Simulating the Multiple Time-Period Arrival in Yield Management

P.K.Suri[#1], Rakesh Kumar[#2], Pardeep Kumar Mittal[#3]

[#1]Dean(R&D), Chairman & Professor(CSE/IT/MCA), H.C.T.M., Kaithal(Haryana), India
[#2]Professor, Department of Computer Science & Applications, Kurukshetra University, Kurukshetra(Haryana), India
[#3]Assistant Professor, Department of Computer Science & Applications, Kurukshetra University, Kurukshetra(Haryana), India

Abstract*- Yield management is such type of problem that depends a lot on time i.e. time of selling is very important. Therefore instead of selling the inventories at one go, it should be divided in some time periods to maximize revenue. This paper aims at capacity allocation during multiple time periods to obtain optimum revenue. The implementation has been done using two techniques i.e. LPP simulator and genetic algorithm. The genetic algorithm has been implemented using Matlab while LPP simulator has been implemented using C. Results in both cases are on expected lines.

***Keywords*- Genetic Algorithm, LPP Simulator, Multiple Time-Period, Yield Management**

I. INTRODUCTION

The work presented in this paper is basically an extension of an earlier work by Mittal et.al [1]. In that paper a strategy was identified for allocating the optimum number of seats for each type of fare class while booking. A specific example for the airlines was considered. The parameter that was taken into account was only fare classes.

In the present work authors intend to add another parameter in terms of time of booking. The time of booking is divided into multiple time-periods and this paper tries to find how many seats must be allocated in each time-period so that total revenue obtained in optimum. Again another specific example involving airlines is taken into consideration. In this specific example total duration of booking is divided into two time-slices although it can be extended to any number of time-slices using the same technique presented here.

The work that has been done in this paper is implementing a simulator for mathematical optimization using linear programming and using genetic algorithm.

The paper has been divided into eight sections including present one, the section 2 provides related literature, while in section 3 problem has been defined and formulated. Section 4 implements the problem using LPP simulation and section 5 implements the problem using genetic algorithm. In section 6 results are presented and section 7 interprets the results. In the last section conclusion has been provided.

II. LITERATURE REVIEW

The airline industry provided researchers with a concrete example of the tremendous impact that Revenue Management tools can have on the operations of a company (e.g. [2]). During the 90’s, the increasing interest in Revenue Management become evident in the different applications that were considered. Models now are having a higher degree of complexity (e.g. multi-class and multiperiod stochastic formulations).

In order to maximize the revenue of airline, an optimized flight booking and transportation terminal open/close decision system has been presented using Genetic Algorithm by George A. et.al. [3]. In that system, the particular booking terminal’s historical booking data was observed. Consequently, its frequency is generated with linguistic variable and deviation of booking is interpreted. Using the observed data and genetic algorithm, the terminal open/close decision system is optimized.

In an article by Pulugurtha & Nambisan [4], a decision-support tool is developed 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.

Effect of arrival patterns has been studied on yield management using GA by Suri et.al [5]. This paper has taken into account the various arrival patterns and their impact on YM has been discussed.

In a paper by Maglaras and Meissner [6], analysis of dynamic pricing and capacity control problem has been done through a rigorous numerical approach.

Bitran and Caldentay [7] in their survey paper examined the research and results of dynamic pricing policies and their relation to Revenue Management. In this paper a capacity control stochastic problem has been formulated and examined. A pricing-inventory model with multiple interdependent products and stochastic demand is implemented with the help of GA by Ganji and Shavandi [8]. Models with additive and multiplicative demand uncertainty were developed. Then a genetic algorithm was used to solve the models. Results were good and satisfying.

III. PROBLEM DEFINITION AND FORMULATION

The problem considered in this paper is same as [1] except multiple time-periods will be considered in this work. Therefore the assumptions remain same and are defined as follows:

In this problem an assumption regarding a flight operating between a specified origin and destination has been made. The reservation for the flight starts form the first date of expected reservation up to the date of departure. The period of reservation is divided more than one time slices. Another assumption is to fix the fare of each class during each time slice and also assumed as known.

For finding the objective function, the purpose of which Following notations has been assumed for this problem:

C_t = Total capacity of a flight

$N_{\alpha,\beta}$ = Number of customers belonging to class β during time slice α.

