



Simulating the Effects of Arrival Patterns in Yield Management

P.K.Suri

(CSE/IT/MCA)

H.C.T.M., Kaithal, India

Rakesh Kumar, Pardeep Kumar Mittal

Dept. of Computer Science

& Applications, K.U.Kurukshetra, India

Abstract: *In airline yield management one of the difficult issues to be taken care of is the arrival pattern of the customers. If arrival pattern of the customers is known, the airlines industry can accordingly plan and optimize the ticket price and maximize its revenue. This paper is an attempt to optimize the yield of an airline industry depending on the various arrival patterns. Various arrival patterns such as uniform, normal and beta distribution are taken into consideration. The results obtained for the optimized yield have been identified and are summarized as the best possible combinations and worst combination.*

Keywords: *Beta distribution, Genetic algorithm, Normal distribution, Uniform distribution, Yield management.*

I. Introduction

Yield Management (YM), also known as Revenue Management, is the process of understanding, influencing and managing demand in order to maximize total revenue. It is applied on a wide variety of business areas like the airline industry, hotels, cruise ship lines, car rental companies, parking companies, theaters and radio/TV broadcasters. These companies have in common that their variable costs are relatively low compared to their fixed costs and hence an increase/loss in revenue will go directly to the bottom line. In addition, these companies sell perishable goods and their capacity is more or less fixed. Because the goods cannot be stocked and the capacity is fixed, the supply is hard to be changed and hence it makes more apply yield management.

The challenge of Yield Management is to sell the right resource to the right customer at the right time and at the right price. For example: the price of an airline ticket depends on the time when it is purchased, the date on which the flight is scheduled, the conditions (whether or not it is possible to cancel or modify the ticket), the amount of unsold seats left on the flight at the time of purchase, etc. It is likely that two people who are sitting next to each other on the same flight have paid very different prices. Hence, Yield Management can also be seen as a form of price discrimination.

In order to apply yield management, a very common method used in literature is optimization. Optimization is the selection of a best element (with regard to some criteria) from some set of available alternatives along with some constraints. An optimization problem consists of maximizing or minimizing a real function by systematically choosing input values from within an allowed set and computing the value of the function. Optimization is the process of finding best available solution of some objective function given a defined domain. To solve various optimization problems, one may use a pure mathematical approach such as LPP formulation and solution, or an algorithmic approach that terminate in a finite number of steps, or iterative approach that converge to a solution (on some specified class of problems), or heuristic approach that may provide approximate solutions to some problems. The method to solve the optimization problem depends on the problem itself.

In present paper, the heuristic approach is being followed with the help of genetic algorithm. When applying yield management in airlines industry, the historical pattern of the customers arriving to purchase a ticket becomes very important. This paper presents the possible solutions to the problem of airlines yield management on the basis on various arrival patterns of customers.

II. Literature Review

Revenue management is the collection of strategies and tactics firms use to scientifically manage demand for their products and services. The practice has grown from its origins in airlines to its status today as a mainstream business practice in a wide range of industry areas, including hospitality, energy, fashion retail, and manufacturing. An article by Kalyan T. Talluri et. al. [1] provides an introduction to this increasingly important subfield of operations research, with an emphasis on use of simulation. This article presents revenue management models for single-product, multiple-product and network models as well. Static and dynamic pricing is also taken into consideration by this article. With the increasing interest in decision support systems and the continuous advance of computer science, revenue management is a discipline which has received a great deal of interest in recent years. Although revenue management has seen many new applications throughout the years, the main focus of research continues to be the airline industry. Ever since first solution method was proposed for the airline revenue management problem, a variety of solution methods have been introduced. In this paper [2] an overview of the solution methods is presented throughout the literature.

Revenue Management has been seen by some researches as a form of price discrimination - the application of the different prices for similar products to different customer groups. By offering multiple rates to different customer groups,

firms hope to diversify and increase their revenue or retain their current revenue level. Furthermore, price management has been seen as systematic offering different prices to different customer groups in response to changes in demand and its characteristics [3]. Other examples of using price management are early booking discounts for those customers who are booking early at the company as well as offering the supplements for travellers in the on weekend. By doing so, the company can cover some major costs and can concentrate on more expensive offers (charging higher price) to receive more revenue. By doing so, it is important that a company realizes which products (or services) it can offer at profitable price for each specific customer segment. Hence, price management plays an important role for the Revenue Management systems [4]. In the certain industries the price for the products or services may influence the demand. Therefore, it is essential for all companies and establishments to adjust to these seasonal periods or price changes in order to be able to correctly predict the demand for their products or services and gain the maximum feasible revenue [5].

An advanced ticket purchasing represents the restriction for passengers who book a ticket at low price that they have to pay (as total amount) immediately or within a few hours/days. Once the price is paid and the reservation is done, there is no option to change or cancel the reservation, in such a case they have to book a new flight [6].

Lee and Hersh [7] considered a discrete-time dynamic programming model, where demand for each fare class is modelled by a nonhomogeneous Poisson process. Using a Poisson process gives rise to the use of a Markov decision model in such a way that, at any given time t , the booking requests before time t do not affect the decision to be made at time t except in the form of less available capacity. The states of the Markov decision model are only dependent on the time until the departure of the flight and on the remaining capacity. The booking period is divided into a number of decision periods. These decision periods are sufficiently small such that not more than one booking request arrives within such a period. The state of the process changes every time a decision period elapses or the available capacity changes. Lee and Hersh provide a recursive function for the total expected revenue and show that solving the model under the decision rule given by them into a booking policy that can be expressed as a set of critical values for either the remaining capacity or the time until departure. For each fare class the critical values provide either an optimal capacity level for which booking requests are no longer accepted in a given decision period, or an optimal decision period after which booking requests are no longer accepted for a given capacity level. The critical values are monotone over the fare classes. Lee and Hersh also provide an extension to their model to incorporate batch arrivals.

