capacity. In such production planning, there is no control to prevent demand exceeding the supply, but the effect can be mitigated by proper inventory management i.e. production and storage of items in a period when demand is less than the capacity, provides a buffer for the future periods with excessive demand.

Sometimes in production planning, deferring the demand until a subsequent period can be economical (i.e. if the penalty cost is less than the extra production cost) when compared to the costs of fulfilling the demand in the same period. Such circumstances can also arise due to varying raw material cost, man-power cost or machine breakdowns. The next model developed considers that all unmet demand can be fulfilled in the future. Assuming the back order penalty cost P_{it}- for the i^{th} item for the t^{th} time interval then the objective function can be formulated as:

1.2.1.3 Model 3: (Assuming Backorders are Allowed)

$$\text{Min} \sum_{t=1}^{T} \sum_{i=1}^{I} [C_{it} X_{it} + h_{it} Y_{it} + P_{it} \delta_{it}] \tag{1.9}$$

Subject to

$$Y_{it-1} + X_{it} - Y_{it} - D_{it} - \delta_{it-1} = \delta_{it} \tag{1.10}$$

and constraint (1.7) and (1.8).

The constraints of model 2 and 3 remain the same except for constraint (1.10) which is the extension of constraint (1.6) with back orders in the flow balance equation.

1.2.1.4 Model 4: (Partial Lost Sale and Back Orders)

In the real world it is sometimes not practical to consider all lost sales as back orders. Consumers will not wait if alternatives are available in the market. This leads to a production planning scenario where a fraction of unmet demand is held as back orders. Assuming α_{it} is the fraction of unmet demand allowed as backorders for the i^{th} item for the t^{th} time period. The objective function will be:

$$\text{Min} \sum_{t=1}^{T} \sum_{i=1}^{I} [C_{it} X_{it} + h_{it} Y_{it} + \alpha_{it} P_{it} \delta_{it} + (1-\alpha_{it}) L_{it} \delta_{it}] \tag{1.11}$$

Subject to

$$Y_{it-1} + X_{it} - Y_{it} - \alpha_{it} \delta_{it-1} - D_{it} = \delta_{it} \tag{1.12}$$

$$\delta_{it} \in [0,1] \tag{1.13}$$

and constraints (1.7) and (1.8).

This model is a hybrid of models 2 and 3. If we take $\delta_{it} = 0$ (all lost sales) objective function (1.11) changes to (1.5) and with $\delta_{it} = 1$, it changes to objective function (1.9).

All the models built in the previous discussion considered capacity and resources as constraints whereas, man-power management is also a vital aspect for robust production planning and its execution. In the next two models man-power is also considered a constraint.

1.2.1.5 Model 5: (Fixed Man-Power)

The simplest way to address production planning with man-power constraints is to consider the man-power available to be fixed. If z_i and W_t are the required manpower input for the i^{th} item and its availability in the t^{th} period, then the objective function will be:

$$\text{Min} \sum_{t=1}^{T} \sum_{i=1}^{I} [C_{it}X_{it} + h_{it}Y_{it} + \alpha_{it}P_{it}\delta_{it} + (1-\alpha_{it})L_{it}\delta_{it}] \tag{1.14}$$

Subject to

$$\sum_{i=1}^{I} z_i X_{it} \leq W_t, \forall t \tag{1.15}$$

and constraints (1.7), (1.8), (1.12) and (1.13).

Constraint (1.15) ensures that the required man–power should not exceed the available man-power in each time period. Although we have added an extra constraint (15), the objective function remains the same as in model 4. The reason behind this is that manpower is considered to be fixed, i.e. a fixed cost will be incurred in each time period irrespective of the degree of utilization of manpower, therefore it becomes irrelevant for further consideration in the objective function. So the objective function will remain the same as presented in the model 4 but with one extra constraint (1.15).

1.2.1.6 Model 6: (Variable Man-Power)

In many practical cases, additional workforce may be required at some periods to cope up with unexpected or seasonal high demand. A company can usually hire temporary workers but this is likely to incur a high cost for a short period of time. In this scenario production planning needs to decide the workforce level for each period, as a variable workforce is required to cope with the unexpected or seasonal high demand.