F_β = Fare for class β.

$U_{\alpha,\beta}$ = Upper limit of demand for class β during time slice α.

$L_{\alpha,\beta}$ = Lower limit of demand for class β during time slice α.

The purpose is obviously is to maximize the revenue. For this, one has 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 above assumptions, the objective function can be written as:

Max. $\Sigma_\beta \Sigma_\alpha N_{\alpha,\beta} F_\beta$ ……………… (1)

Subject to the constraints

$\Sigma_\beta \Sigma_\alpha N_{\alpha,\beta} <= C_t$ & $L_{\alpha,\beta} <= N_{\alpha,\beta} <= U_{\alpha,\beta}$ for all α and β,

$N_{\alpha,\beta} >= 0$, which indicates that number of customers in each time slice can be positive only

IV. Implementation using LPP Simulator

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 be solved using a LPP simulator, which is explained below:

LPP Simulator for Solving YM problem

1. Input the number of variables in the objective function, equalities and inequalities.
2. Input objective functions and constraints.
3. Convert each inequality in the set of constraints to an equation by adding slack & surplus variables.
4. Create the initial simplex tableau.
5. Locate the most negative entry in the bottom row. The column for this entry is called the entering column. (If ties occur, any of the tied entries can be used to determine the entering column.)
6. Form the ratios of the entries in the "b-column" with their corresponding positive entries in the entering column (If all entries in the entering column are 0 or negative, then there is no maximum solution. For ties, choose either entry.) The entry in the departing row and the entering column is called the pivot. The departing row corresponds to the smallest non-negative ratio b_{ij}/a_{ij}
7. Use elementary row operations so that the pivot is 1, and all other entries in the entering column are 0. This process is called pivoting.
8. If all entries in the bottom row are zero or positive, this is the final tableau. If not, go back to Step 5.
9. If a final tableau is obtained, then the linear programming problem has a maximum solution, which is given by the entry in the lower-right corner of the tableau, otherwise solution cannot be achieved by this method.

The above simulator was implemented using C language and a specific example was formulated for verifying the results and working of the simulator.

But the traditional LPP methods becomes complex when there is a presence of discrete integer variables. Also there is large amount of input information is required such as constraints, therefore necessitating complex modelling and simulation. Therefore along with the use of the above basic technique, an attempt has also been made to solve the problem using genetic algorithm.

V. Implementation using Genetic Algorithm

The basic advantage of using GA lies in the fact that it does not require gradient and derivative information and thus can be used to solve even complex real world problems with discontinuous functions. [8].

For solving the above formulated problem, a genetic algorithm has been implemented using Matlab and is stated below:

Genetic Algorithm for Multiple Time-Period Yield Management

1. Init_pop = Randomly Generated population.
2. curr_pop = Init_pop.
3. While (!termination_criterion)
4. Evaluate Fitness of curr_pop using fitness function.
5. Select mating pool according to Roulette-wheel Selection OR Tournament Selection.
6. Apply Crossovers like One-point, Two-point & Uniform Crossovers on mating pool with probability 0.80.
7. Apply Mutation on mating pool with probability 0.03.
8. Replace generation with $(\lambda + \mu)$-update as curr_pop.
9. End While
10. End

VI. Results

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 result obtained after implementing LPP simulator using a specific example is shown in table 1.

Genetic algorithm is used as a second solution technique using various combinations of different operators. The following GA parameters are taken into considerations:

Population size = 75

Maximum number of iterations = 200
Cross-over probability = 0.80
Mutation probability = 0.03
Tournament Selection parameter = 0.75
Number of simulations = 40

Using the above parameters and various combinations one can get the tables 2,3 and 4, which are shown at the end of the paper.