Subramanian et al. [8] extended the model proposed by Lee and Hersh to incorporate cancellations, no-shows and overbooking. They also consider a continuous-time arrival process as a limit to the discrete-time model by increasing the number of decision periods. Liang [9] reformulates and solves the Lee and Hersh model in continuous-time. Van Slyke and Young [10] also obtain continuous-time versions of Lee and Hersh results. They do this by simplifying the DSKP model to the more standard single leg seat inventory control problem and extending it for nonhomogeneous arrival processes. They also allow for batch arrivals. In an article presented by Srinivas and Shashi [11], the focus is on developing a decision-support tool to estimate the number of seats to each fare class. Genetic algorithm is used as a technique to solve this problem. The decision support tool considers the effect of time-dependent demand, ticket cancellations and overbooking policies.

III. Problem Definition and Formulation

In this problem an assumption regarding a flight operating between a specified origin and destination has been made. The reservation for the flight starts from the first date of expected reservation up to the date of departure. The period of reservation is considered as a single time slice. Another assumption is to fix the fare of each class and also assumed as known. Following notations has been assumed for this problem:

C_t = Total capacity of a flight

N_β = Number of customers belonging to class β .

F_β = Fare for class β .

U_β = Upper limit of demand for class β .

L_β = Lower limit of demand for class β .

For finding the objective function, the purpose of which obviously is to maximize the revenue, one have to assume some constraints, which can be:

First assumption is that there will be no cancellations and no-shows. The total number of passenger travelling should be less than or equal to the capacity of the flight. The number of customers travelling in each class should be greater than or equal to lower bound and less than or equal to the upper bound. On the basis of these assumptions, the objective function can be written as:

$$\text{Max. } \sum_\beta N_\beta F_\beta \dots\dots\dots (1)$$

Subject to the constraints

$$\sum_\beta N_\beta \leq C_t \ \& \ L_\beta \leq N_\beta \leq U_\beta \ \text{for all } \beta,$$

$N_\beta \geq 0$, which indicates that number of customers can be positive only

IV. Designing Genetic Algorithm for the Proposed Solution

The formulation specified in above section is basically a Linear Programming Problem with the variables being assumed as integers. Therefore the problem actually becomes an integer programming problem. This problem can easily be solved using any standard method to solve LPP. But the traditional LPP methods becomes complex when there is a presence of discrete integer variables. Also a large amount of input information is required such as constraints, therefore

necessitating complex modelling and simulation. Therefore instead of using the basic techniques, an attempt has been made to solve the problem using genetic algorithm.

A. Genetic Algorithm

Genetic algorithm was developed by Holland in the 1970s to understand the adaptive processes of natural systems [12]. The algorithms start from an initial population of solutions. Then, they iteratively incur the generation of a new population and the replacement of the current population. This replacement is based on selection methods. Due to the large diversity of initial populations, genetic algorithms are naturally more exploration search algorithms. This special characteristic of the GA leads us to improved solutions for the problem. Once the selection of individuals to form the parents is made, the role of the reproduction phase is the application of variation operators such as the mutation and crossover. Mutation operators are unary operators acting on a single individual. The probability P defines the probability to mutate each element (gene) of the representation. The crossover operator is binary and sometimes *n*-ary. The role of crossover operators is to inherit some characteristics of the two parents to generate the offsprings. The crossover rate represents the proportion of parents on which a crossover operator will act.

B. Chromosome Representation

To solve any problem using Genetic Algorithm, the very first issue is the selection of encoding scheme to represent the chromosomes. There are a number of encoding schemes available in literature such as binary, octal, hexadecimal, permutation and tree etc. Encoding scheme selected will further dictate what will be the crossover and mutation operators. There are two basic principles to select an encoding scheme: (i) Principle of minimal alphabet and (ii) Principle of meaningful building blocks[13]. It has been observed that in the said problem, binary encoding scheme is the base option that also satisfies both criteria.

The chromosomes represented in binary form have been generated using arrival pattern of the customers. The three different types of arrival patterns based on three probability distributions have been used. The distributions used in this problem are uniform, normal and beta distribution. A brief overview of these distributions is given below:

Uniform distribution: A Uniform random variable, X, is one whose measurements are not more highly concentrated around one value than they are around any one of the other values. The measurements of X are evenly spread over the entire range of possible values of X. The probability distribution of a uniform random variable is called a Uniform distribution. The probability density function (p.d.f.) of X is constant over a range of values of the random variable i.e.

$$f(x) = \begin{cases} \frac{1}{b-a} & \text{for } a \leq x \leq b, \\ 0 & \text{for } x < a \text{ or } x > b \end{cases}$$