In this model, two types of work force are considered, fixed (W_t) and variable (M_t), with H_t as the cost of hiring in the t^{th} time period. With this consideration, the objective function will be:

$$\text{Min} \sum_{t=1}^{T} \left[\sum_{i=1}^{I} \{C_{it}X_{it} + h_{it}Y_{it} + \alpha_{it}P_{it}\delta_{it} + (1-\alpha_{it})L_{it}\delta_{it}\}] + M_t H_t] \right] \tag{1.16}$$

With constraints as

$$\sum_{i=1}^{I} z_i X_{it} \leq M_t + W_t \tag{1.17}$$

$$M_t \geq 0 \tag{1.18}$$

and constraints (1.7), (1.8), (1.12), (1.13) and (1.15).

1.2.1.7 Model 7: Multiple Demand Location

All the models introduced so far are for a single production unit and a single demand location. However, it is unlikely that demand is centralized. However, in many situations a production plan is needed for multiple production units and demand locations. Therefore, a production planning model with a single production unit and multiple demand locations is now presented and subsequently, this is extended for multiple production units.

While considering multiple demand locations, production planning not only decides the level of production but also the level of supply to the market. In such scenarios it is worth considering shipment or logistics cost as this varies with demand location. Let $j\{1,2,\ldots,J\}$ be the index for the demand location and μ_{it}^{j} and χ_{it}^{j} be the logistic cost per unit and level of supply of the i^{th} item to the j^{th} location during the t^{th} period. The notation for other variables also changes from single dimension to J-dimensional.

Now the objective function will be:

$$\sum_{t=1}^{T} \left[\sum_{i=1}^{I} \left\{ C_{it}X_{it} + h_{it}Y_{it} + \left\{ \sum_{j=1}^{J} \mu_{it}^{j}\chi_{it}^{j} + \alpha_{it}^{j}P_{it}^{j}\delta_{i}^{j} \right. \right. \right.$$
$$\left. \left. \left. + (1-\alpha_{it}^{j})L_{it}^{j}\delta_{it}^{j} \right\} \right\} \right] + M_t H_t \Big] \tag{1.19}$$

Subjected to

$$\chi_{it}^{j} \le D_{it}^{j} \tag{1.20}$$

$$\delta_{it}^{j} = D_{it}^{j} - \chi_{it}^{j} - \alpha_{it}^{j}\delta_{it-1}^{j} \tag{1.21}$$

$$Y_{it-1} + X_{it} - Y_{it} = \sum_{j=1}^{J} \chi_{it}^{j} \quad \forall i,t \tag{1.22}$$

$$\{X_{it}, Y_{it}, \chi_{it}^{j}, \delta_{it}^{j}, D_{it}^{j}\} \ge 0 \quad \forall i,j,t \tag{1.23}$$

$$\alpha_{it}^{j} \in [0,1] \quad \forall i,j,t \tag{1.24}$$

and constraints (1.7), (1.17) and (1.18).

Where,

D_{it}^{j} - demand at the j^{th} location

δ_{it}^{j} - unmet demand at the j^{th} location ,

α_{it}^{j} - fraction of unmet demand changed into backorders,

P_{it}^{j} - backorder penalty cost in the market,

L_{it}^{j} - lost sales cost in the j^{th} market in the t^{th} time period.

Constraint (1.20) ensures that supply to the market should not exceed the demand. Constraint (1.21) and (1.22) are the flow balance between demand location and production unit. Constraints (1.23) and (1.24) are for non-negativity and fractional behaviour.