Results obtained for each combination of operators are shown below:

- Using combination of Roulette Wheel selection and the three types of cross-overs i.e. one-point, two-point and uniform, the following results were obtained during first time period:

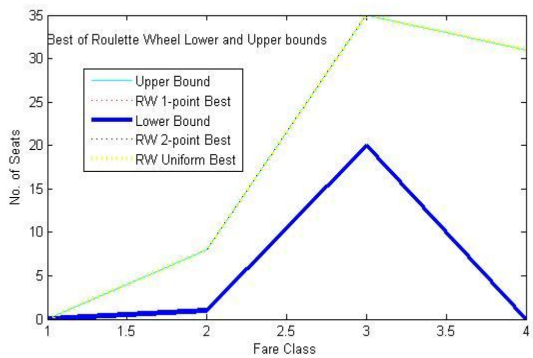

Fig.1 : Lower Bound, Upper Bound, and Estimated Fitness

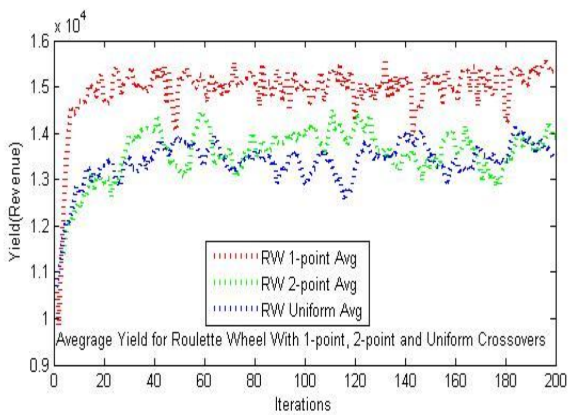

Fig.2: Average Fitness

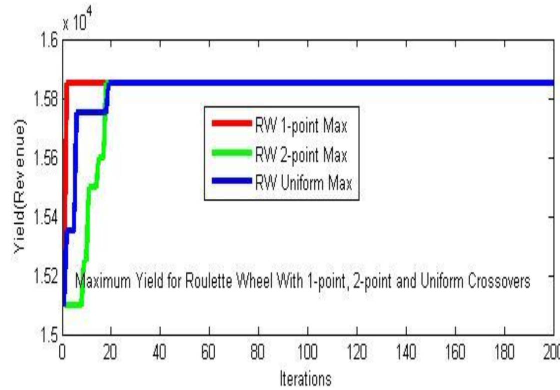

Fig 3: Maximum Fitness

- Using combination of Roulette Wheel selection and the three types of cross-overs i.e. one-point, two-point and uniform, the following results were obtained during second time period:

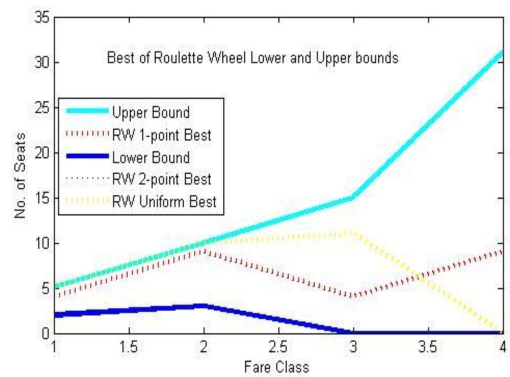

Fig.4 : Lower Bound, Upper Bound, and Estimated Fitness

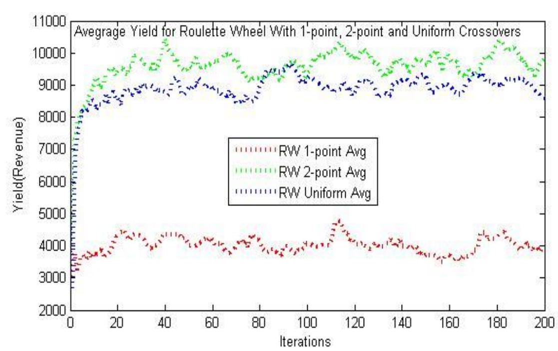

Fig.5: Average Fitness

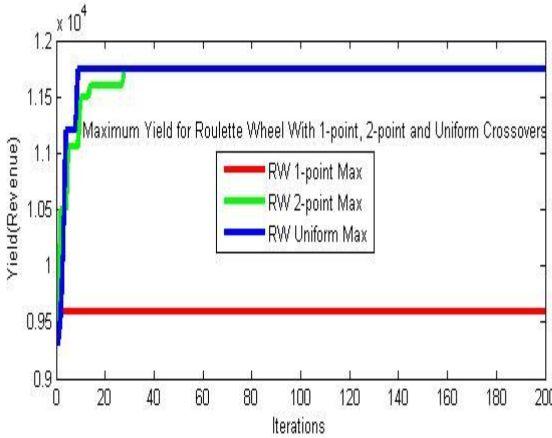

Fig 6: Maximum Fitness

- Using combination of Tournament selection and the three types of cross-overs i.e. one-point, two-point and uniform, the following results were obtained during first time period:

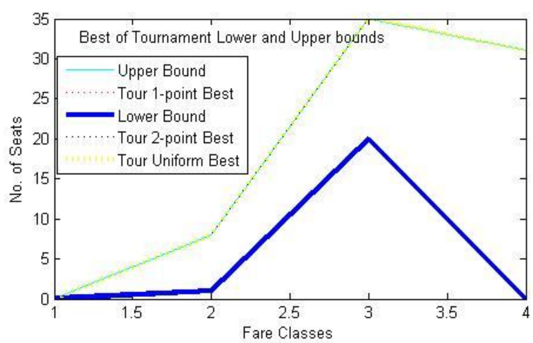

Fig.7 : Lower Bound, Upper Bound, and Estimated Fitness

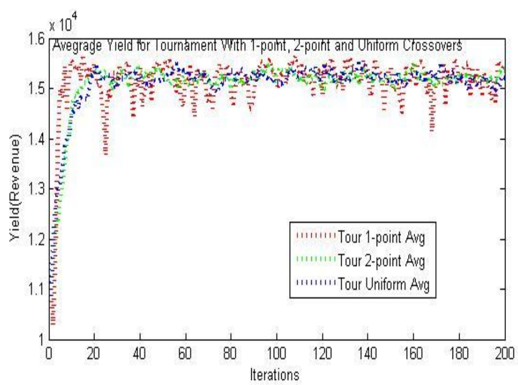

Fig.8 : Average Fitness

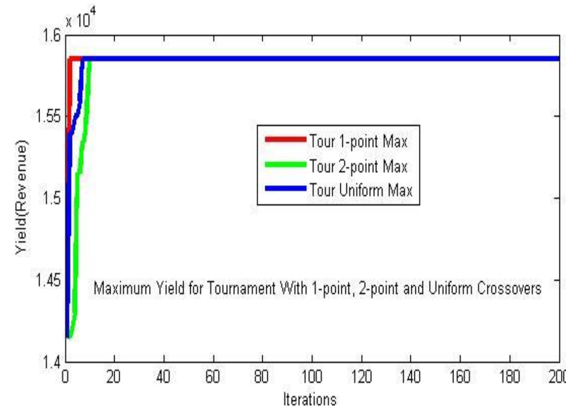

Fig.9 : Maximum Fitness

- Using combination of Tournament selection and the three types of cross-overs i.e. one-point, two-point and uniform, the following results were obtained during second time period:

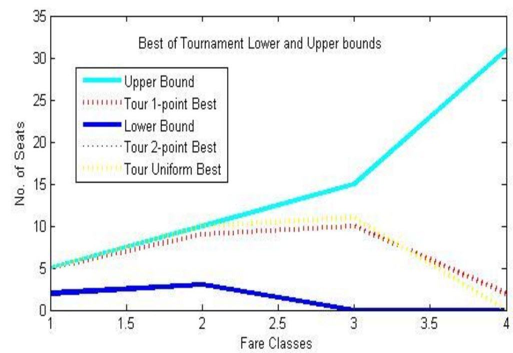

Fig.10 : Lower Bound, Upper Bound, and Estimated Fitness

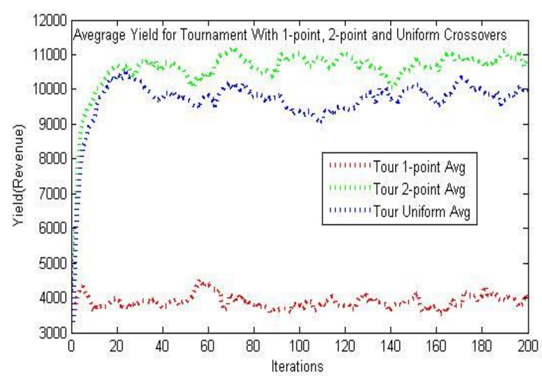

Fig.11 : Average Fitness

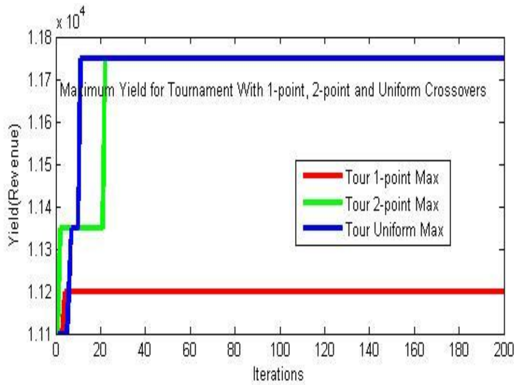

Fig.12 : Maximum Fitness

VII. INTERPRETATION

Upon comparing the results of two techniques implemented in this paper, it has been observed that the results obtained by both the methods are same as shown in table 1. Also the results can be compared with [4] and it can be observed that the solution found using GA and LPP Simulator are optimal and the results obtained are at par with [4]. Looking at the tables 1-4 and graphs shown in fig.1-fig.12 obtained by different combinations of various selection and cross-over methods the following findings has also been observed:

- By looking at table 2, it appears that during first time-period roulette-wheel selection and tournament selection with one-point crossover are best. But as visible from table in both the cases the convergence has been achieved in second iteration in every case, which is premature convergence. Although optimum results has been obtained still diversity in the population is lost
- Looking at table 3, it is clear that during second time-period roulette-wheel selection and tournament selection with one-point crossover are worst as the convergence is not obtained at all in any case. Overall, it can be said that both of these combinations should be completely discarded.
- By looking at table 4, it is clearly visible that uniform crossover with any type of selection are very good combination for finding the optimum results in both the time-periods and hence overall.
- Looking at fig. 2, fig.5, fig.8 and fig.11, it can be observed that the average population is found to be best in the case of two-point crossover along with any type of selection mechanism. Although uniform crossover is also found to be more or less equally good with any type of selection.
- Looking at table 5(a), 5(b), 6(a), 6(b), which are basically t-test table for checking whether any significant difference between mean of two populations exists or not. It can be observed that since P value < 0.05 in each case, the difference between the means found during first and second time-slice is significant. The first time-slice is providing much better results as compared to second time-slice. However, it can be said that the combination used in these tables i.e. roulette wheel with two-point and uniform crossovers are not consistent.
- By observing tables 7(a), 7(b), 8(a) and 8(b), it is clearly visible that since P value > 0.05 in each case, the difference between each time slice is not significant and is by chance. Hence the combination used in these tables i.e. tournament selection with two-point and uniform crossovers is much more consistent as compared to roulette-wheel selection.

TABLE1
LOWER, UPPER AND BEST ESTIMATED DEMANDS IN EACH ASSUMED FARE CLASS IN MULTIPLE TIME-SLICES

		Time-Slice I			Time-Slice II		
		Demand			Demand		
Fare Class	**Fare**	**Lower Limit**	**Upper Limit**	**Best Estimates**	**Lower Limit**	**Upper Limit**	**Best Estimates**
1	100	0	31	31	0	31	0
2	250	20	35	35	0	15	11
3	500	1	8	8	3	10	10
4	800	0	0	0	2	5	5

TABLE 2
NUMBER OF ITERATIONS FOR THE BEST ESTIMATES IN VARIOUS COMBINATIONS OF SELECTION AND CROSS-OVER PARAMETERS DURING FIRST TIME-SLICE

Parameters		Iterations		No. of simulation when max. is not achieved in specified iterations
Selection	Cross-over	Min. in all simulations	Avg. of all simulation except when max is not achieved	
Roulette Wheel	One-Point	2	2	Nil
Roulette Wheel	Two-Point	11	28	Nil
Roulette Wheel	Uniform	4	19	Nil
Tournament	One-Point	2	2	Nil
Tournament	Two-Point	6	60	1
Tournament	Uniform	6	49	Nil

TABLE 3
NUMBER OF ITERATIONS FOR THE BEST ESTIMATES IN VARIOUS COMBINATIONS OF SELECTION AND CROSS-OVER PARAMETERS DURING SECOND TIME-SLICE