Here X belongs to the set U[a, b]. The random variable X is just as likely to assume a value within one interval as it is to assume a value in any other interval of equal width. It is the probability model that is used when there is limited information about the pattern of outcomes of a random experiment or equivalently limited information about the values of the random variable. The p.d.f. of a Uniform random variable is symmetrical about $(b + a)/2$, therefore

$$E(X) = \frac{1}{2}(a + b)$$

$$V(X) = \frac{1}{12}(b - a)^2$$

Uniform distribution is also called the Rectangular distribution because of the shape of the p.d.f. as can be seen in fig.1

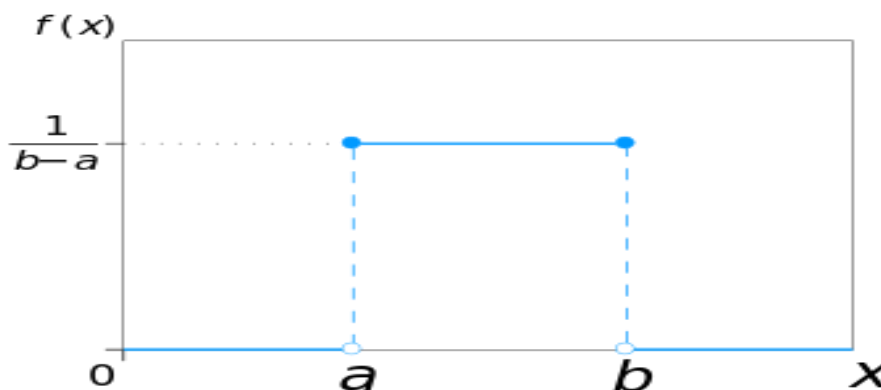


Fig.1: Probability density function for Uniform Distribution[14]

Normal Distribution: If a continuous random variable X has a Normal distribution, then we write $X \sim N(\mu, \sigma^2)$, where μ is the mean and σ^2 is the variance. The Normal distribution has the following properties: (1) It has a perfectly symmetrical p.d.f. curve, symmetrical about μ and there is no skewness. Also $P(X > m) = P(X < m) = 0.50$ since the total area under the p.d.f. curve is 1. The p.d.f. of X is as follows:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

The normal distribution can be represented as in fig.2.

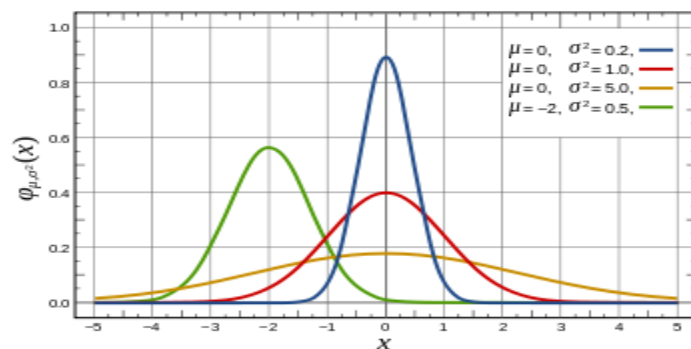


Fig.2: Probability density function for Normal Distribution[15]

Beta Distribution: In a beta distribution, the variable X is continuous and random and its probability distributions is defined on the interval $[0, 1]$ parametrized by two positive shape parameters, denoted by α and β , that appear as exponents of the random variable and control the shape of the distribution. The beta distribution has been applied to model the behavior of random variables limited to intervals of finite length in a wide variety of disciplines. For example, it has been used as a statistical description of allele frequencies in population genetics; time allocation in project management; variability of soil properties; and heterogeneity in the probability of HIV transmission.[16]

The probability density function of the beta distribution, for $0 \leq x \leq 1$, and shape parameters $\alpha > 0$ and $\beta > 0$, is a power function of the variable x and of its reflection $(1-x)$ as follows:

$$\begin{aligned}
 f(x; \alpha, \beta) &= \text{constant} \cdot x^{\alpha-1} (1-x)^{\beta-1} \\
 &= \frac{x^{\alpha-1} (1-x)^{\beta-1}}{\int_0^1 u^{\alpha-1} (1-u)^{\beta-1} du} \\
 &= \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1} (1-x)^{\beta-1} \\
 &= \frac{1}{B(\alpha, \beta)} x^{\alpha-1} (1-x)^{\beta-1}
 \end{aligned}$$

where $\Gamma(z)$ is the gamma function. The beta function, B , appears as a normalization constant to ensure that the total probability integrates to 1. In the above equations x is a realization—an observed value that actually occurred—of a random process X .

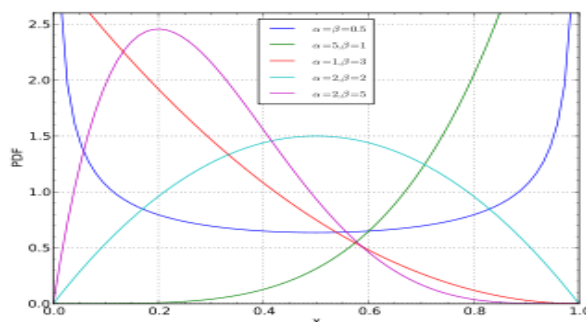


Fig.3. Probability Density Function for Beta Distribution[16]

C. Initial Population

Once the representation is fixed, the first issue in developing the algorithm is that of the initial population. This is typically performed by randomly generating individuals through the distribution for the arrival pattern, so that the population can cover wider areas of the search space.

D. Evaluation Function

The evaluation function associates a value to every individual in the population, and corresponds to the quality of that individual. The evaluation function is often referred to as fitness function in the Evolutionary Computation field. The fitness value of each individual may vary according to the arrival pattern. In present problem an evaluation function in Matlab has been designed to evaluate the chromosomes.