1.2.1.8 Model 8: Multiple Plant and Multiple Demand Locations

Let k {1,2,.....,K} be the index for plants. Now, with the objective of minimizing the total cost, production planning will decide: (i) production (X_{it}^{k}) and inventory level (Y_{it}^{k}) for items in each period. (ii) shipment level (χ_{it}^{jk}) from production units to markets. (iii) workforce (M_{t}^{k}) level for each production unit, for every period. If μ_{it}^{jk} is the per unit shipment cost then the objective function can be written as:

$$\text{Min}\left\{\sum_{t=1}^{T}\sum_{k=1}^{K}\sum_{i=1}^{I} C_{it}^{k}X_{it}^{k}\right\} + \left\{\sum_{t=1}^{T}\sum_{i=1}^{I}\sum_{j=1}^{J}\left[\sum_{k=1}^{K}\mu_{it}^{jk}\chi_{it}^{jk}\right]\right.$$
$$\left. + \alpha_{it}^{j}P_{it}^{j}\delta_{it}^{j} + (1-\alpha_{it}^{j})L_{it}^{j}\delta_{it}^{j}\right\} + \left\{\sum_{t=1}^{T}\sum_{k=1}^{K} M_{t}^{k}H_{t}^{k}\right\} \tag{1.25}$$

Subject to the following constraints:

$$Y_{it-1}^{k} + X_{it}^{k} - Y_{it}^{k} = \sum_{j=1}^{J} \chi_{it}^{jk} \;\; \forall i,k,t \tag{1.26}$$

$$\delta_{it}^{j} = D_{it}^{j} + \alpha_{it}^{j}\delta_{it-1}^{j} - \sum_{k=1}^{K} \chi_{it}^{jk} \;\; \forall i,j,t \tag{1.27}$$

$$\sum_{i=1}^{I} z_i^k X_{it}^{k} \le M_t^k + W_t^k \;\; \forall k,t \tag{1.28}$$

$$X_{it}^{k} \le MC_{it}^{k} \;\; \forall i,k,t \tag{1.29}$$

$$\sum_{k=1}^{K} \chi_{it}^{jk} \le D_{it}^{j} \tag{1.30}$$

$$\{X_{it}^{k}, Y_{it}^{k}, \chi_{it}^{jk}, \delta_{it}^{j}, D_{it}^{j}\} \ge 0 \;\; \forall i,j,k,t \tag{1.31}$$

and constraint (1.24).

As explained earlier, constraints (1.26) and (1.27) are the flow balancing equations between plants and demand locations. Constraint (1.28) is for man-power management. Demand and maximum production level are taken care of by the constraints (1.29) and (1.30) and finally constraint (1.31) is the non-negativity condition.

In this section production planning models have been developed based on various scenarios. More exhaustive models can be found in the literature discussing various factors such as, convex/concave cost functions, lead time, various transportation facilities, outsourcing etc., but these will not be discussed here. This is because the purpose of the model introduced in this chapter is to provide examples which focus on the steps required for model development and the way in which solutions can be obtained through Genetic algorithms (GA). In the next section GA is introduced as a powerful tool for solving production planning problems.

1.3 Genetic Algorithm

In the last two decades, as the complexity of the optimization problem has increased, evolutionary algorithms have gained popularity.

Evolutionary algorithms are based on the principle of evolution i.e. survival of fittest. A Genetic Algorithm (evolutionary algorithm) mimics the natural evolution (Darwin's Theorem) process that is (i) adaptation to the environment and (ii) discarding the weaker population. In the initialization step, the GA starts with some feasible solutions (called the initial population), represented in the form of a string (chromosome). Based on a fitness value (better the objective value, better the fitness), individuals undergo transformations (crossover and mutation). These individuals then strive for survival and selection as part of the next generation, survival is biased towards the fittest. Transformation using a crossover operation tries to store the best characteristics of the previous generation. It selects a pair of "parents" based on their fitness values to produce members of the next generation. Mutation is then used to change part of a solution and generate a new set of solutions. Mutation avoids the localized search. Crossover and mutation can be seen as exploitation and exploration of the solution space. Balancing between exploration and exploitation is achieved through a probability factor (which will be discussed in the next section). After a few generations, the program converges to a near optimal solution. Many variants of GA can be found in literature; however this chapter concentrates more on a simple application of GA rather than on examining these multiple variants. In the next section, an example application of a GA in the production planning domain is demonstrated.

1.3.1 Procedure of Genetic Algorithm (GA)

In order to illustrate the steps inside the GA, model 2 is examined with the number of items and time periods = 2. The first step of GA generates the initial feasible solutions in the form of a string (chromosome) as shown in Figure 1.1.