Parameters		Iterations		No. of simulation when max. is not achieved in specified iterations
Selection	Cross-over	Min. in all simulations	Avg. of all simulation except when max is not achieved	
Roulette Wheel	One-Point	Nil	Nil	Nil
Roulette Wheel	Two-Point	15	65	2
Roulette Wheel	Uniform	9	54	1
Tournament	One-Point	Nil	Nil	Nil
Tournament	Two-Point	6	55	Nil
Tournament	Uniform	8	35	1

TABLE 4
NUMBER OF ITERATIONS FOR THE BEST ESTIMATES IN VARIOUS COMBINATIONS OF SELECTION AND CROSS-OVER PARAMETERS DURING MULTIPLE TIME-SLICES

Parameters		Iterations		No. of simulation when max. is not achieved in specified iterations
Selection	Cross-over	Min. in all simulations	Avg. of all simulation except when max is not achieved	
Roulette Wheel	One-Point	Nil	Nil	Nil
Roulette Wheel	Two-Point	15	65	2
Roulette Wheel	Uniform	9	54	1
Tournament	One-Point	Nil	Nil	Nil
Tournament	Two-Point	6	60	1
Tournament	Uniform	8	49	1

TABLE 5(A)
GROUP STATISTIC FOR ROULETTE-WHEEL SELECTION WITH 2-POINT CROSSOVER DURING 1ST AND 2ND TIME-SLICE

Group Statistics

	Slice	N	Mean	Std. Deviation	Std. Error Mean
RW1and2Slice	1.00	20	27.5000	12.42451	2.77820
	2.00	18	54.3333	35.35201	8.33255

TABLE 5(B)
INDEPENDENT SAMPLE T-TEST FOR ROULETTE-WHEEL SELECTION WITH 2-POINT CROSSOVER DURING 1ST AND 2ND TIME-SLICE

Independent Samples Test

		Levene's Test for Equality of Variances		t-test for Equality of Means						
									95% Confidence Interval of the Difference	
		F	Sig.	T	Df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	Lower	Upper
RW1and2Slice	Equal variances assumed	14.426	.001	-3.187	36	.003	-26.83333	8.41993	-43.90974	-9.75692
	Equal variances not assumed			-3.055	20.760	.006	-26.83333	8.78350	-45.11247	-8.55420

TABLE 6(A)
GROUP STATISTIC FOR ROULETTE-WHEEL SELECTION WITH UNIFORM CROSSOVER DURING 1ST AND 2ND TIME-SLICE

Group Statistics

	Slices	N	Mean	Std. Deviation	Std. Error Mean
RWUNI1and2slice	1.00	20	19.0500	12.71334	2.84279
	2.00	19	53.6316	43.98385	10.09059

TABLE 6(B)
INDEPENDENT SAMPLE T-TEST FOR ROULETTE-WHEEL SELECTION WITH UNIFORM CROSSOVER DURING 1ST AND 2ND TIME-SLICE

Independent Samples Test

	Levene's Test for Equality of Variances	t-test for Equality of Means

		F	Sig.	t	Df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	95% Confidence Interval of the Difference	
									Lower	Upper
RWUNI1and2slice	Equal variances assumed	16.835	.000	-3.373	37	.002	-34.58158	10.25230	-55.35471	-13.80845
	Equal variances not assumed			-3.299	20.846	.003	-34.58158	10.48339	-56.39277	-12.77039

TABLE 7(A)
GROUP STATISTIC FOR TOURNAMENT SELECTION WITH 2-POINT CROSSOVER DURING 1ST AND 2ND TIME-SLICE

Group Statistics

	Slicetour	N	Mean	Std. Deviation	Std. Error Mean
Tour2pt1and2slice	1.00	19	57.6316	66.32178	15.21526
	2.00	20	55.1500	70.91602	15.85730

TABLE 7(B)
INDEPENDENT SAMPLE T-TEST FOR TOURNAMENT SELECTION WITH 2-POINT CROSSOVER DURING 1ST AND 2ND TIME-SLICE

Independent Samples Test

		Levene's Test for Equality of Variances		t-test for Equality of Means						
		F	Sig.	T	Df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	95% Confidence Interval of the Difference	
									Lower	Upper
Tour2pt1and2slice	Equal variances assumed	.391	.536	.113	37	.911	2.48158	22.01505	-42.12514	47.08830