E. Parent Selection

Selection is the method through which certain elements in the population are chosen to be combined. This selection mechanism tries, in general, to choose parents that are likely to produce a high-quality descendant. Typically, two individuals are chosen to reproduce and yield descendants. Different kinds of selection mechanism are Roulette-Wheel, Ranking, Tournament, Multiparent, etc. In this paper, researchers have used two selection mechanisms: (i) Roulette-Wheel and (ii) Tournament selection and are explained below: Roulette-Wheel Selection: In this method, a certain

probability is to be chosen for every individual in the population. This probability depends directly on the absolute fitness of the individual [12].

Tournament Selection: It consists of choosing k individuals completely at random, and then selecting the two individuals with highest fitness function. Obviously, the complexity of this method depends on the value of k .

F. Reproduction

For the purpose of reproduction, the parents(individuals) are combined in such a way that a high quality individual (descendant) will be obtained. This mechanism is also known as crossover. The different crossover operators are one point, multiple point, uniform, arithmetic, etc. In present case, one point, two point and uniform crossover has been used, and are explained as follows:

One point crossover: It consists of choosing a point randomly, and copying the genes of a parent, from the beginning until this point, to the descendant, and the genes of the other parent from that point till the end.

As an example, imagine two parents of the form:

First = (0 1 0 0 1 1 1 1 0), Second = (0 0 0 1 1 0 1 0 0)

and $k = 5$ is the crossover point, the descendant would either be

(0 1 0 0 1 / 0 1 0 0) or (0 0 0 1 1 / 1 1 1 0)

Two point crossover: In this method the difference from the previous method is that instead of one point, two points are chosen randomly. Then, to generate a descendant it would copy the genes of each parent in turns after each crossover point.

Uniform crossover: It treats each gene independently and decides from which parent it is going to be inherited (typically with the same probability).

G. Mutation

This operator is the source of great diversity. It is based in the biological fact that some genes can mutate for different reasons, and thus, the descendant can acquire genes that are from neither of its parents. The most common ones are random bit modification, swap mutation, insert mutation, scramble mutation, etc. In the present case, the random bit modification has been used which is explained below:

Random bit modification: Consists on changing the value of some bits with a given probability. The operator changes the value of every bit in the sequence with a certain probability. If the representation is binary as in our case, the effect is that of flipping a bit, either from 0 to 1 or from 1 to 0.

H. Selection of the New Generation

This is the mechanism that replaces the last population by a new one. In order to do so, some algorithms completely replace the previous population for the new set of descendants or offspring. However, this is usually not a very effective technique, and GAs normally implements mechanism to generate the new population from both, the previous one and the offspring. Some of these mechanisms are fitness based, generation based, replace worst, etc. In this case the fitness based replacement has been used i.e. selection focuses on keeping the individuals with higher fitness for the next generation. A technique associated with this operator is to always maintain the highest quality individual in the population. This technique is usually referred to as elitism.

I. Termination

The termination condition indicates when it is time for the algorithm to stop. At this point, the algorithm will usually return the best individual according to its fitness function. One can distinguish two kinds of termination condition:

Objective reached: when a GA is implemented to reach a certain goal (i.e., a solution of a certain quality), reaching that goal should be the indication for the algorithm to stop.

External conditions: However, the previous case is very rarely achieved, due to the stochastic nature of these algorithms. Therefore, different criteria must be used. Different conditions include fixed number of generations reached, maximum time allowed reached, fitness improvement does not occur for a certain period of time/generations, manual inspection, a combination of the above. In present case, researchers have maintained a fixed number of iterations.

Genetic Simulator: Single-Leg Flight Yield Management

```
1. function GeneticSimulator()
2. pop ← InitialPopulation(); // Initial population has been generated using uniform, normal, or beta distribution
3. Evaluate(pop); // Fitness function
4. while not Termination() // Termination criteria which is fixed number of iterations
5. parents ← selectParents(pop); // Selection based on Roulette-Wheel or Tournament Selection
6. descendants ← Combine(parents); // One-point, two-point or uniform crossovers
7. Mutate(descendants); // Mutation with probability 0.03
8. pop ← SelectPopulation(pop; descendants); // Fitness based replacement
9. end while
```

V. Results

Genetic algorithm is used as a solution technique for the single-leg airlines problem using various combinations of different operators. In this case, a single flight is considered to operate between given origin and destination. The capacity of the flight is assumed to be 100. The following GA parameters are taken into considerations:

(i) Population size = 75 (ii) Maximum number of iterations = 50 (iii) Cross-over probability = 0.80

(iv) Mutation probability = 0.03 (v) Tournament Selection parameter = 0.75 (vi) Number of simulations = 40
Using the above parameters and various combinations one can get the following tables:

TABLE 1: NUMBER OF ITERATIONS FOR THE BEST ESTIMATES IN VARIOUS COMBINATIONS OF ARRIVAL PATTERN(DISTRIBUTION), SELECTION AND CROSS-OVER PARAMETERS