There are two decision variables (Production and Inventory) for two items and two periods, so the length of the string will be $(2 \times 2 \times 2 = 8)$ as shown above. N such solutions are generated and these are called the initial population of solutions. In the above figure, decision variables are shown in the form of ${}^{n}X_{ij}$ and ${}^{n}Y_{ij}$ with X as production and Y as inventory units, n as solution number in the population and i,j are items and time period respectively. Care should be taken while generating initial solutions to ensure that they are feasible, as in the above example the production is kept below maximum capacity level and inventory is kept below the production level. The next step of GA calculates the fitness value.

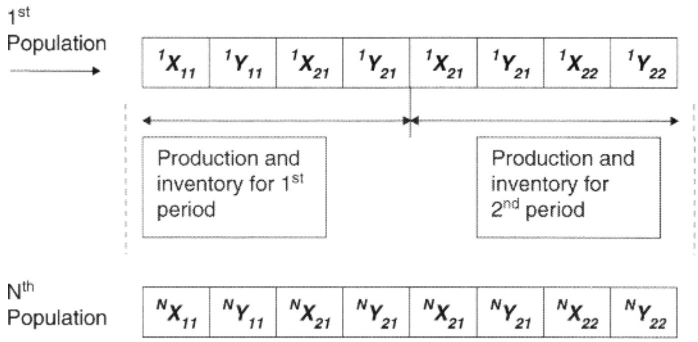

Figure 1.1 String representation in GA.

1.3.1.1 *Fitness Calculation*

The objective value is calculated for each generated population by using the equation (15). Hence, for each solution let it be f_i (for i = 1,…., N). The fitness calculation (sometimes it called fitness function) of any solution is based on this fitness value. It is clear that the better the fitness function, the better will be the solution. In case of minimization, a lower value of f_i should have superior fitness and this can be achieved by making fitness (F_i) = $1/f_i$, assuming $f_i \neq 0$. With maximization $F_i = f_i$. However, it is imperative to check the feasibility of the solution, i.e. all X and Y should strictly follow the constraint equations (1.2), (1.3) and (1.4) before the fitness calculation. Although, initial feasible solutions have been generated, in subsequent generations the application of crossover and mutation transformations may lead to infeasibility. Numerous methods have been suggested in the literature for dealing with the infeasibility and two basic methods are – (i) Penalty mechanism and (ii) Repair mechanism.

1.3.1.2 *Penalty Mechanism*

For minimization, this mechanism simply adds an extra cost in the objective function, which reduces the profit for maximization and increases the cost for minimization to reduce its fitness and probability of this solution (string or chromosome) taking part in the consequent activity (crossover). Again there are various methods by which an extra cost can be added in the objective functions. One could be (assuming minimization function): $f_i' = (1+\lambda) f_i$ with $\lambda > 0$ if the i^{th} solution violates the constraints. Another method could be, $f_i' = f_i + \lambda\theta_i$, where $\lambda > 0$ and θ_i is the degree of violation. For example, in case of checking constraint (5) in case of model 2 we can write: $\theta_i = \left|\delta_{it} - Y_{it-1} - X_{it} + Y_{it} + D_{it}\right|$. The fitness value will now be $F_i = 1/f_i'$ assuming $f_i' \neq 0$.

1.3.1.3 *Repair Mechanism*

Using a repair mechanism, the infeasible solutions are made feasible and the way of achieving this is problem dependent. For example, as discussed earlier, for model 2 there are two decision variables, X and Y. X can be infeasible if it is more than the production capacity, similarly, Y can be infeasible if it is more than X (inventory cannot be more than production). Either situation can lead to infeasibility of that particular solution. In order to make a solution feasible two steps can be followed: (i) if X is more than the production capacity, change the value of X randomly to a value which is below the production capacity. (ii) if Y is more than X, change the value of Y randomly to below X.

The next step of GA includes selection, crossover and mutation.