Independent Samples Test

		Levene's Test for Equality of Variances		t-test for Equality of Means						
									95% Confidence Interval of the Difference	
		F	Sig.	T	Df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	Lower	Upper
Tour2pt1and2slice	Equal variances assumed	.391	.536	.113	37	.911	2.48158	22.01505	-42.12514	47.08830
	Equal variances not assumed			.113	36.992	.911	2.48158	21.97631	-42.04696	47.01012

TABLE 8(A)
GROUP STATISTIC FOR TOURNAMENT SELECTION WITH UNIFORM CROSSOVER DURING 1ST AND 2ND TIME-SLICE

Group Statistics

	Slicetour	N	Mean	Std. Deviation	Std. Error Mean
Touruni1and2slice	1.00	19	47.2105	50.85008	11.66581
	2.00	19	34.5789	28.38880	6.51284

TABLE 8(B)
INDEPENDENT SAMPLE T-TEST FOR TOURNAMENT SELECTION WITH UNIFORM CROSSOVER DURING 1ST AND 2ND TIME-SLICE

Independent Samples Test

		Levene's Test for Equality of Variances		t-test for Equality of Means						
									95% Confidence Interval of the Difference	
		F	Sig.	t	Df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	Lower	Upper
Touruni1and2slice	Equal variances assumed	10.528	.003	.945	36	.351	12.63158	13.36069	-14.46517	39.72832

Independent Samples Test

		Levene's Test for Equality of Variances		t-test for Equality of Means						
		F	Sig.	t	Df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	95% Confidence Interval of the Difference Lower	Upper
Touruni1and2slice	Equal variances assumed	10.528	.003	.945	36	.351	12.63158	13.36069	-14.46517	39.72832
	Equal variances not assumed			.945	28.227	.352	12.63158	13.36069	-14.72665	39.98981

VIII. Conclusion

In this paper, the solution of the problem of yield management with multiple time-periods during arrival of the customers has been considered. GA has been used in the above problem with a number of variants in selection and crossover. Another technique of solving the problem that has been used is an LPP simulator. The results obtained from both the techniques are same and also match with earlier results, which prove GA can act as a very good tool for the solving yield management problem. 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 for finding the optimum results were tournament selection along with either uniform cross-over or two-point cross-over. 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 be improving solutions found by GA using some better techniques such as overbooking and cancellation, multi-segment flights, fitness scaling and elitism in GA etc.

References

[1] Mittal, Pardeep Kumar, Kumar Rakesh, Suri, P.K., “Genetic Simulator for Airline Yield Management”, *International Journal of Engineering Research & Technology(IJERT), ISSN 2278-0181, Vol3. Issue 9, September, 2013, pp. 2379-2386*

[2] Smith B., J. Leimkuhler, R. Darrow, J. Samuels. 1992. *Yield Management at American Airlines.* Interfaces, 22, 8-31.

[3] Aloysius George, B.R.Rajakumar, D. Binu, “Genetic Algorithm Based Airlines Booking Terminal Open/ Close Decision System”, *ICACCI’12, August 3-5, 2012, Chennai, T Nadu, India.*

[4] 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.*

[5] Suri P.K., Kumar Rakesh, Mittal Pardeep Kumar, “Simulating the Effect of Arrival Patterns in Yield Management”, *International Journal of Advanced Research in Computer Science and Software Engineering(IJARCSSE), Vol.3, Issue 9, September 2013, ISSN 2277 128X, pp. 534-544.*

[6] Maglaras Costis, Meissner Joern, “Dynamic Pricing Strategies for Multi-Product Revenue Management Problem”, *Manufacturing & Service Operations Management, Vol. 8, No. 2, Spring 2006, pp. 136–148*

[7] Ganji Alireza Inanlou and Shavandi Hassan, “Genetic algorithms for Uncertain Pricing-Inventory Models with Multiple Interdependent Products”, *International Journal of Mining, Metallurgy & Mechanical Engineering (IJMMME) Volume 1, Issue 1 (2013) ISSN 2320–4060 (Online)*

[8] Goldberg, D.E.(1989), *Genetic Algorithms in Search Optimization and Machine Learning,* Addison-Wesley, New York.