Parameters			Iterations		No. of simulation when max. is not achieved in specified iterations
Arrival Pattern (Distribution)	Selection	Cross-over	Min. in all simulations	Avg. of all simulation except when max is not achieved	
Uniform	Roulette Wheel	One-Point	2	16	10
Uniform	Roulette Wheel	Two-Point	18	27	36
Uniform	Roulette Wheel	Uniform	3	23	24
Uniform	Tournament	One-Point	2	22	4
Uniform	Tournament	Two-Point	8	25	14
Uniform	Tournament	Uniform	7	14	16
Normal	Roulette Wheel	One-Point	4	27	16
Normal	Roulette Wheel	Two-Point	29	29	38
Normal	Roulette Wheel	Uniform	38	43	36
Normal	Tournament	One-Point	17	32	10
Normal	Tournament	Two-Point	13	22	32
Normal	Tournament	Uniform	14	28	20
Beta	Roulette Wheel	One-Point	5	22	8
Beta	Roulette Wheel	Two-Point	20	29	30
Beta	Roulette Wheel	Uniform	2	30	30
Beta	Tournament	One-Point	2	20	12
Beta	Tournament	Two-Point	12	24	16
Beta	Tournament	Uniform	8	22	8

TABLE 2: LOWER, UPPER AND BEST ESTIMATED DEMANDS IN EACH ASSUMED FARE CLASS

Fare Class	Fare	Demand		Best Estimation
		Lower Limit	Upper Limit	
1	100	0	63	30
2	250	30	45	45
3	500	13	20	20
4	800	2	5	5

Results obtained for each combination of operators are shown below:

- Using combination of Roulette Wheel selection and one-point crossover, one can get following results for the various arrival patterns(distribution):

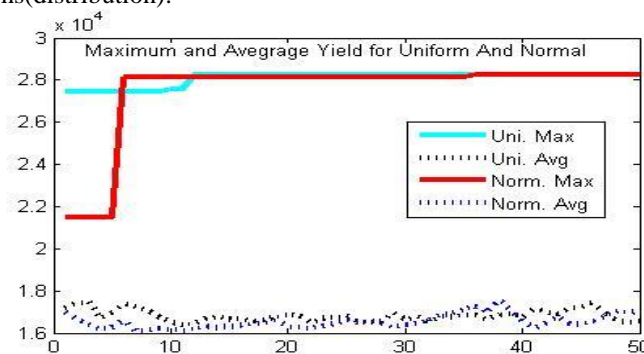


Fig.4 : Uniform and Normal distributions as arrival patterns

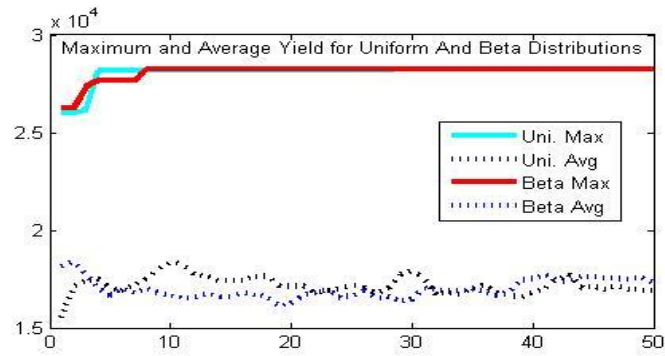


Fig.5: Uniform and Beta distribution as arrival patterns

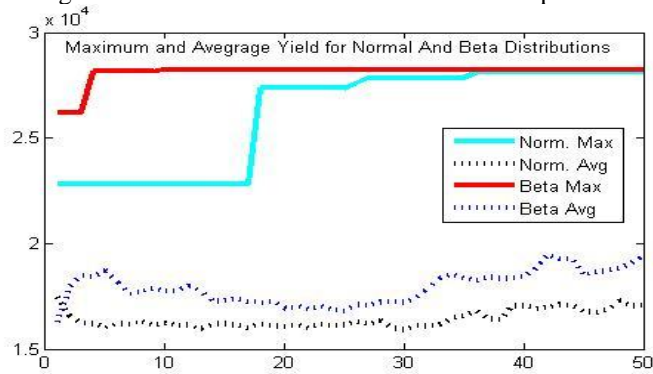


Fig 6: Normal and Beta distribution as arrival patterns

- Using combination of Roulette-Wheel selection and the two-point crossover, following results for the various arrival patterns(distributions) have been observed:

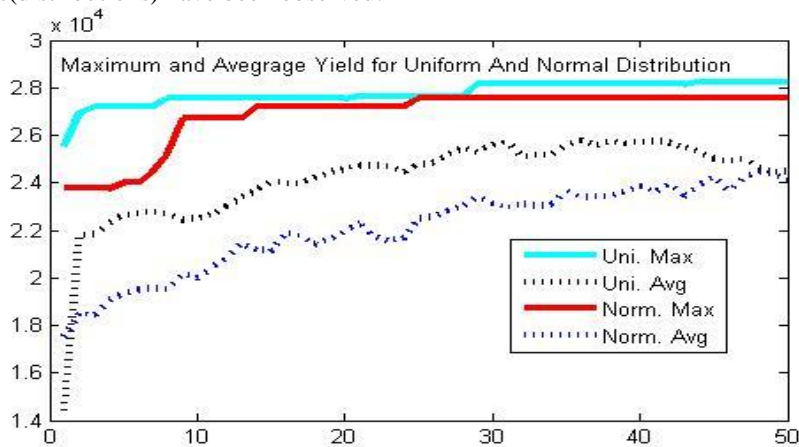


Fig.7 : Uniform and Normal distributions as arrival patterns

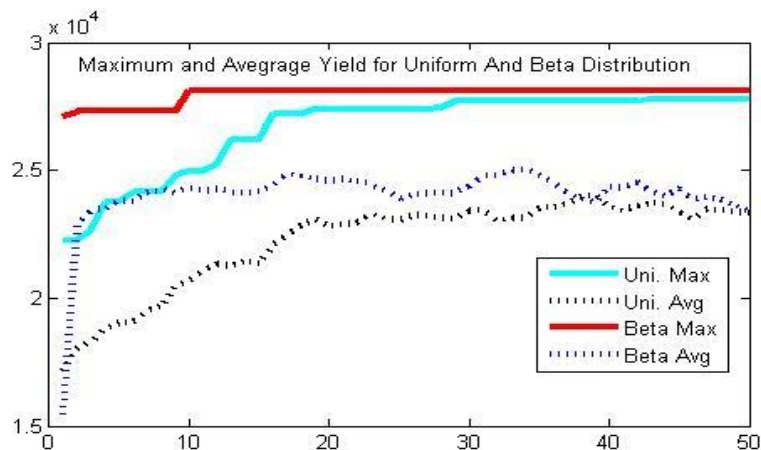


Fig.8: Uniform and Beta distribution as arrival patterns

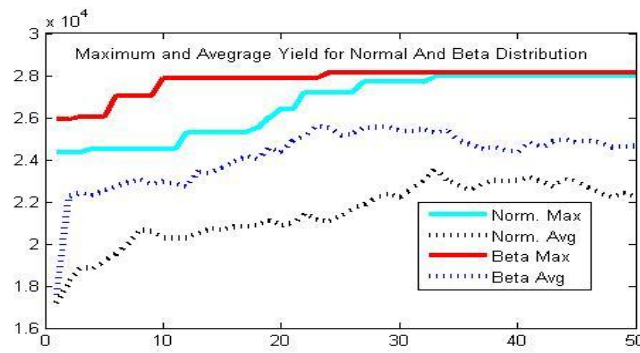


Fig 9: Normal and Beta distributions as arrival patterns

- Using combination of Roulette Wheel along with uniform crossover, one can get following results for the various arrival patterns(distributions):

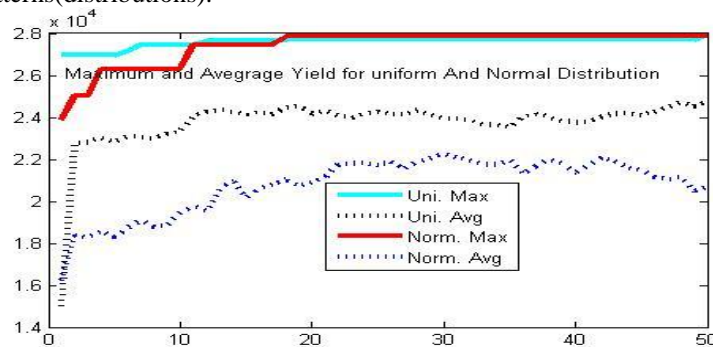


Fig.10 : Uniform and Normal Distribution

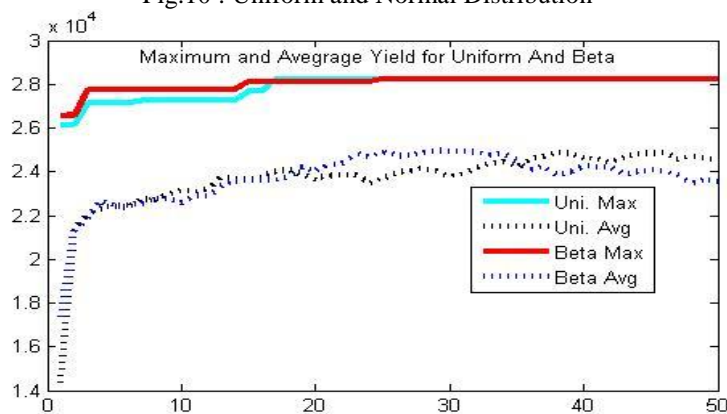


Fig.11 : Uniform and Beta Distribution

- Using combination of Tournament selection along with one-point crossover, following results have been observed on the basis of various arrival patterns:

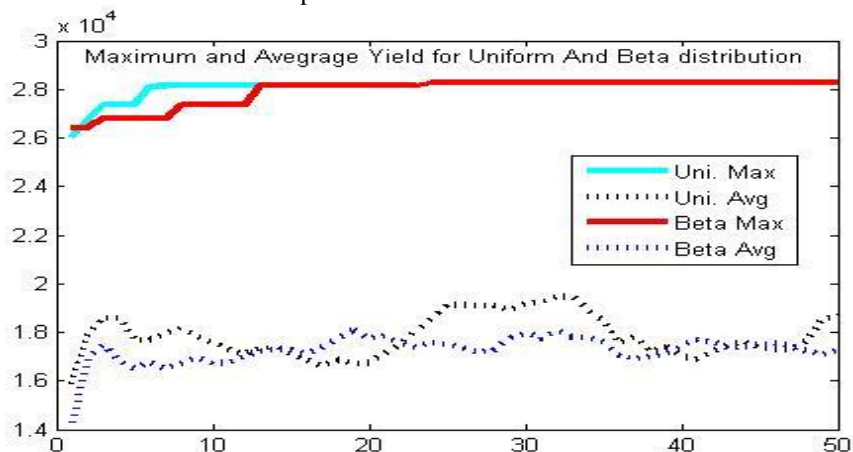


Fig.12 : Uniform and Normal Distribution

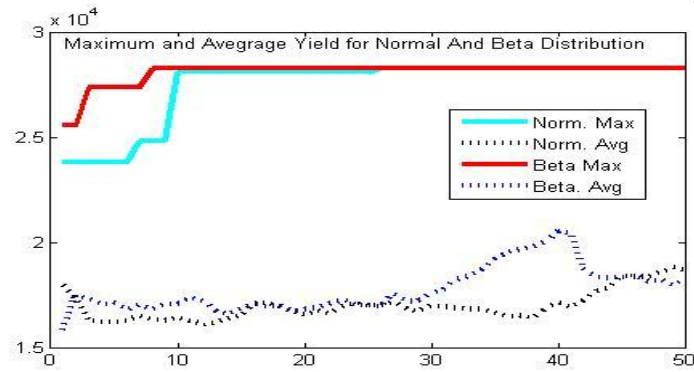


Fig.13: Normal and Beta Distribution

- Using combination of Tournament selection along with two-point crossover, one can obtain the following results on the basis of various arrival patterns:

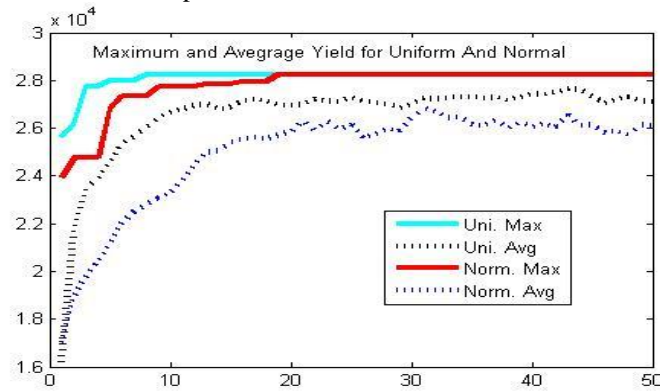


Fig.14 : Uniform and Normal Distribution

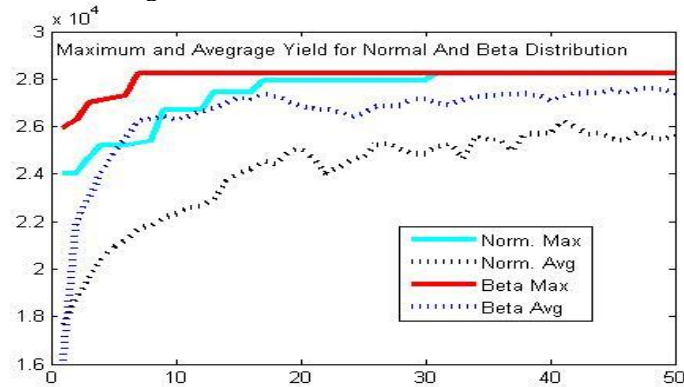


Fig.15 : Normal and Beta Distribution

- Using combination of Tournament selection along with uniform crossover, one can obtain the following results on the basis of various arrival patterns:

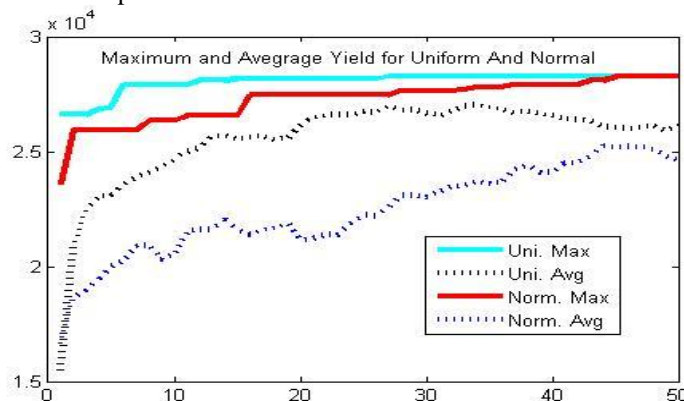


Fig.16: Uniform and Normal Distribution

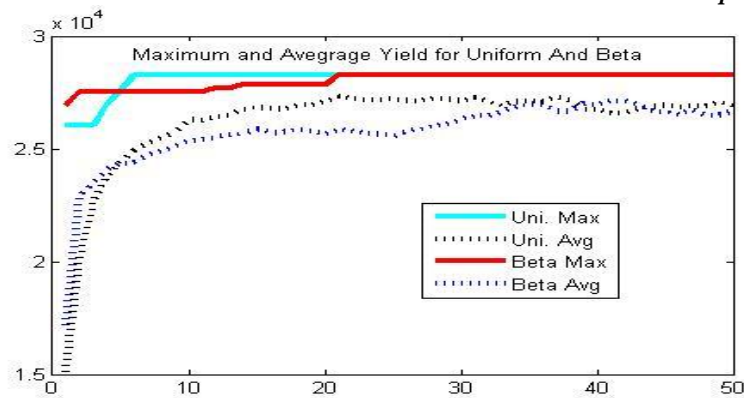


Fig.17: Uniform and Beta Distribution

VI. Interpretation

Upon comparing the above results with [16], it has been observed that the results obtained are at par with [16] and the results obtained using GA are optimal. However the major emphasis of this paper is on performance of GA on the basis of various arrival patterns along with what are the best possible combination for each type of arrival pattern. Looking at the tables and graphs obtained by different combinations of various arrival patterns(distributions), selection and cross-over methods one can interpret the following:

1. The optimized results are obtained quite early in most of the cases when roulette-wheel selection is used along with one-point crossover operators irrespective of the arrival pattern.
2. In case of uniform and beta distribution as an arrival pattern, the optimized results are obtained quite early in almost all the cases, while in normal distribution the optimized results arrive in more number of iterations except in case of roulette-wheel selection and one-point crossover.
3. The average number of iteration to obtain the optimized result are lesser in case of uniform distribution as compared to beta and normal distribution.
4. Two-point crossover along with roulette-wheel selection should be completely discarded in every type of arrival pattern, as in most of the cases optimized results are not obtained in specified number of iterations. The uniform crossover along with roulette-wheel selection should also be discarded for the same reasons, however it may be used in case of uniform distribution as an arrival pattern.
5. The best and worst cases in each of the arrival patterns(distributions) can be summarized in the following table:

TABLE 3: BEST AND WORST COMBINATION OF OPERATORS FOR VARIOUS ARRIVAL PATTERNS (DISTRIBUTIONS)

Arrival Pattern(Distribution)	Best Possible Combination(s)	Worst Combination(s)
Uniform	(i) Roulette-Wheel Selection along with One-Point Crossover (ii) Tournament Selection along with One-Point Crossover	Roulette-Wheel Selection along with Two-Point Crossover
Normal	Roulette-Wheel Selection along with One-Point Crossover	(i) Roulette-Wheel Selection along with Two-Point Crossover (ii) Roulette-Wheel Selection along with Uniform Crossover
Beta	(i) Roulette-Wheel Selection along with One-Point Crossover (ii) Tournament Selection along with One-Point Crossover (iii) Tournament Selection along with Uniform Crossover	Roulette-Wheel Selection along with Two-Point Crossover

VII. Conclusion

In this paper the researcher has tried to identify the best possible use of genetic algorithm in yield management. The particular case study which has been picked is airlines yield management. The various arrival patterns such as uniform, normal and beta distribution were taken into consideration. The various combinations of different operators were tried in an attempt to find the best possible combination for maximizing the profit for airlines. The combinations which proved to be best depend on the arrival pattern of customers. If somehow arrival pattern can be identified using historical data of customers, this paper can prove to be very useful for finding the optimized results in a better way. The worst combination for each of the arrival pattern has also been identified so that they are not used for the purpose of optimizing the yield.

VIII. Future Scope

In this paper a basic case of yield management in airlines has been taken into consideration with various arrival patterns. The results obtained although can prove to be useful for airlines industry; still there are a number of things that can be considered for practical implementation. Some of them can more than one decision period, overbooking and cancellation of customers, etc. The concept of genetic algorithm can also be applied to other industry such as hotel industry, sea-cargo industry, and even in cloud computing scheduling, operating system scheduling, etc.

References

- [1] Kalyan T. Talluri, Garrett J. van Ryzin, Itir Z. Karaesman, Gustavo J. Vulcano(2009), *Revenue Management : Models and Methods*, Proceedings of the Winter Simulation Conference, IEEE
- [2] Kevin Pak, Nanda Piersma(2002), *Airline Revenue Management*, ERIM Report Series Reference No. ERS-2002-12-LIS
- [3] Anthony Ingold, Una McMahon-Beattie and Ian Yeoman: *Yield Management strategies for the service industries* , second edition 2000, Continuum Books
- [4] Robert L. Phillips: *Pricing and Revenue optimization*, 2005, Stanford University Press , Stanford, California
- [5] Robert G Cross, *Revenue Management: Hard-Core Tactics for Market Domination* ,1997, Broadway Books, New York
- [6] Stephen Shaw, *Airline marketing and Management*, fifth edition, Ashgate publishing Limited, 2004
- [7] Lee, T.C. and Hersh, M. (1993), *A Model for Dynamic Airline Seat Inventory Control with Multiple Seat Bookings*, *Transportation Science*, 27, 252-265.
- [8] Subramanian, J., Stidham jr., S. and Lautenbacher, C.J. (1999), *Airline Yield Management with Overbooking, Cancellations, and No-Shows*, *Transportation Science*, 33, 147-167.
- [9] Liang, Y. (1999), *Solution to the Continuous Time Dynamic Yield Management Model*, *Transportation Science*, 33, 117-123.
- [10] Van Slyke, R. and Young, Y. (2000), *Finite Horizon Stochastic Knapsacks with Applications to Yield Management*, *Operations Research*, 48, 155-172.
- [11] Srinivas S. Pulugurtha & Shashi S. Nambisan, *A Decision-Support Tool for Airline Yield Management Using Genetic Algorithm*, *Computer Aided Civil and Infrastructure Engineering*, 2003, 214-233.
- [12] J.H. Holland, *Adaption in Natural and Artificial Systems*, MIT Press, 1975.
- [13] S.N.Sivanandam, S.N.Deepa, *Introduction to Genetic Algorithm*, Springer-Verlag Berlin Heidelberg 2008
- [14] [http://en.wikipedia.org/wiki/Uniform_distribution_\(continuous\)](http://en.wikipedia.org/wiki/Uniform_distribution_(continuous))
- [15] http://en.wikipedia.org/wiki/Normal_distribution
- [16] http://en.wikipedia.org/wiki/Beta_distribution