1.3.1.4 *Selection Procedure*

As discussed earlier, next generation solutions are obtained through crossover of the members of the present generation (called parent solutions). Selection of the parents depends on the fitness value and there are various selection procedures. One of them is probability based selection, widely known as roulette wheel selection. The roulette wheel selection procedure assigns a probability factor for each solution by the equation: probability factor

$$(p_i) = F_i \Big/ \left(\sum_{i=1}^{N} F_i) \right),$$

clearly $0 \leq p_i \leq 1$. The probability factor is higher for the solutions with higher fitness. The next step involves the defining of a selection range (S_i) for each solution.

$$S_i = \left(\sum_{j=1}^{i-1} p_j + p_i \right)$$

and clearly $0 \leq S_i \leq 1$ and $S_i = S_{i-1} + p_i$. A number R is randomly generated between 0 and 1 and if $S_{i-1} < R \leq S_i$ the i^{th} solution is selected as a parent. Then crossover takes place between a pair of parent solutions.

1.3.1.5 Crossover

In crossover, part of the solution is exchanged between the pair of parents to create the child solutions. Various procedures have been successfully implemented for the crossover such as, one-point crossover, two point crossover, multi-point crossover etc. In the following example for model 2 the single point crossover operation is used to illustrate the process. Let the i^{th} and the k^{th} solutions be selected for crossover as shown in Figure 1.2.

The crossover point is defined within the string length. In the next step the solutions are swapped after the crossover point (Figure 1.2) to generate the child solutions (Figure 1.3):

The most important factor which controls the crossover operation is the "crossover probability (P_C)". It generally defines the percentage of population which will go through the crossover i.e. if it is 0.8, then 80 percent of the population are likely to go through the crossover. To implement this in the algorithm, after selection of parents and before crossover is performed, a random number (R) is generated, such that $0 \leq R \leq 1$, and if $R \leq P_C$, crossover is performed.

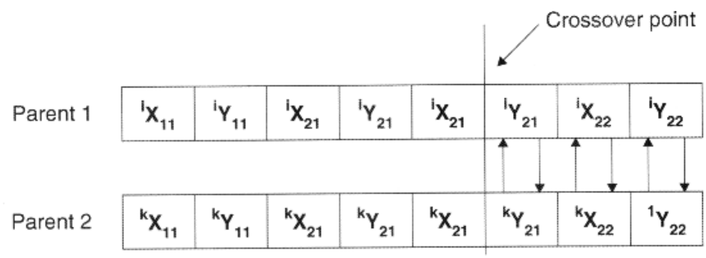

Figure 1.2 Crossover.

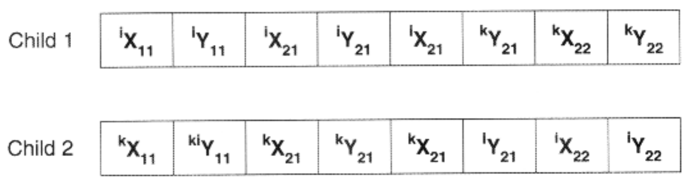

Figure 1.3 Child string after crossover.

1.3.1.6 Mutation

In the mutation operation the solution is randomly changed in order to create a new possibility of exploration of solution space. Mutation is carried out on a solution that has been shown in Figure 1.4.

Selection of the mutation point is carried out randomly and the exchanged value (in the above case A) should be feasible. Like crossover, the mutation operation is controlled by the mutation probability (P_m) and is implemented in the algorithm in a similar fashion.

Selection, crossover and mutation operations are carried out in the algorithm unless the termination criteria have been reached. Generally two conditions are adopted for termination of the algorithm: (i) when the maximum number of generations has been reached (ii) when the solution is not improving by a predefined value. The pseudocode for GA has been shown in Figure 1.5.

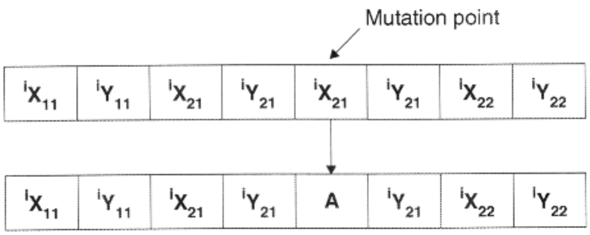

Figure 1.4 Mutation operation.

```
Procedure of GA

Begin
    t ← 0
    Initialize population
    Evaluate population
    While (termination criteria not reached) do
    t ← t + 1
    Selection of population
    Crossover
    Mutation
    Evaluate population (t)
    end
end
```

Figure 1.5 Pseudocode for GA.

1.4 Implementation of GA

To illustrate the effectiveness of GA, discussed above, a complex production planning problem is used to evaluate the computational efficiency of a GA. Assuming (Model 7) a single production plant with four items and four demand locations, with the following normal distribution for data generation is used in the example (This example uses the normal distribution in order to generate the randomized data for checking the consistency and robustness of the algorithm):

Data	Demand	Max. Prod. Cap	Prod. Cost	Inventory Cost	Trans. Cost	Back Order Coeff.	Man power Cost
Normal Dist.	N (150, 10)	N (750, 25)	N (10, 2)	N(2.5, 1)	N(5, 2)	N(0.6, 0.1)	N(7, 2.5)

1.4.1 Algorithm Parameters: (Population Size, Probability of Crossover and Mutation)

Although, there is no straightforward way to configure the GA parameters, a generic setting with population size 100, 0.8 and 0.05 as crossover and mutation probability is used as the starting point. The convergence graph obtained using single point crossover and one point mutation is shown in the Figure 1.6.

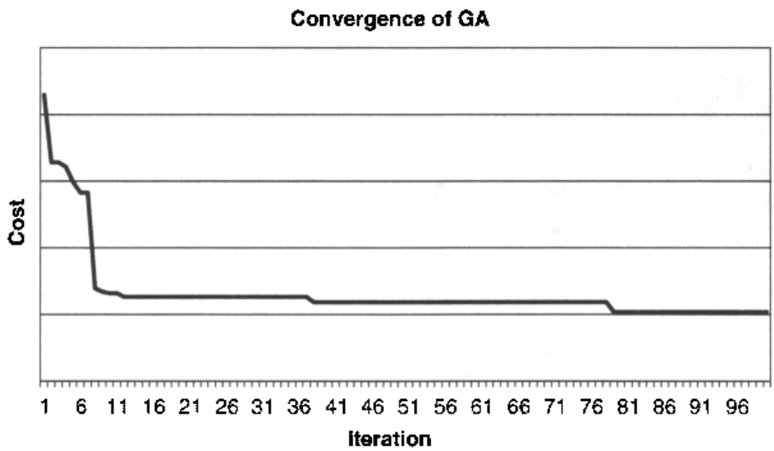

Figure 1.6 Convergence of GA.

The graph seems to be converging towards a near optimal value, exhaustive experiments are required to finalize the parameters and this is called parameter tuning.

1.4.2 Parameter Tuning

Parameter tuning requires several experiments with different configurations to obtain the best result. The effect of varying population size from 50 in steps of 50 has been studied. Results obtained are shown in the Figure 1.7. It is clear that the population size of 100 gives better and faster (less iterations are required in comparison to lower population sizes and less infeasible solutions are generated when compared to higher population sizes) results, hence, it can be inferred that a population size of 100 is sufficient to get the near optimal solution.

The next step requires the tuning of crossover (p_c) and mutation (p_m) probability. Although it is not possible to show all the experimental results, two graphs (Figures 1.8 and 1.9) where **pm** was kept as a constant (0.01 and 0.5) and varied **pc** from 0.6 to 0.8 are provided. As shown in Figures 1.8 and 1.9. The algorithm with pc 0.8 gives better results than the others. Further experiments are needed to get the best value.

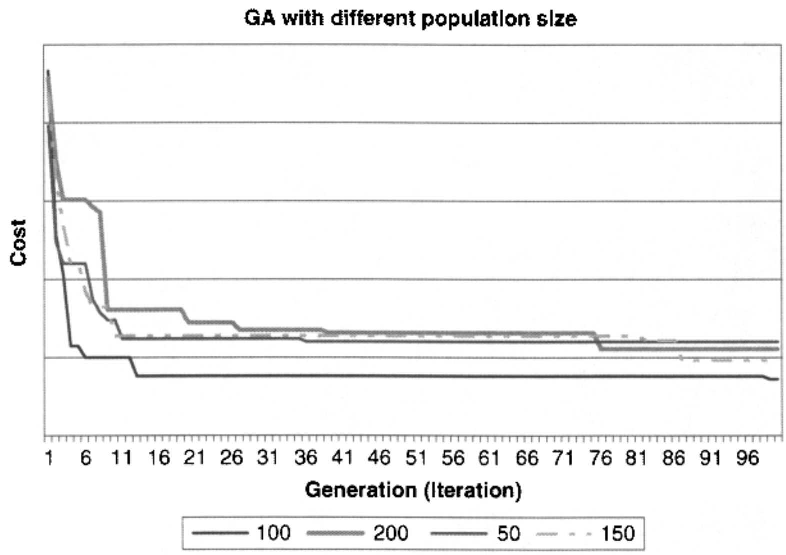

Figure 1.7 Parameter tuning with different population size.

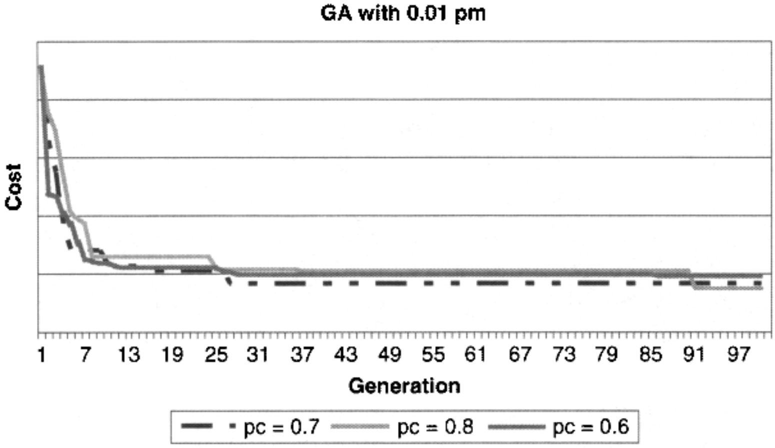

Figure 1.8 Parameter Tuning.

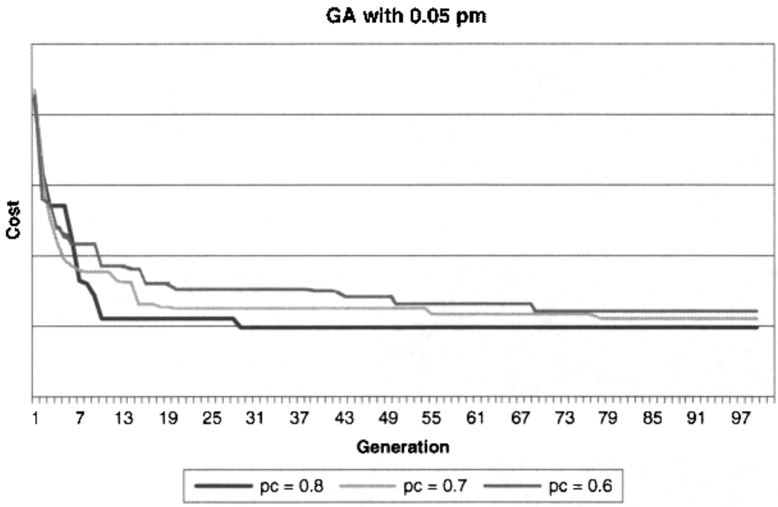

Figure 1.9 Parameter Tuning.

An important point to notice is that parameter tuning cannot be generalized and it varies from problem to problem making it a paramount step for the successful implementation of the algorithm.

1.5 Summary

In this chapter, different production planning models have been developed and the fundamentals of genetic algorithm (GA) and its successful implementation have been discussed. Although, the models developed may not cover every specific production planning consideration, nonetheless they will help readers to understand and mathematically map practical production planning problems and then successfully apply GAs to get good solutions.

Further Reading

1. Gen, M., Cheng, R., 1997. *Genetic Algorithms and Engineering Design*. John Wiley and Sons, New York, USA.
2. Goldberg, D.E., 1989. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley.
3. Mula, J., Poler, R., García-Sabater, J.P., Lario, F.C. Models for production planning under uncertainty: A review. *International Journal of Production Economics*, 103 (2006), pages- 271-285.